What we are going to learn

- Symmetric cryptography
  - Block cipher
  - Stream cipher
- key distribution

Aims of cryptography

- **message privacy**: ensuring that only the intended parties to the communication can read the message
- **message integrity**: ensuring that the message received is the same as the message sent

Some Terminology

- **Ciphers** are algorithms for obscuring the content of any given message
- **Cryptography** is the art of encrypting messages
- **Cryptanalysis** is the art of discovering the content of (cracking) encrypted messages

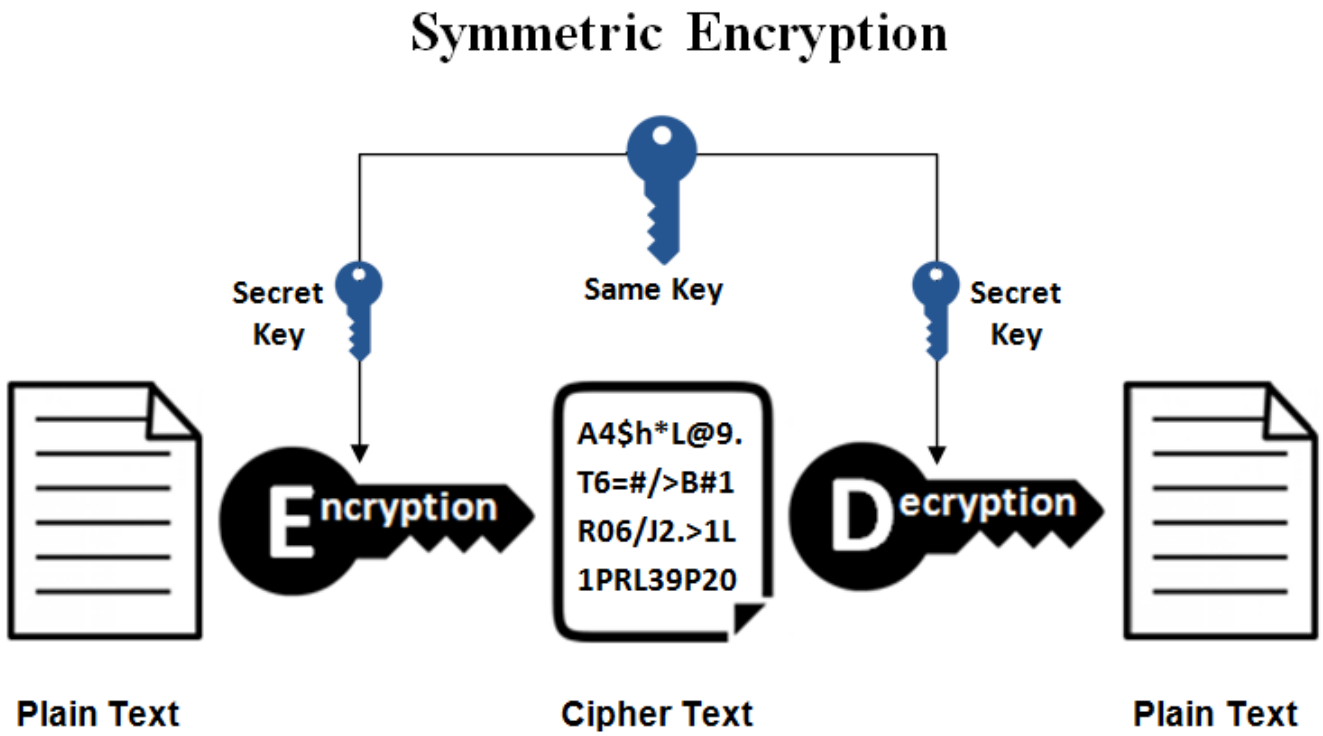
Kerchkhoffs's principle

Principle

A cryptosystem should be secure even if everything about the system, except the key, is public knowledge.

- Ciphers are public knowledge
- Keys may be secret

Symmetric Encryption



Symmetric Encryption

- Plaintext $M \xrightarrow{enc\ K} E(M, K)$
- $E(M, K) \xrightarrow{dec\ K} D(E(M, K), K) = M$

Julius Caesar's ciphers in Gallic Wars

- $E(M, K) =$ shift each of M forward by K places in the alphabet
- $D(M, K) =$ shift each of M backward by K places in the alphabet
- It is wise to omit spaces,

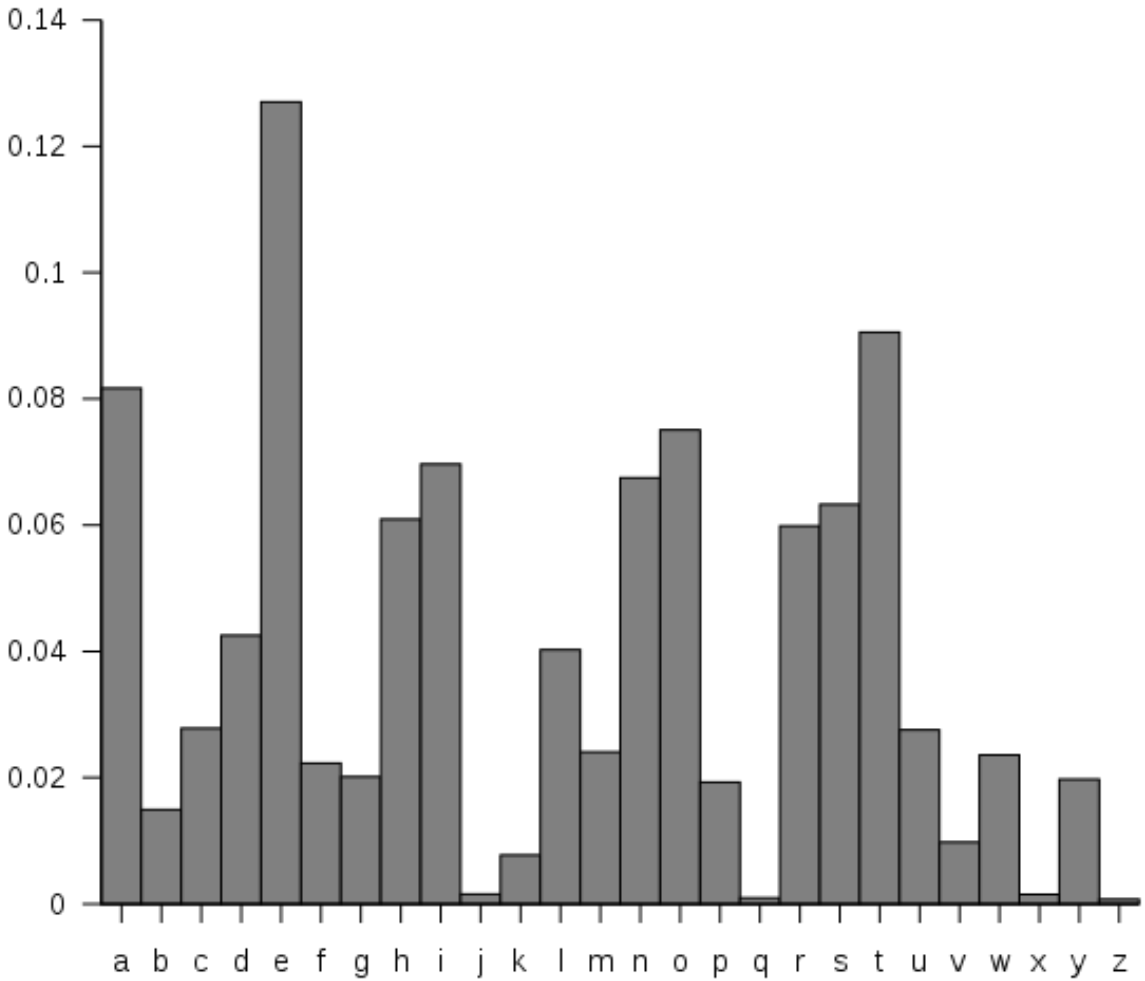
Julius Caesar's ciphers in Gallic Wars

- $E(M, K) =$ shift each of M forward by K places in the alphabet
- $D(M, K) =$ shift each of M backward by K places in the alphabet
- It is wise to omit spaces, as 1-letter words are likely to be 'a' and 'I', and 2-letter words 'an', 'at', 'it', 'he', 'do' etc
- Key space is $\{1, \ldots, 25\}$

Substitution Cipher

- a key is a bijective mapping from the alphabet to itself.
- what is the key space?

Statistical Cryptanalysis of Substitution Ciphers



courtesy of wikipedia

Statistical Cryptanalysis of Substitution Ciphers

- In sufficiently long texts letters tend to appear with predicable frequency
- Try partial substitution in which the most frequent letter of the cipher text corresponds to ‘E’, the next most frequent to ‘T’, etc.
- Some change of order of frequency may occur. Switch correspondence around and decrypt according to the partial guess until it starts to look like English words

Vigenère Poly-alphabetical Cipher

- Originally described by Giovan Battista Bellaso in his 1553 book, later misattributed to Blaise de Vigenre in the 19th century
- One of the commonly known stream cipher
- To get $E(M, K)$
  1. align K over M, repeated as many as necessary
  2. substitute each letter a in M, where b is the letter in K, by the letter in a's row and b's column

Vigenère Poly-alphabetical Cipher

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

Block Ciphers

- Divide the plaintext M into blocks of equal length $M = M_1 M_2 M_3 \ldots$
- A block cipher encrypts the blocks separately into $E(M, K) = E'(M_1, K)E'(M_2, K)E'(M_3, K)$

Block Cipher Modes of Operation

- Electronic Codebook (ECB) Mode:
  $E(M_1 M_2 M_3, K) = E(M_1, K)E(M_2, K)E(M_3, K)$

Block Cipher Modes of Operation

- Electronic Codebook (ECB) Mode:
  $E(M_1 M_2 M_3, K) = E(M_1, K)E(M_2, K)E(M_3, K)$

  If there is a repeated plaintext block, then there is a repeated ciphertext block, which gives a handle for cryptanalysis

Block Cipher Modes of Operation

- Electronic Codebook (ECB) Mode:
  $E(M_1 M_2 M_3, K) = E(M_1, K)E(M_2, K)E(M_3, K)$

  If there is a repeated plaintext block, then there is a repeated ciphertext block, which gives a handle for cryptanalysis

- Cipher Block Chaining (CBC):
  $E'(M_{i+1}, K) = E(E'(M_i, K) \otimes M_{i+1}, K)$, i.e., the input to the encryption algorithm is the XOR of the current plaintext M_{i+1} and the preceding ciphertext block; the same key is used for each block.

Data Encryption Standard

- Initiated in 1972 by a US standards bureau NSB (now called National Institute of Standards and Technology (NIST))
- Designed by researchers at IBM at the request of NSB, input from the National Security Agency (NSA)
- A block cipher, each block 64 bits in length, key is 56 bits in length
- Multiple rounds of permutation/shifting, substitution and transposition
- Adopted as a standard in 1976, to be reviewed every five years
- recertified in 'triple-DES' in 1999
- Extensively used in financial industry

triple-DES

- Currently recommended way of using DES
- 168-bit key length, effectively three keys
- $3\text{-}DES(M, K_1K_2K_3) = E_{K_3}(D_{K_2}(E_{K-1}(M)))$

Advanced Encryption Standard (AES)

- As a replacement to DES
- a symmetric block cipher with a block length of 128 bits
- support for key lengths of 128, 192, and 256 bits
- 5 finalists announced August 1999
- Rijndael algorithm adopted as AES in Nov 2001

Stream Ciphers

- processes the input elements continuously, producing output one element at a time
- Not as widely used as block ciphers, but faster
- Used with a pseudorandom number generator (key stream generator) which takes an input key and generates a key stream used for encryption/decryption

Quiz

Alice and Bob live thousands of miles apart, and they can only communicate via a special postal service. The post workers always deliver mail packages to their destination, however contents in a package may be stolen unless there is a lock put on the package. They have plenty of usable/empty packages, locks and keys but they don't have any of the same locks or keys. If Bob wants to send Alice an expensive ring through the mail, how can he achieve that?

Key Distribution

Alice and Bob wish to communicate over the Internet. Eve, an eavesdropper, is listening on the line. Therefore, Alice and Bob needs to establish a shared secret key.

Key Distribution

Alice and Bob wish to communicate over the Internet. Eve, an eavesdropper, is listening on the line. Therefore, Alice and Bob needs to establish a shared secret key.

A non-solution:

$$A \longrightarrow B : K$$

$$A \longrightarrow B : E_K(M)$$

If Eve knows K and $E_K(M)$, she can compute M.

Key Distribution

If a given cipher has the following property:

$$E_{K_A}(E_{K_B}(M)) = E_{K_B}(E_{K_A}(M))$$

We are able to use the protocol on the next slide.

Key Distribution

A generates a shared key K

$A \longrightarrow B : E_{K_A}(K)$

$B \longrightarrow A : E_{K_B}(E_{K_A}(K))$

A decrypts with K_A, by computing
$D_{K_A}(E_{K_B}(E_{K_A}(K))) = D_{K_A}(E_{K_A}(E_{K_B}(K))) = E_{K_B}(K)$

$A \longrightarrow B : E_{K_B}(K)$

B computes $D_{K_B}(E_{K_B}(K)) = K$.

Key Distribution

If a given cipher has the following property:

$$E_{K_A}(E_{K_B}(M)) = E_{K_B}(E_{K_A}(M))$$

However, very few good ciphers satisfy the above property

Modular Arithmetic

$x \equiv y \ mod \ n$ if there exists k such that $x = k \times n + y$

I.e., x and y have the same remainder if divided by n

For every integer x, there exists $m \in \{0, 1, 2, \ldots n-1\}$ such that $x \equiv m \ mod \ n$

Discrete Logarithm

- Let n be a large prime

  g is a generator modulo n, if
  $a \ mod \ n$, $a^2 \ mod \ n$, $\ldots$, $a^{n-1} \ mod \ n$
  enumerates all members in $\{1, 2, 3, \ldots n-1\}$
- Given n prime, a generator modulo n always exists
- Let n be 5, then 3 is a generator modulo n
  - $3 \ mod \ 5 = 3$, $3^2 \ mod \ 5 = 4$, $3^3 \ mod \ 5 = 2$, $3^4 \ mod \ 5 = 1$

Discrete Logarithm

- Let n be a large prime

  g is a generator modulo n, if
  $a \ mod \ n$, $a^2 \ mod \ n$, $\ldots$, $a^{n-1} \ mod \ n$
  enumerates all members in $\{1, 2, 3, \ldots n-1\}$
- Given n prime, a generator modulo n always exists
- Let n be 5, then 3 is a generator modulo n
  - $3 \ mod \ 5 = 3$, $3^2 \ mod \ 5 = 4$, $3^3 \ mod \ 5 = 2$, $3^4 \ mod \ 5 = 1$
- **Discrete Logarithm Problem**: From $g^i \ mod \ n$, compute i

  This is a hard problem

Diffie-Hellman key exchange

- A and B agree on a large prime q and generator a, presumably known by everyone else (including attackers)

  A generates random value X_A

  B generates random value X_B
- $A \longrightarrow B : a^{X_A} \ mod \ q$
- $B \longrightarrow A : a^{X_B} \ mod \ q$
- A computes $(a^{X_B} \ modq)^{X_A} \ mod \ q = a^{X_A X_B} \ mod \ q$

  B computes $(a^{X_A} \ modq)^{X_B} \ mod \ q = a^{X_A X_B} \ mod \ q$

Security of the protocol depends on the hardness of discrete logarithm — from $a^{X_A} \ mod \ q$, it is hard to compute X_A.