4. CLASSICAL (SECRET-KEY) CRYPTOSYSTEMS

In this chapter we deal with some of the very old or quite old cryptosystems that were primarily used in the pre-computer era.

These cryptosystems are too weak nowadays, too easy to break, especially with computers.

However, these simple cryptosystems give a good illustration of several of the important ideas of the

cryptography and cryptanalysis's.

SECRET-KEY CRYPTOGRAPHY

Cryptology (= cryptography + cryptoanalysis)
has more than two thousand years of history.

Basic historical observations

- People have always had fascination with keeping information away from others.
- Some people – rulers, diplomats, militaries, businessmen – have always had needs to keep some information away from others.

Importance of cryptography nowadays

- Applications: cryptography is the key tool to make secure modern information transmission, and to create information society...
- Foundations: cryptography gave rise to several new key concepts of the scientific foundation of informatics: one-way functions, computationally perfect pseudorandom generators, zero-knowledge proofs, holographic proofs, program self-testing and self-correcting, ...

Approaches and paradoxes of cryptography

Sound approaches to cryptography

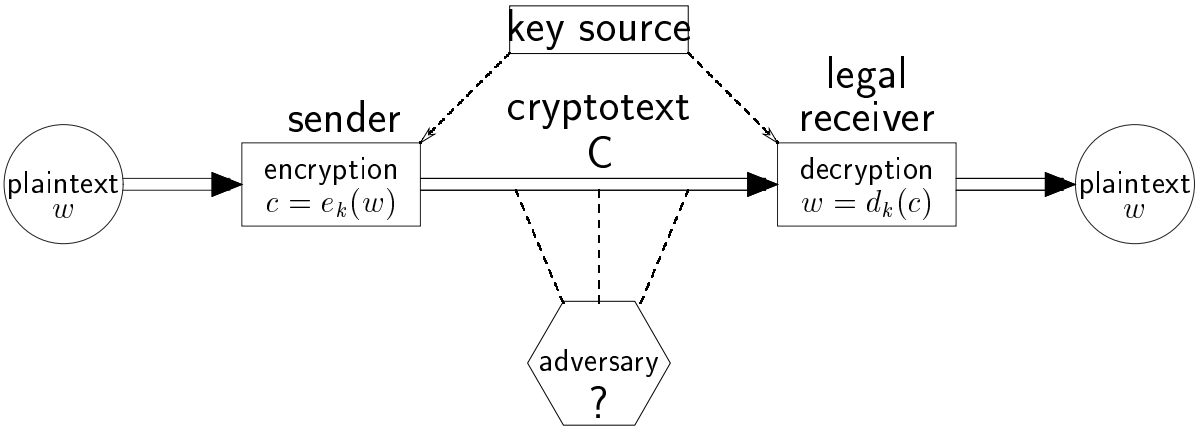
- Shannon's approach based on information theory
- Current approach based on complexity theory
- New approach based on the laws of quantum physics

Paradoxes of modern cryptography:

- Positive results of modern cryptography are based on negative results of complexity theory.
- Computers that were designed originally for cryptoanalysis seem to be now more useful for cryptography.

Cryptosystems – ciphers

The cryptography deals the problem of sending a message (plaintext, cleartext), through an insecure channel, that may be tapped by an adversary (eavesdropper, cryptanalyst), to a legal receiver.



Components of cryptosystems:

Plaintext-space: $\mathcal{P}$ – a set of plaintexts over an alphabet Σ
Cryptotext-space: $\mathcal{C}$ – a set of cryptotexts (ciphertexts) over alphabet Δ
Key-space: $\mathcal{K}$ – a set of keys

Each key k determines an encryption algorithm e_k and an decryption algorithm d_k such that, for any plaintext w, $e_k(w)$ is the corresponding cryptotext and

$$w = d_k(e_k(w)), \text{ or } w \in d_k(e_k(w))$$

(encryption algorithms can be randomized algorithms.)

100 – 42 B.C.
CAESAR cryptosystem
Shift cipher

to encrypt words in the English alphabet.

Key-space: $\{0, 1, \ldots, 25\}$

An encryption algorithm e_k substitutes any letter by the one occurring k positions ahead (cyclically) in the alphabet.

A decryption algorithm d_k substitutes any letter by the one occurring k positions backward (cyclically) in the alphabet.

Example $e_2(\text{EXAMPLE}) = \text{GZCOSNG}$,
$e_3(\text{EXAMPLE}) = \text{HADPTOH}$,

$e_1(\text{HAL}) = \text{IBM}$, $e_3(\text{COLD}) = \text{FROG}$

Example Find the plaintext to the following cryptotext obtained by the encryption with CAESAR with $k = ?$.

Cryptotext: VHFUHW GH GHXA, VHFUHW GH GLHX, VHFUHW GH WURLV, VHFUHW GH WRXV.

Numerical version of CAESAR is defined on the set $\{0, 1, 2, \ldots, 25\}$ by encryption algorithm:

$$e_k(i) = (i + k) \pmod{26}.$$

POLYBIOUS cryptosystem

to encrypt words in the English alphabet without J.

Key-space: Polybious checkerboards 5×5 with 25 English letters and with rows+columns labeled by symbols.

Example

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

Encryption algorithm: Each symbol is substituted by the pair of symbols, denoting the row and the column of the chalkboard in which the symbol is placed.

Decryption algorithm: ????

Kerckhoff's Principle

The philosophy of modern cryptoanalysis is embodied in the following principle formulated in 1883 by Jean Guillaume Hubert Victor François Alexandre Auguste Kerckhoffs von Nieuwenhof (1835 – 1903).

The security of a cryptosystem must not depend on keeping secret the encryption algorithm. The security should depend only on keeping secret the key.

Requirements for good cryptosystems (Sir Francis R. Bacon (1561 – 1626))

1. Given e_k and a plaintext w, it should be easy to compute $c = e_k(w)$.
2. Given d_k and a cryptotext c, it should be easy to compute $w = d_k(c)$.
3. A cryptotext $e_k(w)$ should not be much longer than the plaintext w.
4. It should be unfeasible to determine w from $e_k(w)$ without knowing d_k.
5. The so called avalanche effect should hold: A small change in the plaintext, or in the key, should lead to a big change of the cryptotext (i.e. a change of one bit of the plaintext should result in a change of all bits of the cryptotext with the probability close to 0.5).
6. The cryptosystem should not be closed under composition (i.e. not for every two keys k_1, k_2 there is a key k such that

$$e_k(w) = e_{k_1}(e_{k_2}(w))$$

for all plaintexts w.).
7. The set of keys should be VERY large.

Cryptoanalysis

The aim of cryptoanalysis is to get as much information about the plaintext or the key.

Main types of the cryptoanalysis attacks.

1. Cryptotexts-only attack. The cryptanalysts get cryptotexts

$$c_1 = e_k(w_1), \ldots, c_n = e_k(w_n)$$

and try to infer the key k or as many plaintexts $w_1, \ldots, w_n$ as possible.

2. Known-plaintexts attack. The cryptanalysts know some pairs

$$w_i, e_k(w_i), \quad 1 \leq i \leq n,$$

and try to infer k, or at least w_{n+1} for a new cryptotext $e_k(w_{n+1})$.

3. Chosen-plaintexts attack. The cryptanalysts choose plaintexts $w_1, \ldots, w_n$ to get cryptotexts $e_k(w_1), \ldots, e_k(w_n)$, and try to infer k or at least w_{n+1} for a new cryptotext $c_{n+1} = e_k(w_{n+1})$. (For example, if they get temporary access to an encryption machinery.)

4. Known-encryption-algorithm attack. The encryption algorithm e_k is given and the cryptanalysts try to get the decryption algorithm d_k.

5. Chosen-cryptotext attack. The cryptanalysts know some pairs

$$(c_i, d_k(c_i)), \quad 1 \leq i \leq n,$$

where the cryptotexts c_i have been chosen by the cryptanalysts. The aim is to determine the key. (For example, if cryptanalysts get a temporary access to a decryption machinery.)

WHAT CAN a BAD EVE DO?

Let us assume that Alice sends an encrypted message to Bob. What can a bad enemy, called usually Eve (eavesdropper), do?

- She can read (and try to decrypt) the message.
- She can try to get the key that was used and then decrypt all messages encrypted with the same key.
- She can change the message sent by Alice into another message, in such a way that Bob will have the feeling, after he gets the changed message, that it was from Alice.
- She can pretend to be Alice and communicate with Bob, in such a way that Bob thinks he is communicating with Alice.

An eavesdropper can therefore be passive – Eve or active – Mallot.

BASIC GOALS

of

BROADLY UNDERSTOOD CRYPTOGRAPHY

Confidentiality: Eve should not be able to decrypt the message Alice sends to Bob.

Data integrity: Bob wants to be sure that Alice's message has not been altered by Eve.

Authentication: Bob wants to be sure that only Alice could have sent the message he has received.

Non-repudiation: Alice cannot claim that she did not send messages she did send.

The cryptosystem presented in this slide was probably never used. In spite of that this cryptosystem played an important role in the history of modern cryptography.

HILL cryptosystem

for a fixed n and the English alphabet.

Key-space: matrices M of degree n with elements from the set $\{0, 1, \ldots, 25\}$ such that $M^{-1} \mod 26$ exists.

Plaintext + cryptotext space: English words of length $k \times n$ for some k.

Encoding: For a block-word w of length n let c_w be the column vector of length n of the integer codes of symbols of w. $(A \to 0, B \to 1, C \to 2, \ldots)$

Encryption: $c_c = Mc_w \mod 26$

Decryption: $c_w = M^{-1}c_c \mod 26$

Example $M = \begin{pmatrix} 4 & 7 \\ 1 & 1 \end{pmatrix}, \quad M^{-1} = \begin{pmatrix} 17 & 11 \\ 9 & 16 \end{pmatrix}$

Plaintext $w =$ LONDON

$$c_{LO} = \begin{pmatrix} 11 \\ 14 \end{pmatrix}, \quad c_{ND} \begin{pmatrix} 13 \\ 3 \end{pmatrix}, \quad c_{ON} = \begin{pmatrix} 14 \\ 13 \end{pmatrix}$$

$$Mc_{LO} = \begin{pmatrix} 12 \\ 25 \end{pmatrix}, \quad Mc_{ND} \begin{pmatrix} 21 \\ 16 \end{pmatrix}, \quad Mc_{ON} = \begin{pmatrix} 17 \\ 1 \end{pmatrix}$$

Cryptotext: MZVQRB

Theorem If $M = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$, then $M^{-1} = \frac{1}{\det M} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}$.

Proof: Exercise.

SECRET-KEY CRYPTOSYSTEMS

A cryptosystem is called secret-key cryptosystem if some secret piece of information – the key – has to be agreed first between any two parties that have, or want, to communicate through the cryptosystem. Examples: CAESAR, HILL

Two basic types of secret-key cryptosystems:
- **substitution** based **cryptosystems**
- **transposition** based **cryptosystems**

Two basic types of substitution cryptosystems
- **monoalphabetic** – a fixed substitution – CAESAR, POLYBIOUS
- **polyalphabetic** – substitution keeps changing during encryption

A monoalphabetic cryptosystem with letter-by-letter substitution is uniquely specified by a permutation of letters. (Number of permutations (keys) is $26!$)

Example AFFINE cryptosystem is given by two integers

$$1 \leq a, b \leq 25, \gcd(a, 26) = 1.$$

Encryption: $e_{a,b}(x) = ((ax + b) \mod 26)$

Example $a = 3, b = 5 \quad e_{3,5}(x) = (3x + 5) \mod 26$
$e_{3,5}(3) = 14, e_{3,5}(15) = 24$ — $e_{3,5}(D) = 0, e_{3,5}(P) = Y$

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

Decryption: $d_{a,b}(y) = a^{-1}(y - b) \mod 26.$

Cryptanalysis's

The basic cryptanalytic attack against monoalphabetic substitution cryptosystems begins with a frequency count: the number of each letter in the cryptotext is counted. The distribution of letters in the cryptotext is then compared with the distribution of letters in the plaintext language.

The letter with the highest frequency in the cryptotext is likely to be substitute for the letter with highest frequency in the plaintext language .... The likehood grows with the length of cryptotext.

Frequency counts for English:

	%		%		%
E	12.31	L	4.03	B	1.62
T	9.59	D	3.65	G	1.61
A	8.05	C	3.20	V	.93
O	7.94	U	3.10	K	.52
N	7.19	P	2.29	Q	.20
I	7.18	F	2.28	X	.20
S	6.59	M	2.25	J	.10
R	6.03	W	2.03	Z	.09
H	5.14	Y	1.88		5.27
	70.02		24.71		

and for other languages:

English	%	German	%	Finnish	%
E	12.31	E	18.46	A	12.06
T	9.59	N	11.42	I	10.59
A	8.05	I	8.02	T	9.76
O	7.94	R	7.14	N	8.64
N	7.19	S	7.04	E	8.11
I	7.18	A	5.38	S	7.83
S	6.59	T	5.22	L	5.86
R	6.03	U	5.01	O	5.54
H	5.14	D	4.94	K	5.20

French	%	Italian	%	Spanish	%
E	15.87	E	11.79	E	13.15
A	9.42	A	11.74	A	12.69
I	8.41	I	11.28	O	9.49
S	7.90	O	9.83	S	7.60
T	7.26	N	6.88	N	6.95
N	7.15	L	6.51	R	6.25
R	6.46	R	6.37	I	6.25
U	6.24	T	5.62	L	5.94
L	5.34	S	4.98	D	5.58

The 20 most common diagrams are (in decreasing order) TH, HE, IN, ER, AN, RE, ED, ON, ES, ST, EN, AT, TO, NT, HA, ND, OU, EA, NG, AS. The six most common trigrams: THE, ING, AND, HER, ERE, ENT.

Cryptoanalysis of a cryptotext encrypted using the AFINE cryptosystem with an encryption algorithm

$$e_{a,b}(x) = ax + b \mod 26,$$

where $0 \leq a, b \leq 25, \gcd(a, 26) = 1$. (Number of keys: $12 \times 26 = 312$.)

Example Assume that an English plaintext is divided into blocks of 5 letter and encrypted by an AFINE cryptosystem (ignoring spaces and interpunctions) as follows:

How to find the plaintext?

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

Frequency analysis provides:

X–32	J–11	D–2
U–30	O– 6	V–2
H–23	R– 6	F–1
B–19	G– 5	P–1
L–19	M– 4	E–0
N–16	Y– 4	I–0
K–15	Z– 4	Q–0
S–15	C– 3	T–0
W–14	A– 2	

Frequency table for English

	%		%		%
E	12.31	L	4.03	B	1.62
T	9.59	D	3.65	G	1.61
A	8.05	C	3.20	V	.93
O	7.94	U	3.10	K	.52
N	7.19	P	2.29	Q	.20
I	7.18	F	2.28	X	.20
S	6.59	M	2.25	J	.10
R	6.03	W	2.03	Z	.09
H	5.14	Y	1.88		5.27
	70.02		24.71		

First guess: $E = X,\ T = U$

Equations: $4a + b = 23 (\mathrm{mod}\ 26)$
$19a + b = 20 (\mathrm{mod}\ 26)$

Solution: $a = 5, b = 3.$

Translation table

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

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

provides the plaintext that starts with

KGWTG CKTMO OTMIT DMZEG

what does not make a sense.

Second guess: $E = X, A = H$

Equations: $4a + b = 23\ (\mathrm{mod}\ 26)$
$b = 7 (\mathrm{mod}\, 26)$

Solutions: $a \not\equiv 4$ or $a = 17$

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

Plaintext:

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

Example of a monoalphabetic cryptosystem

Symbols of the English alphabet will be replaced by squares with or without points and with or without surrounding lines using the following rule:

A:	B:	C:
D:	E:	F:
G:	H:	I:

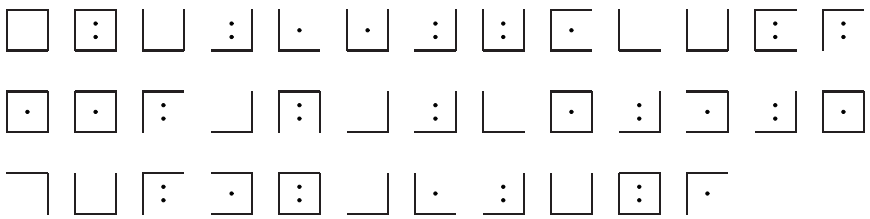
J·	K·	L·
M·	N·	O·
P·	Q·	R·

S	T	U
V	W	X
Y	Z	

For example the plaintext:

WE TALK ABOUT FINNISH SAUNA MANY TIMES LATER

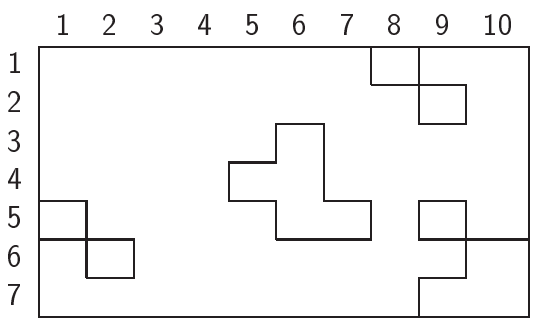
results in the cryptotext:



Garbage in between method: the message (plaintext or cryptotext) is supplemented by "garbage letters".

I		L	O	V	E		Y	O	U
I		H	A	V	E		Y	O	U
D	E	E	P		U	N	D	E	R
M	Y		S	K	I	N		M	Y
L	O	V	E		L	A	S	T	S
F	O	R	E	V	E	R		I	N
H	Y	P	E	R	S	P	A	C	E

Richelieu cryptosystem used sheets of card board with holes.



Polyalphabetic Substitution Cryptosystems

Playfair cryptosystem

Invented around 1854 by Ch. Wheatstone.

Key – a Playfair square or a word or text in which repeated letters are then removed and the remaining letters of alphabets are added and divided to form an array.

Encryption: of a pair of letters x, y

- If x and y are neither in the same row nor in the same column, then the smallest rectangle containing x, y is taken and symbols xy are replaced by the pair of symbols in the remaining corners of the square.
- If x and y are in the same row (column), then they are replaced by the pair of symbols to the right (bellow) them.

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

Playfair square

Example PLAYFAIR is encrypted as LCMNNFCS

Playfair was used in World War I by British army.

VIGENERE and AUTOCLAVE cryptosystems

Several polyalphabetic cryptosystems are the following modification of the CAESAR cryptosystem.

A 26×26 table is first designed with the first row containing a permutation of all symbols of alphabet and all columns represent CAESAR shifts starting with the symbol of the first row.

Secondly, for a plaintext w and a key k – a word of the same length as w.

Encryption: the i-th letter of the plaintext – w_i – is replaced by the letter in the w_i-row and k_i-column of the table.

VIGENERE cryptosystem: a short keyword p is chosen and

$$k = Prefix_{|w|}p^{\infty}$$

VIGENERE is actually a cyclic version of the CAESAR cryptosystem:

AUTOCLAVE cryptosystem: $k = Prefix_{|w|}pw$.

Example

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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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	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

Keyword:	HAMBURG
Plaintext:	INJEDEMMENSCHENGESICHTESTEHTSEINEG
Vigenere-key:	HAMBURGHAMBURGHAMBURGHAMBURGHAMBUR
Autoclave-key:	HAMBURGINJEDEMMENSCHENGESICHTESTEH
Vigerere-cryp.:	PNVFXVSTEZTWYKUGQTCTNAEEVYYZZEUOYX
Autoclave-cryp.:	PNVFXVSURWWFLQZKRKKJLGKWLMJALIAGIN

CRYPTOANALYSIS of cryptotexts produced by VINEGAR cryptosystem

1. Task 1 – to find the length of the key

Kasiski method (1852) – invented also by Charles Babbage (1853).

Basic observation If a subword of a plaintext is repeated at a distance that is a multiple of the length of the key, then the corresponding subwords of the cryptotext are the same.

Example: cryptotext: CHRGQPWOEIRULYANDOSHCHRIZKEBUSNOFKYWROPDCHR

Substring "CHR" occurs in positions 1, 21, 41, 66: expected keyword length is therefore 5.

Method. Determine the greatest common divisor of the distances between identical subwords (of length 3 or more) of the cryptotext.

Friedman method Let n_i be the number of occurrences of i-th letter in the cryptotext. Let l be the length of the keyword. Let n be the length of the cryptotext. Then it holds

$$l \doteq \frac{0.027n}{(n-1)I - 0.038n + 0.065}, \qquad I = \sum_{i=1}^{26} \frac{n_i(n_i-1)}{n(n-1)}$$

Once the length of the keyword is found it is easy to determine the key using the statistical method of analyzing monoalphabetic cryptosystems.

Derivation of the Friedman method

1. Let n_i be the number of occurrences of i-th alphabet symbol in a text of length n.

The probability that if one selects a pair of symbols from the text, then they are the same is

$$I = \frac{\sum_{i=1}^{26} n_i(n_i-1)}{n(n-1)} = \sum_{i=1}^{26} \frac{\binom{n_i}{2}}{\binom{n}{2}}$$

and it is called **index of coincides**.

2. Let p_i be the probability that a randomly chosen symbol is the i-th symbol of the alphabet. The probability that two randomly chosen symbol are the same is

$$\sum_{i=1}^{26} p_i^2$$

For English text one has

$$\sum_{i=1}^{26} p_i^2 = 0.065$$

For randomly chosen text:

$$\sum_{i=1}^{26} p_i^2 = \sum_{i=1}^{26} \frac{1}{26^2} = 0.038$$

Approximately

$$I = \sum_{i=1}^{26} p_i^2$$

Assume that a cryptotext is organized into l columns headed by the letters of the keyword

letters S_i of keyword	S_1	S_2	S_3	$\ldots$	S_l
	x_1	x_2	x_3	$\ldots$	x_l
	x_{l+1}	x_{l+2}	x_{l+3}	$\ldots$	x_{2l}
	x_{l+1}	x_{l+2}	x_{l+3}	$\ldots$	x_{3l}
	$\vdots$	$\vdots$	$\vdots$		$\vdots$

First observation Each column is obtained using the CAESAR cryptosystem.

Probability that two randomly chosen letters are the same in

the same column
different columns
is $\begin{matrix} 0.64 \\ 0.38 \end{matrix}$.

The number of pairs of letters in the same column:

$$l\binom{\frac{n}{l}}{2} = \frac{l}{2} \cdot \frac{n}{l}\left(\frac{n}{l} - 1\right) = \frac{n(n-l)}{2l}$$

The number of pairs of letters in different columns:

$$\frac{l(l-1)}{2} \cdot \frac{n^2}{l^2} = \frac{n^2(l-1)}{2l}$$

The expect number A of pairs of equals letters is

$$A = \frac{n(n-l)}{2l}0.065 + \frac{n^2(l-1)}{2l}0.038$$

Since

$$I = \frac{A}{\frac{n(n-1)}{2}} = \frac{1}{l(n-1)}[0.027n + l(0.038n - 0.065)]$$

one gets the above formula for l.

One-time pad cryptosystem

Binary case:

plaintext w
key k
cryptotext c

} are binary words of the same length

Encryption: $c = w \oplus k$

Decryption: $w = c \oplus k$

Example:

$$\begin{aligned} w &= 101101011 \\ k &= 011011010 \\ c &= 110110001 \end{aligned}$$

What happens if the same key is used twice or 3 times for encryption?

$$c_1 = w_1 \oplus k, c_2 = w_2 \oplus k, c_3 = w_3 \oplus k$$

$$c_1 \oplus c_2 = w_1 \oplus w_2$$

$$c_1 \oplus c_3 = w_1 \oplus w_3$$

$$c_2 \oplus c_3 = w_2 \oplus w_3$$

Perfect secret cryptosystems

By Shanon, a cryptosystem is perfect if the knowledge of the cryptotext provides no information whatsoever about its plaintext (with the exception of its length).

It follows from Shannon's results that perfect secrecy is possible if the key-space is as large as the plaintext-space. In addition, a key has to be as long as plaintext and the same key should not be used twice.

An example of a perfect cryptosystem ONE-TIME PAD cryptosystem (Gilbert S. Vernam (1917) – AT&T + Major Joseph Mauborgne).

If used with the English alphabet, it is simply a polyalphabetic substitution cryptosystem of VIGENERE with the key being a randomly chosen English word of the same length as the plaintext.

Proof of perfect secrecy: by the proper choice of the key any plaintext of the same length could provide the given cryptotext.

Did we gain something? The problem of secure communication of the plaintext got transformed to the problem of secure communication of the key of the same length.

Yes: 1. ONE-TIME PAD cryptosystem is used in critical applications
2. It suggests an idea how to construct practically secure cryptosystems.

Transposition Cryptosystems

The basic idea is very simple: permutate the plaintext to get the cryptotext. Less clear it is how to specify and perform efficiently permutations.

One idea: choose n, write plaintext into rows, with n symbols in each row and then read it by columns to get cryptotext.

Example

I N J E D E M M E N
S C H E N G E S I C
H T E S T E H T S E
I N E G E S C H I C
H T E T O J E O N O

Cryptotexts obtained by transpositions, called anagrams, were popular among scientists of 17th century. They were used also to encrypt scientific findings.

Newton wrote to Leibnitz

$$a^7c^2d^2e^{14}f^2i^7l^3m^1n^8o^4q^3r^2s^4t^8v^{12}x^1$$

what stands for

"data aequatione quodcumque fluentes quantitates involvente, fluxiones invenire et vice versa"

Example $a^2cdef^3g^2i^2jkmn^3o^5prs^2t^2u^3z$

Solution:

KEYWORD CAESAR cryptosystem

Choose a $0 < k < 25$ and a string (word/sentence), called keyword, with at most 25 different symbols.

The keyword is now written bellow the alphabet, beginning with the k-symbol and the remaining letters are written in the alphabetic order after the keyword.

Example Keyword: HOW MANY ELKS

$k = 8$

0							8																	
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
P	Q	R	T	U	V	X	Z	H	O	W	M	A	N	Y	E	L	K	S	B	C	D	F	G	I	J

Exercise Decrypt the following cryptotext encrypted using the KEYWORD CAESAR and determine the keyword and k

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

Step 1. Make the frequency counts:

	Number
U	32
C	31
Q	23
F	22
V	20
P	15
T	15
I	14
A	8
	$180 = 74.69\%$

	Number
X	8
K	7
N	7
E	6
M	6
R	6
B	5
Z	5
D	4
	$54 = 22.41\%$

	Number
W	3
Y	2
G	1
H	1
J	0
L	0
O	0
S	0
	$7 = 2.90\%$

Step 2. Cryptotext contains two one-letter words T and Q. They must be A and I. Since T occurs once and Q three times it is likely that T is I and Q is A.

The three letter word UPC occurs 7 times and all other 3-letter words occur only once. Hence

UPC is likely to be THE.

Let us now decrypt the remaining letters in the high frequency group: F,V,I

From the words TU, TF $\Rightarrow$ F=S

From UV $\Rightarrow$ V=O

From VI $\Rightarrow$ I=N

The result after the remaining guesses

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
L V E W P S K M N ? Y ? R U ? H A F ? I T O B C G D

ANAGRAMS — EXAMPLES

German:

IRI BRÄTER, GENF	Briefträgerin
FRANK PEKL, REGEN	...
PEER ASSSTIL, MELK	...
INGO DILMR, PEINE	...
EMIL REST, GERA	...
KARL SORDORT, PEINE	...

English

algorithms	*logarithms*
antagonist	*stagnation*
compressed	*decompress*
coordinate	*decoration*
creativity	*reactivity*
deductions	*discounted*
descriptor	*predictors*
impression	*permission*
introduces	*reductions*
procedures	*reproduces*

CRYPTOSYSTEMS WITH STREAMS OF KEYS

In some cryptosystems the same key can be used to encrypt arbitrarily long plaintext. In other cryptosystems with a given key only plaintexts of a fixed length can be encrypted. In such a case two basic encryption techniques are used to encode a sequence of plaintexts (obtained by dividing each long plaintext w into a sequence of subplaintexts $w_1w_2w_3\ldots$).

- The fixed key k is used to encrypt all subplaintexts. In such a case the resulting cryptotext has the form

$$c = c_1c_2c_3\ldots = e_k(w_1)e_k(w_2)e_k(w_3)\ldots$$

- A stream of keys is used to encrypt subplaintexts. The basic idea is to generate a key-stream $K = k_1, k_2, k_2, \ldots$ and then to compute the cryptotext as follows

$$c = c_1c_2c_3\ldots e_{k_1}(w_1)e_{k_2}(w_2)e_{k_3}(w_3).$$

Various techniques are used to compute keys. For example, given a key k

$$k_i = f_i(k, x_1, x_2, \ldots, x_{i-1})$$

In such a case encryption and decryption processes generate the following sequences:

Encryption: To encrypt the plaintext $w_1w_2w_3\ldots$ the sequence

$$k_1, c_1, k_2, c_2, k_3, c_3, \ldots$$

of keys and sub-cryptotexts is computed.

Decryption: To decrypt the cryptotext $c_1c_2c_3\ldots$ the sequence

$$k_1, w_1, k_2, w_2, k_3, w_3, \ldots$$

of keys and subplaintexts is computed.

EXAMPLES

A keystream is called **synchronous** if it is independent of the plaintext.

KEYWORD VIGENERE cryptosystem can be seen as an example of a synchronous keystream cryptosystem.

Another type of the binary keystream cryptosystem is specified by an initial sequence of keys

$$k_1, k_2, k_3 \ldots, k_m$$

and an initial sequence of binary constants

$$b_1, b_2, b_3, \ldots, b_{m-1}$$

and the remaining keys are computed using the rule

$$k_{i+m} = \sum_{j=0}^{m-1} b_j k_{i+j} \bmod 2$$

A keystrem is called **periodic** with period p if $k_{i+p} = k_i$ for all i.

Example Let the keystream is generated by the rule

$$k_{i+4} = k_i \oplus k_{i+1}.$$

If the initial sequence of keys is (1,0,0,0), then we get the following keystream:

$$1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 01, 1, 1, \ldots$$

of period 15.

PERFECT SECRECY — BASIC CONCEPTS

Let $\mathbf{P}$, $\mathbf{K}$ and $\mathbf{C}$ be sets of plaintexts, keys and cryptotexts.

Let $p_{\mathbf{K}}(k)$ be the probability that the key k is chosen from $\mathbf{K}$ and let a priory probability that plaintext w is chosen is $p_{\mathbf{P}}(w)$.

If for a key $k \in \mathbf{K}$, $C(k) = \{e_k(w) \,|\, w \in \mathbf{P}\}$, then for the probability $p_{\mathbf{C}}(y)$ that c is the cryptotext that is transmitted it holds

$$p_{\mathbf{C}}(c) = \sum_{\{k \,|\, c \in C(k)\}} p_{\mathbf{K}}(k) p_{\mathbf{P}}(d_k(c)).$$

For the conditional probability $p_{\mathbf{C}}(c|w)$ that c is the cryptotext if w is the plaintext it holds

$$p_{\mathbf{C}}(c|w) = \sum_{\{k \,|\, w = d_k(c)\}} p_{\mathbf{K}}(k).$$

Using Bayes' conditional probability formula $p(y)p(x|y) = p(x)p(y|x)$ we get for probability $p_{\mathbf{P}}(w|c)$ that w is the plaintext if c is the cryptotext the expression

$$p_{\mathbf{P}}(w|c) = \frac{P_{\mathbf{P}}(w) \, \Sigma_{\{k \,|\, w = d_k(c)\}} \, p_{\mathbf{K}}(k)}{\Sigma_{\{k \,|\, c \in C(\mathbf{K})\}} \, p_{\mathbf{K}}(k) p_{\mathbf{P}}(d_{\mathbf{k}}(c))}.$$

PERFECT SECRECY – BASIC RESULTS

Definition A cryptosystem has perfect secrecy if

$$p_{\mathbf{P}}(w|c) = p_{\mathbf{P}}(w) \text{ for all } w \in \mathbf{P} \text{ and } c \in \mathbf{C}.$$

(That is, the a posteriori probability that the plaintext is w, given that the cryptotext is c is obtained, is the same as a priori probability that the plaintext is w.)

Example CAESAR cryptosystem has perfect secrecy if any of the 26 keys is used with the same probability to encode any symbol of the plaintext.

Proof. Exercise.

An analysis of perfect secrecy: The condition $p_{\mathbf{P}}(w|c) = p_{\mathbf{P}}(w)$ is for all $w \in \mathbf{P}$ and $c \in \mathbf{C}$ equivalent to the condition $p_{\mathbf{C}}(c|w) = p_{\mathbf{C}}(c)$.

Let us now assume that $p_{\mathbf{C}}(c) > 0$ for all $c \in \mathbf{C}$.

Fix $w \in \mathbf{P}$. For each $c \in \mathbf{C}$ we have $p_{\mathbf{C}}(c|w) = p_{\mathbf{C}}(c) > 0$. Hence, for each $c \in \mathbf{C}$ there must exists at least one key k such that $e_k(w) = c$. Consequently, $|\mathbf{K}| \geq |\mathbf{C}| \geq |\mathbf{P}|$.

In a special case $|\mathbf{K}| = |\mathbf{C}| = |\mathbf{P}|$ the following nice characterization of the perfect secrecy can be obtained:

Theorem A cryptosystem in which $|\mathbf{P}| = |\mathbf{K}| = |\mathbf{C}|$ provides perfect secrecy if and only if every key is used with the same probability and for every $w \in \mathbf{P}$ and every $c \in \mathbf{C}$ there is a unique key k such that $e_k(w) = c$.

Proof. Exercise.

PRODUCT CRYPTOSYSTEMS

A cryptosystem $\mathcal{S} = (\mathcal{P}, \mathcal{K}, \mathcal{C}, e, d)$ with the sets of plaintexts $\mathcal{P}$, keys $\mathcal{K}$ and cryptotexts $\mathcal{C}$ and encryption (decryption) algorithms e (d) is called **endomorphic** if $\mathcal{P} = \mathcal{C}$.

If $\mathcal{S}_1 = (\mathcal{P}, \mathcal{K}_1, \mathcal{P}, e^{(1)}, d^{(1)})$ and $\mathcal{S}_2 = (\mathcal{P}, \mathcal{K}_2, \mathcal{P}, e^{(2)}, d^{(2)})$ are endomorphic cryptosystems, then the **product cryptosystem** is

$$\mathcal{S}_1 \otimes \mathcal{S}_2 = (\mathcal{P}, \mathcal{K}_1 \otimes \mathcal{K}_2, \mathcal{P}, e, d),$$

where encryption is performed by the procedure

$$e_{(k_1,k_2)}(w) = e_{k_2}(e_{k_1}(w))$$

and decryption by the procedure

$$d_{(k_1,k_2)}(c) = d_{k_1}(d_{k_2}(c)).$$

Example (Multiplicative cryptosystem): $\mathcal{P} = \mathcal{C} = \mathbf{Z_{26}}$ and $\mathcal{K} = \{a \in \mathbf{Z}_{26}, \gcd(a, 26) = 1\}$. Encryption: $e_a(w) = aw \bmod p$; decryption: $d_a(c) = a^{-1}c \bmod 26$.

If M denote the multiplicative cryptosystem, then clearly CAESAR $\otimes$ M is actually the AFFINE cryptosystem.

Two cryptosystems $\mathcal{S}_1$ and $\mathcal{S}_2$ are called **commutative** if $\mathcal{S}_1 \otimes \mathcal{S}_2 = \mathcal{S}_2 \otimes \mathcal{S}_1$. A cryptosystem $\mathcal{S}$ is called **idempotent** if $\mathcal{S} \otimes \mathcal{S} = \mathcal{S}$.

Exercise: Which of the cryptosystems described in this chapter are idempotent?