Cryptography and computer security

Cryptography

Literature

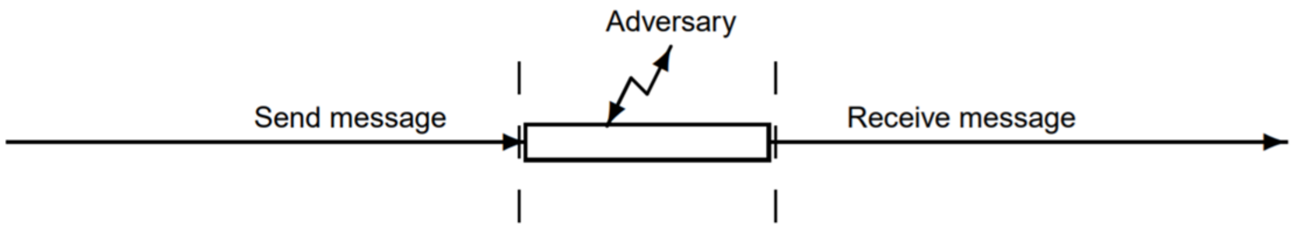
- https://www.cs.vsb.cz/ochodkova/courses/kpb/cryptography-and-network-security_-principles-and-practice-7th-global-edition.pdf chapters 3.1., 3.2.
- https://cacr.uwaterloo.ca/hac/ chapters 1.1 - 1.5, 7.3

Cryptology

- **Cryptology** - The art and science of making and breaking “secret codes”, the study of techniques for ensuring the secrecy and/or authenticity of information, includes both cryptography and cryptanalysis
  - **Cryptography** - making “secret codes”, it means hidden writing, and it refers to the practice of using encryption to conceal text
  - **Cryptanalysis** - it studies encryption and encrypted messages, hoping to find the hidden meanings, breaking “secret codes”

Why crptography?

- Cryptography deals with the development of algorithms that can be used to:
  - confidentiality of messages (their content, not their existence),
  - authentication - the traceability of the origin of a message, secure identification of the subject who
    - the information created,
    - who receives it,
    - who handles it.
  - Integrity control - information can only be modified/generated by an authorised entity,
  - to ensure nonrepudiation (undeniability)
    - delivery
    - And the origin of information.



Basic Terminology

- Plaintext - the original intelligible message, M, P
- Ciphertext the transformed message, C
- Cipher - an algorithm for transforming an intelligible message into one that is unintelligible
- Key - some critical information used by the cipher, known only to the sender & receiver, K
- Encrypt (encipher) - the process of converting plaintext to ciphertext using a cipher and a key, E
- Decrypt (decipher) - the process of converting ciphertext back into plaintext using a cipher and a key, D
- Correctness condition: $\forall E_k : M \rightarrow C$ and $\forall D_k : C \rightarrow M$: $D_k(E_k(M)) = M$, for $\forall M$, $\forall K$

How to „Speak“ Crypto

- A cipher or cryptosystem is used to encrypt the plaintext
- The result of encryption is ciphertext
- We decrypt ciphertext to recover plaintext
- A key is used to configure a cryptosystem
- A symmetric key cryptosystem uses the same key to encrypt as to decrypt
- A asymmetric (public key) cryptosystem uses a public key to encrypt and a private key to decrypt (or private key to digital signing and public one to signature verification)

Kerckhoffs’ Principle

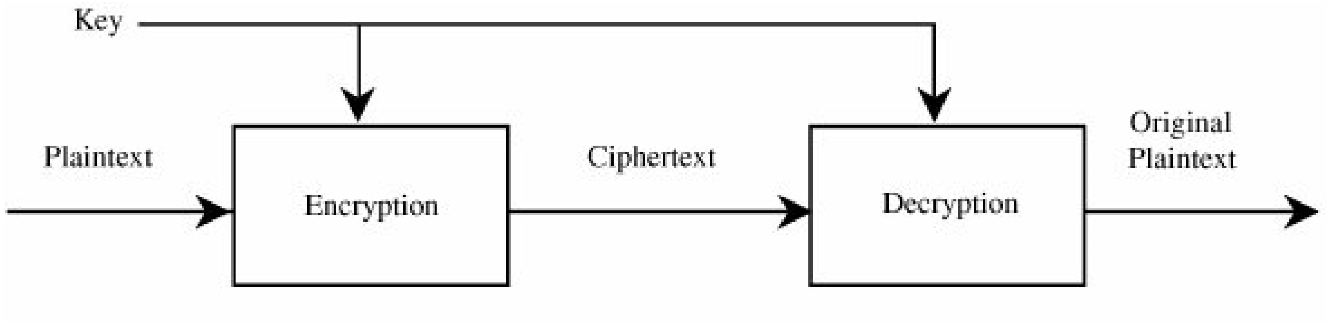
- Basic assumptions
  - The system is completely known to the attacker
  - Only the key is secret
  - That is, crypto algorithms are not secret
- This is known as **Kerckhoffs’ Principle** (1883 Auguste Kerckhoffs)
- Why do we make this assumption?
  - Experience has shown that proprietary algorithms are weak when exposed
  - Proprietary algorithms never remain secret
  - Better to find weaknesses beforehand

Classifying Cryptographic Algorithms

- Cryptographic algorithms
  - Symmetric (secret-key) encryption algorithms
    - block ciphers
    - stream ciphers
  - Asymmetric (public-key) encryption algorithms
  - Digital signature algorithms
  - Hash functions
  - PRNGs

Symmetric Cipher Model

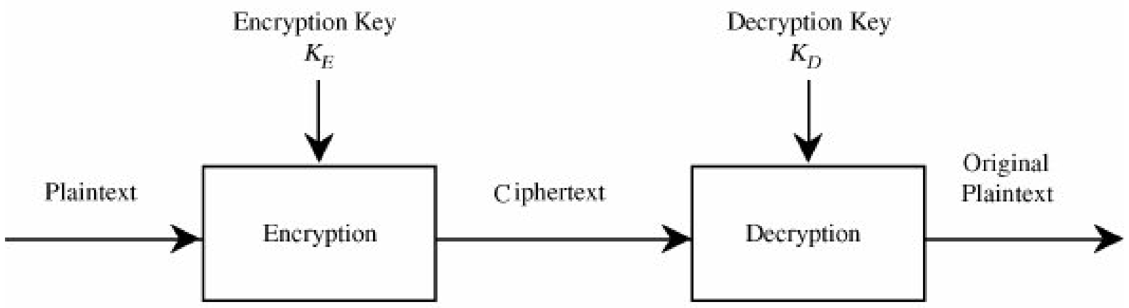
- A symmetric encryption algorithm is one where the sender and the recipient share a common secret key.
- All traditional (historical) encryption algorithms are symmetric.
- https://en.wikipedia.org/wiki/Symmetric-key_algorithm



(a) Symmetric Cryptosystem

Asymmetric Cipher Model

- The sender and receiver use different keys, wo keys (public and private)
  - Sender uses recipient's **public key** to encrypt, recipient uses **private key** to decrypt.
  - Or sender uses his private key to sign, recipient uses sender's public key to verify signature.
  - https://en.wikipedia.org/wiki/Public-key_cryptography



(b) Asymmetric Cryptosystem

Brute-force attack

- or exhaustive key search, https://en.wikipedia.org/wiki/Brute-force_attack
- Always theoretically possible to simply try every key
- Most basic attack, directly proportional to key size
- Assume either know or can recognise when plaintext is found
- Tabulate for reasonable assumptions about number of operations possible →

Brute-force attack

Key size (bits)	Number of alternative keys	Time required at 1 decryption/μs	Time required at 10^6 decryption/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

Suggested key sizes and other parameters

NIST Recommendations (2016) - Page 2

Keys length recommendations

Date	Minimum of Strength	Symmetric Algorithms	Factoring Modulus	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash (A)	Hash (B)
(Legacy)	80	2TDEA*	1024	160	1024	160	SHA-1**	
2016 - 2030	112	3TDEA	2048	224	2048	224	SHA-224 SHA-512/224 SHA3-224	
2016 - 2030 & beyond	128	AES-128	3072	256	3072	256	SHA-256 SHA-512/256 SHA3-256	SHA-1
2016 - 2030 & beyond	192	AES-192	7680	384	7680	384	SHA-384 SHA3-384	SHA-224 SHA-512/224
2016 - 2030 & beyond	256	AES-256	15360	512	15360	512	SHA-512 SHA3-512	SHA-256 SHA-512/256 SHA-384 SHA-512 SHA3-512

All key sizes are provided in bits. These are the minimal sizes for security.

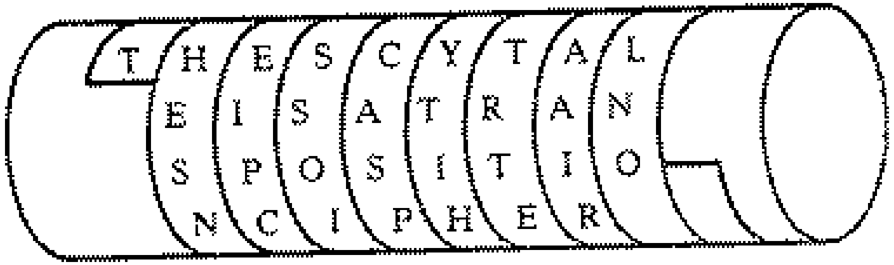
TDEA (Triple Data Encryption Algorithm) and AES are specified in [10].
Hash (A): Digital signatures and hash-only applications.
Hash (B): HMAC, Key Derivation Functions and Random Number Generation.

Cryptography and computer security

Historical cryptography

When was cryptography "invented"? (1)

- Egypt, Mesopotamia
- Hebrews - The Bible (Old Testament) contains sections encrypted with the Hebrew cipher atbash (500 BC)
  - the first *alef* character is replaced by the last *taw*,
  - the second *bet* is exchanged for penultimate *shin*
  - https://en.wikipedia.org/wiki/Atbash
  - Greeks - 500 years BC Scytale



- The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography by Simon Singh

When was cryptography "invented"? (2)

- Caesar (100-44 BC)
  ```
  C = L FDPH L VDZ L FRQTXHUHG
  M = I CAME I SAW I CONQUERED
  ```
- Key = 3, shift in alphabet
- Shift in alphabet by 3 characters:
  ```
  M = ABCDEFGHIJKLMNOPQRSTUVWXYZ
  C = DEFGHIJKLMNOPQRSTUVWXYZABC
  ```

Basic cryptographic principles

- In the past, there were two basic design principles for encryption algorithms (symmetric, which used a shared secret key):
  - Substitution (substitution) - replacing characters of plaintext with other characters.
    - Monoalphabetic ciphers
    - Polyalphabetic ciphers
  - Transposition (permutation) - does not change the characters of the plaintext, but changes their order.
- Currently, pure substitution or permutation algorithms are not used.
  - Algorithms (symmetric) that combine both techniques are used, called product ciphers.

Substitution algorithms

- Substitution algorithms
  - Monoalphabetic substitution algorithms (simple substitution)
    - Each character of the plaintext is replaced in the ciphertext in the same way (by the same character), i.e. all occurrences are replaced in the same way, e.g. the character A will always be replaced by e.g. X
  - Polyalphabetic substitution algorithms
    - Multiple characters can be used for a plaintext character in a ciphertext, i.e. "multiple alphabets", multiple substitutions are used. E.g. one time a plaintext A will be replaced by a ciphertext X, another time by a ciphertext B, etc.
- https://en.wikipedia.org/wiki/Substitution_cipher

Monoalphabetic Substitution Algorithms (1)

- **Shift Cipher** (generalization of Caesar's cipher)
  - Alphabet (plaintext alphabet and ciphertext alphabet) – English alphabet (without space) 'A'=0, ..., 'Z'=25, number of characters n = 26
  - $M=C=K=Z_{26}$
    - Let Z be the infinite set of integers, then Z_{26} is the finite set of integers modulo 26, $Z_{26} = \{0, 1, ..., 25\}$
  - message m is encrypted by blocks (block length is 1 character)
  - The key k is n (=1) characters, $k \in \{0, 1, ..., 25\}$,
  - $c = e_k(m) = (m+k) \bmod 26$
  - $m = d_k(c) = (c-k) \bmod 26$
  - 26 possible keys (or 25, since a shift of 0 positions is meaningless)
  - The Caesar cipher is a Shift cipher with k=3
    - $c = e_k(m) = (m+3) \bmod 26$
    - $m = d_k(c) = (c-3) \bmod 26$

Monoalphabetic substitution (2)

- General substitution (algorithm) works with key obtained by any permutation of 26 characters of alphabet, e.g.

(PA) plaintext alphabet: `ABCDEFGHIJKLMNOPQRSTUVWXYZ`

(CA) ciphertext alphabet (=key): `DKVQFIBJWPESCXHTMYAUOLRGZN`

```
M = IFWEW ISHTO REPLA CELET TERS
C = WIRFR WAJUH YFTSD VFSFU UFYA
```

- The key is an arbitrary permutation of PA.
  - There are 26! keys > 4×10^{26} (key space)

Monoalphabetic substitution (3)

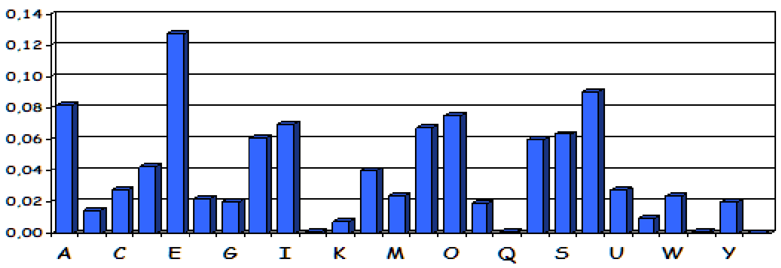
- Modification
  - the alphabet is determined by a key (password) and a mechanism for completing the remaining characters of the alphabet
  - Advantage - the entire permuted alphabet does not have to be passed, only the "password" to create it

```
K = JULIUSCAESAR
PA = ABCDEFGHIJKLMNOPQRSTUVWXYZ
CA = JULISCAERTVWXYZBDFGHKMNOPQ
```

Frequency analysis

- Known ciphertext only attack
  - There are different types of cryptanalytic attacks depending on what the attacker has at his disposal
- Described by Abu Al-Kindi in "Manuscript on Deciphering Cryptographic Message", 9th century AD.
- It is based on the analysis of the properties of natural language, on its statistical properties, on the frequencies of occurrence of individual characters, pairs (bigrams), triplets (trigrams) of characters, etc.
- Suitable for longer texts (and not suitable for specific texts, e.g. *"from Zanzibar to Zambia to Zaire, ozone zones make zebras run zany zigzags"*)
- https://en.wikipedia.org/wiki/Frequency_analysis

Some frequency tables



LET	COUNT	PERCENT	bar graph
E	445.2 B	12.49%	E
T	330.5 B	9.28%	T
A	286.5 B	8.04%	A
O	272.3 B	7.64%	O
I	269.7 B	7.57%	I
N	257.8 B	7.23%	N
S	232.1 B	6.51%	S
R	223.8 B	6.28%	R
H	180.1 B	5.05%	H
L	145.0 B	4.07%	L
D	136.0 B	3.82%	D
C	119.2 B	3.34%	C
U	97.3 B	2.73%	U
M	89.5 B	2.51%	M
F	85.6 B	2.40%	F
P	76.1 B	2.14%	P
G	66.6 B	1.87%	G
W	59.7 B	1.68%	W
Y	59.3 B	1.66%	Y
B	52.9 B	1.48%	B
V	37.5 B	1.05%	V
K	19.3 B	0.54%	K
X	8.4 B	0.23%	X
J	5.7 B	0.16%	J
Q	4.3 B	0.12%	Q
Z	3.2 B	0.09%	Z

th 3.56%	of 1.17%	io 0.83%
he 3.07%	ed 1.17%	le 0.83%
in 2.43%	is 1.13%	ve 0.83%
er 2.05%	it 1.12%	co 0.79%
an 1.99%	al 1.09%	me 0.79%
re 1.85%	ar 1.07%	de 0.76%
on 1.76%	st 1.05%	hi 0.76%
at 1.49%	to 1.05%	ri 0.73%
en 1.45%	nt 1.04%	ro 0.73%
nd 1.35%	ng 0.95%	ic 0.70%
ti 1.34%	se 0.93%	ne 0.69%
es 1.34%	ha 0.93%	ea 0.69%
or 1.28%	as 0.87%	ra 0.69%
te 1.20%	ou 0.87%	ce 0.65%

https://norvig.com/mayzner.html

Relative frequency (effect of corpus choice)

- English

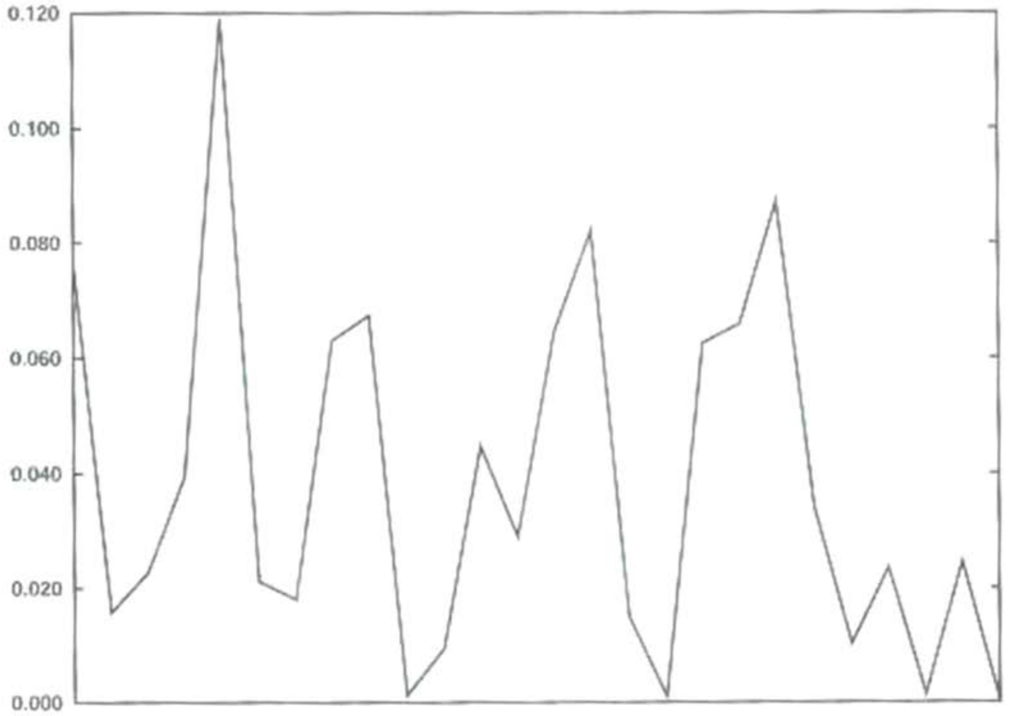


Figure 1-1 Frequency distribution table for Shakespeare's complete works [3]. The letters are shown left to right, *A* through *Z*, with the *y*-value being the frequency of that character occurring in *The Complete Works of William Shakespeare* [3].

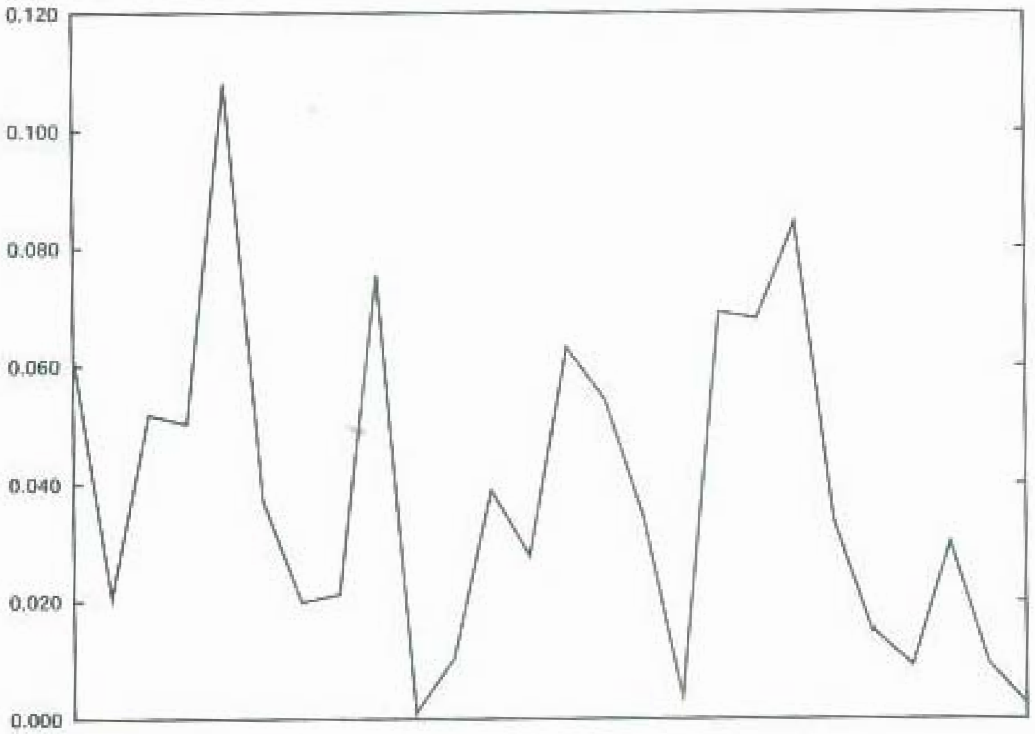


Figure 1-2 Frequency distribution table for "vanilla" Linux 2.6.15.1 source code (including only alphabetic characters). The total size is approximately 205 megabytes.

Monoalphabetic substitution - Polygram ciphers

- Polygram ciphers (polygram encryption algorithms**)**
  - the plaintext character is encrypted into the character group ciphertext
  - or character group → to character group
- Polybois Square (https://en.wikipedia.org/wiki/Polybius_square)

SQUARE →DCDADEAADBAE

	A	B	C	D	E
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

Monoalphabetic substitution - Polygram ciphers

- Bigram to bigram - Playfair, letters from each of the bigrams can appear in three positions in the square: on the same row, on the same column or on a different row and column:
    - If both letters of the bigram are on the same row, they are replaced by the letters to the right of them. If one of the letters is the last one in the row, it is replaced by the first one in the same row.
    - If both letters are in the same column, they are replaced by the letters below them. If one of the letters is the last in a column, it is replaced by the first in the same column.
    - If the two letters are in different rows and columns, each is replaced by the letter at the intersection of the row of that letter and the column of the other letter.
- See https://en.wikipedia.org/wiki/Playfair_cipher for an example.

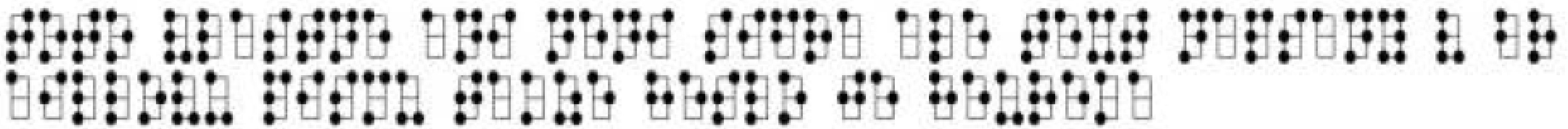
Monoalphabetic substitution - different alphabets

- There is no need to replace letters with letters again. We can use arbitrary characters, i.e. the **plain and crypto alphabet can be different**
- Ex: plaintext is in English without a space.
- NO:

```
5 3 ‡ ‡ † 3 0 5 ) ) 6 * ; 4 8 2 6 ) 4 ‡ . ) 4 ‡ ) ; 8 0 6 *
; 4 8 † 8 ¶ 6 0 ) ) 8 5 ; 1 ‡ ( ; : ‡ * 8 † 8 3 ( 8 8 ) 5 *
† ; 4 6 ( ; 8 8 * 9 6 * ? ; 8 ) * ‡ ( ; 4 8 5 ) ; 5 * † 2 :
* ‡ ( ; 4 9 5 6 * 2 ( 5 * - 4 ) 8 ¶ 8 * ; 4 0 6 9 2 8 5 ) ;
) 6 † 8 ) 4 ‡ ‡ ; 1 ( ‡ 9 ; 4 8 0 8 1 ; 8 : 8 ‡ 1 ; 4 8 † 8
5 ; 4 ) 4 8 5 † 5 2 8 8 0 6 * 8 1 ( ‡ 9 ; 4 8 ; ( 8 8 ; 4 (
‡ ? 3 4 ; 4 8 ) 4 ‡ ; 1 6 1 ; : 1 8 8 ; ‡ ? ;
```

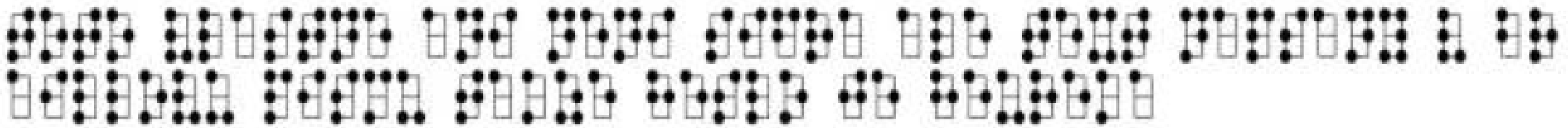
- The solution in the short story The Gold-Bug, Edgar Allan Poe, e.g. http://en.wikipedia.org/wiki/The_Gold-Bug

Monoalphabetic substitution - different alphabets

- 0 B1 2 345 ox6 r7Ma 890 juFy k1 L234 k enG5 I6ex e7 Gxx 8 9a s 01rZ h2 Gpk k x345 e 6y nkj7 k8 T9v dT a0 1G2v s 3e B4 56 789 I0h krT F 123 vuo4 5x6 xn7 c89n uy ZFs 01 s 2w F3 d456 uje 789 oM0 1 r 2L3s 4L5 6u78 G9 0T1h Gj deF v2Qv Qe 3d4h p 5n6M d7 y8y 9M0 1s23 j4c v wwe u 5x Zw e678 M Q901 jn2 3I j4 5x6
- 

Monoalphabetic substitution - different alphabets

- 0 B1 2 345 ox6 r7Ma 890 juFy k1 L234 k enG5 I6ex e7 Gxx 8 9a s 01rZ h2 Gpk k x345 e 6y nkj7 k8 T9v dT a0 1G2v s 3e B4 56 789 I0h krT F 123 vuo4 5x6 xn7 c89n uy ZFs 01 s 2w F3 d456 uje 789 oM0 1 r 2L3s 4L5 6u78 G9 0T1h Gj deF v2Qv Qe 3d4h p 5n6M d7 y8y 9M0 1s23 j4c v wwe u 5x Zw e678 M Q901 jn2 3I j4 5x6
    - The text was converted into ciphertext using morse code. The following encoding was used to write the morse code: dot = any letter, comma = any digit



- This is a coding where Braille blind coding is used to code the letter.

Polyalphabetic substitution

- Multiple substitutions are used for each letter of the plain alphabet, i.e. multiple cipher alphabets.
- Which cipher alphabet is used depends on the key.
- Better hides frequency dependencies, provides better frequency distribution of ciphertext characters.

The Vigenère cipher

- Blaise de Vigenère - The Vigenère cipher
  - le chiffre indéchiffrable
  - published in "*Traicté des Chiffres*" in 1585 a polyalphabetic cipher, which was not broken until 1854 (1863)
  - broken by Ch. Babbage and F. Kasiski respectively.
  - Cryptanalysis, known as the Kasiski test, is based on the assumption that the key is shorter than the plaintext and thus must be used repeatedly.

The Vigenère cipher

```
M = THISP ROCES SCANA LSOBE EXPRE SSED
K = CIPHE RCIPH ERCIP HERCI PHERC IPHE
C = VPXZT IQKTZ WTCVP SWFDM TETIG AHLH

  C -> CDEFGHIJKLMNOPQRSTUVWXYZAB
  I -> IJKLMNOPQRSTUVWXYZABCDEFGH
  P -> PQRSTUVWXYZABCDEFGHIJKLMNO
  H -> HIJKLMNOPQRSTUVWXYZABCDEFG
  E -> EFGHIJKLMNOPQRSTUVWXYZABCD
  R -> RSTUVWXYZABCDEFGHIJKLMNOPQ

'T' key 'C' maps to 'V'
'H' key 'I' maps to 'P'
'I' key 'P' maps to 'X' etc.
```

Vigenère table

```
    ABCDEFGHIJKLMNOPQRSTUVWXYZ

A | ABCDEFGHIJKLMNOPQRSTUVWXYZ |
B | BCDEFGHIJKLMNOPQRSTUVWXYZA |
C | CDEFGHIJKLMNOPQRSTUVWXYZAB |
D | DEFGHIJKLMNOPQRSTUVWXYZABC |
E | EFGHIJKLMNOPQRSTUVWXYZABCD |
F | FGHIJKLMNOPQRSTUVWXYZABCDE |
G | GHIJKLMNOPQRSTUVWXYZABCDEF |
H | HIJKLMNOPQRSTUVWXYZABCDEFG |
I | IJKLMNOPQRSTUVWXYZABCDEFGH |
J | JKLMNOPQRSTUVWXYZABCDEFGHI |
K | KLMNOPQRSTUVWXYZABCDEFGHIJ |
L | LMNOPQRSTUVWXYZABCDEFGHIJK |
M | MNOPQRSTUVWXYZABCDEFGHIJKL |
N | NOPQRSTUVWXYZABCDEFGHIJKLM |
O | OPQRSTUVWXYZABCDEFGHIJKLMN | ...
```

Polyalphabetic substitution

- $M=C=K=(Z\ )_{26}{}^{n}$, the message m is encrypted in blocks of n characters,
- The key is a string of n characters, $k=(k_1\ , k_2\ ,\ldots,k_n\ )$,
- $e_k\ (m_1\ ,\ m_2\ ,\ldots,m_n\ )=(m\ +k_1\ )\bmod 26,\ldots,(m\ +k_n\ )\ \bmod 26$,
- $d_k\ (c_1\ ,\ c_2\ ,\ldots,c_n\ )=(c\ -k_1\ )\bmod 26,\ldots,(c\ -k_n\ )\ \bmod 26$.
- The number of all different keys is 26^n ,

Plaintext:	`attackatdawn`
Key:	`LEMONLEMONLE`
Ciphertext:	`LXFOPVEFRNHR`

$c_1 = A + L \bmod 26,\ c_2 = T + E \bmod 26\ldots,$

i.e. $c_1 = 0 + 11 \bmod 26,\ c_2 = 19+ 4 \bmod 26\ldots,$

Autokey Cipher

- Vigenère proposed "the **autokey** cipher" (he wanted to find a way to create a key as long as the plaintext)
- Keyword DECEPTIVE

M = WEAREDISCOVEREDSAVEYOURSELF

K = DECEPTIVEWEAREDISCOVEREDSAV

C = ZICVTWQNGKZEIIGASXSTSLVVWLA

Hill's cipher

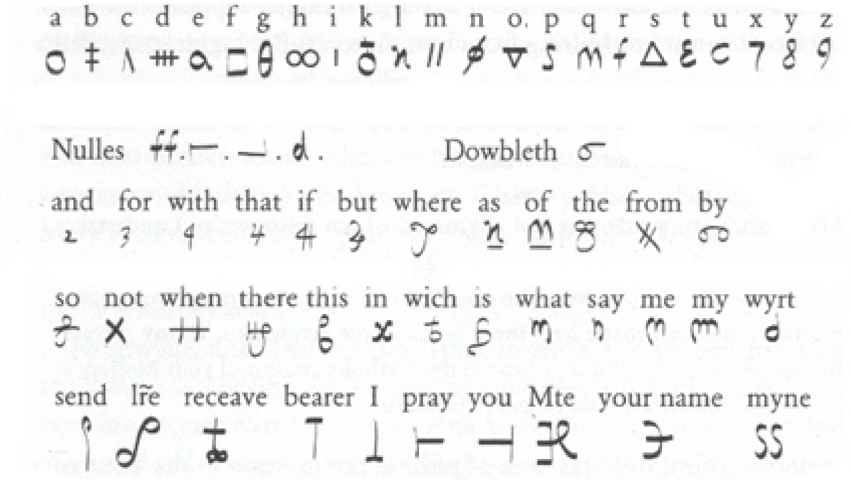
- Polygram cipher (Lester Hill 1929), counting with matrices
- $C = E_k(M) = K*M \bmod 26$
- $M = D_k(C) = K^{-1}*C \bmod 26 = K*K^{-1} *M = M$
- C and M are column vectors of length n (block size n),
- matrix K is the matrix of the key n*n, n represents the number of characters in the group
- $KK^{-1} \bmod 26 = I$, where I is the identity matrix
- Hides frequency dependencies (even of digrams, trigrams)
- https://en.wikipedia.org/wiki/Hill_cipher
- https://www.geeksforgeeks.org/hill-cipher/

Homophonic cipher

- A substitution cipher that assigns to each plaintext character one of a set of possible different ciphertext characters.
- The number of potential substitutes being proportional to the frequency of the letter.
  - For example, the letter 'a' accounts for roughly 8% of all letters in English, so we assign 8 symbols to represent it. Each time an 'a' appears in the plaintext it is replaced by one of the 8 symbols chosen at random, and so by the end of the encipherment each symbol constitutes roughly 1% of the ciphertext.
  - The letter 'b' accounts for 2% of all letters and so we assign 2 symbols to represent it. Each time 'b' appears in the plaintext either of the two symbols can be chosen, so each symbol will also constitute roughly 1% of the ciphertext. This process continues throughout the alphabet, until we get to 'z', which is so rare that is has only one substitute.....,
- after encryption, the frequency of each character will be approximately the same.
- https://www.simonsingh.net/The_Black_Chamber/homophonic_cipher.html

Nomenclatures

- Nomenclature
  - is an enhancement of the homophonic cipher - it adds additional code equivalents for the most commonly used words, syllables and names.
  - Nomenclatures gradually grew until they contained thousands of code names and words, providing the basis for the emergence of codebook encryption.
  - The disadvantage of nomenclatures - they were not frequently replaced (one person used his nomenclature for his whole life).
  - The advantage - simplicity and speed of use.



Code books

- Code - a special cryptographic system that works with linguistic (language) elements. These elements can be selected words, whole sentences or clauses. For example, the code of *an egg* can mean a grenade, ...
  - If the meaning of the codes is publicly known (e.g. radio Q-code, where QPA means "The passcode is ... . ", QTC "How many telegrams do you have to transmit ?"), it is a code in the classical sense.
  - https://en.wikipedia.org/wiki/Q_code
- If the meaning of the codes is kept secret, it is a special encryption system called a **codebook** (it was even the most used method of encryption during World War I & II).
  - https://en.wikipedia.org/wiki/Code_talker
- Zimmermann telegram (German diplomatic code system 13042)
  - https://en.wikipedia.org/wiki/Zimmermann_Telegram

Code books

- They are convenient for situations that are anticipated in their preparation, and therefore contain code equivalents only for selected plaintexts (but encryption algorithms are suitable for any situation, since they can convert any plaintext into ciphertext).
- Disadvantages
  - long and demanding preparation of quality code
  - the need to ensure perfect secrecy in the printing of these books
  - (costly) secure and fast distribution of codebooks to end users is required
  - the need to use one type of codebook for a long time (replacing it with a new one is expensive and complicated)
  - the "loss" of a single copy compromises the entire system and it is necessary to switch immediately to a new codebook