The mathematics of secrets

Édouard Rousseau

July 7, 2021

Mathematical Summer in Paris

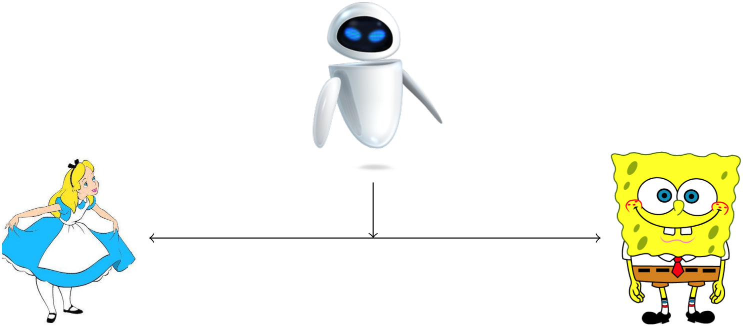
What is cryptography?

Crypto-what?

- **cryptography**/**cryptology** comes from ancient Greek
  - “crypto” means “secret”, “hidden”
  - “graphy” means “to write”
  - “logy” means “study”
- the science of “secret writing”
- the study of secret codes

What is the goal?

- two people (usually Alice and Bob) want to communicate
- a third person (Eve) can "hear" the communication



Goal: secure the communication

USES

- military communications (Caesar cipher, Enigma, ...)
- online payments
- secured websites
- encrypted chat apps (Whatsapp, Telegram, ...)
- ...

Figure: Enigma machine, used by Germany during WW2

Caesar's cipher

- used by Julius Caesar to write letters
- **idea:** shift all letters by a constant number k

Example: with $k = 3$, we have $\texttt{A} \to \texttt{D}$, $\texttt{B} \to \texttt{E}$, $\texttt{C} \to \texttt{F}$, $\ldots$, $\texttt{Z} \to \texttt{C}$

"Injustice anywhere is a threat to justice everywhere"

Caesar's cipher

- used by Julius Caesar to write letters
- **idea:** shift all letters by a constant number k

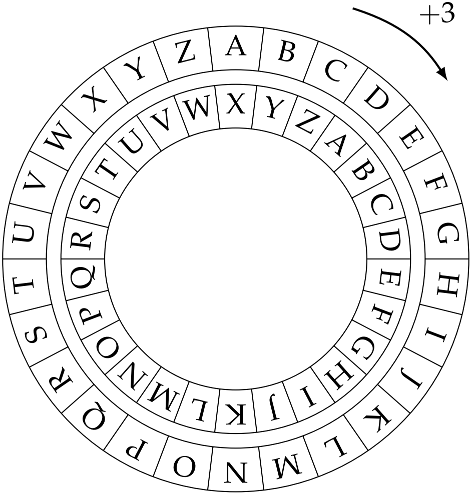
Example: with $k = 3$, we have $\mathtt{A} \to \mathtt{D}$, $\mathtt{B} \to \mathtt{E}$, $\mathtt{C} \to \mathtt{F}$, $\ldots$, $\mathtt{Z} \to \mathtt{C}$

"Injustice anywhere is a threat to justice everywhere"

`INJUSTICEANYWHEREISATHREATTOJUSTICEEVERYWHERE`

$\downarrow$

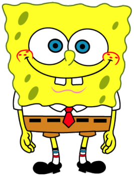
`LQMXVWLFHDQBZKHUHLVDWKUHDWWRMXVWLFHHYHUBZKHUH`


+3
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
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

SYMMETRIC CRYPTOGRAPHY

- In **symmetric** cryptography, participants share a **secret**/**key** prior to the communication

Secret

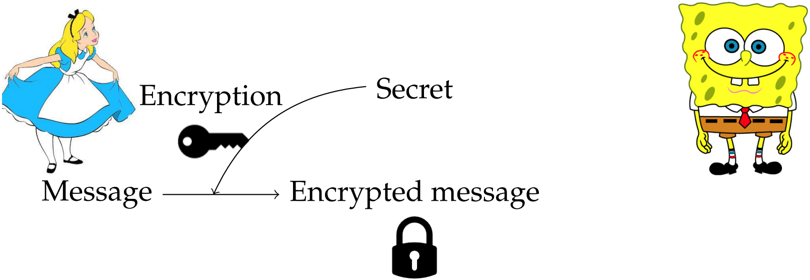


Message

- In Caesar's cipher, the key is the number k

SYMMETRIC CRYPTOGRAPHY

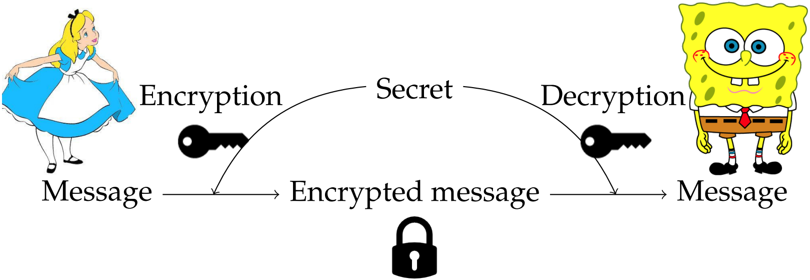
- In **symmetric** cryptography, participants share a **secret**/**key** prior to the communication



- In Caesar's cipher, the key is the number k

Symmetric cryptography

- In **symmetric** cryptography, participants share a **secret**/**key** prior to the communication



- In Caesar's cipher, the key is the number k

DECRYPTION

- If we know $k = 3$, decrypting the message is easy

LQMXVWLFHDQBZKHUHLVDWKUHDWWRMXVWLFHHYHUBZKHUH

$\downarrow$

INJUSTICEANYWHEREISATHREATTOJUSTICEEVERYWHERE

And whithout the key?

- without the key, it becomes harder
- we have to try all keys...

KYZJFEVJYFLCUEFKSVJFVRJP

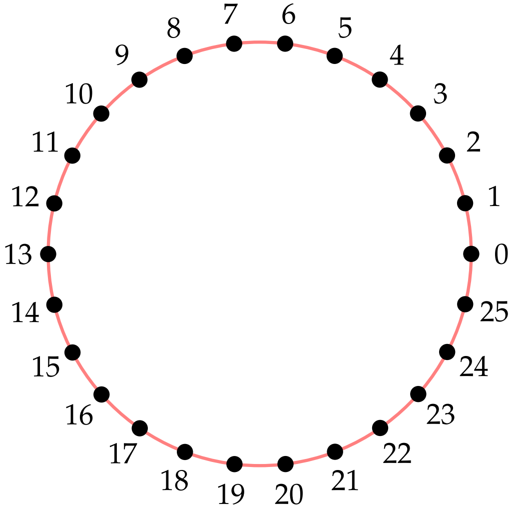
- $k = 1 \rightsquigarrow$ JXYIEDUIXEKBTDEJRUIEUQIO ☹
- $k = 2 \rightsquigarrow$ IWXHDCTHWDJASCDIQTHDTPHN ☹
- ...
- $k = 17 \rightsquigarrow$ THISONESHOULDNOTBESOEASY ☺

A bit of mathematics

Cryptography and mathematics

- **Mathematics** is a great tool to represent cryptosystems!
- Caesar's cipher: $\mathtt{A} = 0, \mathtt{B} = 1, \mathtt{C} = 2, \ldots, \mathtt{Z} = 25$
  - Encryption of the letter x is then represented by a simple addition $x + k$
  - Decryption of y is the substration $y - k$
- But shifting $\mathtt{Z} = 25$ by $k = 1$ should give $\mathtt{A} = 0$
  - $25 + 1 = 0$?
  - Yes! Use **modular integers**!

$\mathbb{Z}/26\mathbb{Z}$



- Let's define that structure properly!

Divisions in $\mathbb{Z}$

Definition (Divisibility)

Let $a, b \in \mathbb{Z}$ be two integers. We say that a **divides** b when there exists $c \in \mathbb{Z}$ such that

$$b = a \times c.$$

We also note $a \mid b$.

Example

- $3 \mid 12$ since $12 = 3 \times 4$
- $-2 \mid 6$ since $(-2)(-3) = 6$
- $42 \mid 0$ since $0 = 42 \times 0$

Euclidean division

Definition (Euclidean division)

Let $a, b \in \mathbb{Z}$ be two integers, with $b \geq 1$, then there exist unique integers $q \in \mathbb{Z}$ and $r \in \mathbb{N}$ such that

$$a = bq + r$$

and $0 \leq r < b$. The integer q is called the **quotient** of the euclidean division of a by b, while r is called the **remainder**. We also write $a = r \mod b$.

Example

- $17 = 5 \times 3 + 2$ thus $17 = 2 \mod 5$
- $39 = 14 \times 2 + 11$ thus $39 = 11 \mod 14$
- $18 = 3 \times 6 + 0$ thus $18 = 0 \mod 3$

Congruences

Definition (Congruence)

If $a, b \in \mathbb{Z}$ are integers, then a is said to be **congruent** to b **modulo** n if

$$a = b \mod n.$$

The integer n is called the **modulus** of the congruence.

Proposition

- $a = b \mod n \iff n \mid (a - b)$
- *if* $a = b \mod n$ *then* $b = a \mod n$
- *if* $a = b \mod n$ *and* $b = c \mod n$ *then* $a = c \mod n$
- *if* $a = c \mod n$ *and* $b = d \mod n$ *then* $a + b = c + d \mod n$
- *if* $a = c \mod n$ *and* $b = d \mod n$ *then* $a \times b = c \times d \mod n$

Definition of $\mathbb{Z}/n\mathbb{Z}$

Definition (Equivalence class)

The **equivalence class** of an integer $a \in \mathbb{Z}$ is the set of all integers congruent to a modulo n.

Definition (Integers modulo n)

The **integers modulo n**, denoted $\mathbb{Z}/n\mathbb{Z}$ (sometimes also $\mathbb{Z}_n$) is the set of (equivalence classes of) integers $\{0, 1, \ldots, n-1\}$. Addition, substraction and multiplication in $\mathbb{Z}/n\mathbb{Z}$ are performed modulo n.

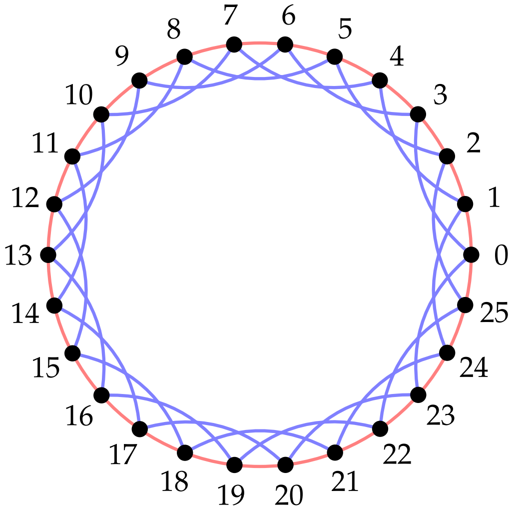
Example

- In $\mathbb{Z}/3\mathbb{Z}$, we have $2+2=1$, because $2+2=4=1 \mod 3$.
- In $\mathbb{Z}/6\mathbb{Z}$, we have $2 \times 3 = 0$, because $6 = 0 \mod 6$.

Representing Caesar's cipher

- We represent the letters of our message by elements in $\mathbb{Z}/26\mathbb{Z}$
- Encryption is only addition by k modulo 26
- Decryption is substration by k modulo 26

CAESAR'S CIPHER IN $\mathbb{Z}/26\mathbb{Z}$ WITH $k = 3$



Invertible elements in $\mathbb{Z}/n\mathbb{Z}$

Definition (Invertible elements)

An element x in $\mathbb{Z}/n\mathbb{Z}$ is called **invertible** when there exists an element $y \in \mathbb{Z}/n\mathbb{Z}$ such that

$$x \times y = 1.$$

The element y is called the **inverse** of x and is denoted by x^{-1}.
The **set of invertible elements** of $\mathbb{Z}/n\mathbb{Z}$ is denoted $(\mathbb{Z}/n\mathbb{Z})^{\times}$.

Example

- In $\mathbb{Z}/10\mathbb{Z}$, we have $3 \times 7 = 21 = 1$ thus 3 and 7 are invertible, and $3^{-1} = 7$.
- We have $(\mathbb{Z}/10\mathbb{Z})^{\times} = \{1, 3, 7, 9\}$.

Algebraic structure of $(\mathbb{Z}/n\mathbb{Z})^\times$

Definition

The set $(\mathbb{Z}/n\mathbb{Z})^\times$ is called a **cyclic group** if it is generated by the powers of one element $g \in (\mathbb{Z}/n\mathbb{Z})^\times$. The element g is called a **generator**.

Example

- Remember $(\mathbb{Z}/10\mathbb{Z})^\times = \{1, 3, 7, 9\}$
- $3^0 = 1$
- $3^1 = 3$
- $3^2 = 9$
- $3^3 = 27 = 7$
- $3^4 = 81 = 1$
- $\ldots$

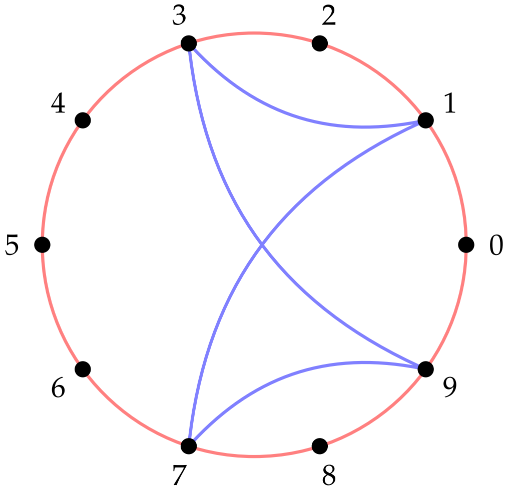


Figure: The set $\mathbb{Z}/10\mathbb{Z}$ and its invertible elements forming a cyclic group.

Some important results

Proposition

Let $x \in \mathbb{Z}/n\mathbb{Z}$*. Then* x *is invertible (i.e.* $x \in (\mathbb{Z}/n\mathbb{Z})^\times$*) if and only if* $\gcd(x, n) = 1$*.*

Example

- $3 \in (\mathbb{Z}/10\mathbb{Z})^\times$
- $7 \in (\mathbb{Z}/32\mathbb{Z})^\times$
- $4 \in (\mathbb{Z}/15\mathbb{Z})^\times$

Theorem

The set of invertible elements $(\mathbb{Z}/n\mathbb{Z})^\times$ *is a cyclic group if and only if* n *is* $1, 2, 4, p^k$ *or* $2p^k$ *with* p *an odd prime and* $k > 0$*.*

More problems in cryptography

Symmetric cryptography

- Caesar's cipher belongs to **symmetric cryptography**
  - Alice and Bob both know the **secret key**
  - This key allows both to **encrypt and decrypt**

Problem: Alice and Bob have to exchange their key **before** the communication

- How can they make the exchange **secure**?
  - They could meet in person $\rightsquigarrow$ not practical
  - Or use symmetric cryptography $\rightsquigarrow$ need a key again

Diffie and Hellman

- In 1976, Whitfield Diffie and Martin Hellman published a (now famous) article "New Directions in Cryptography"

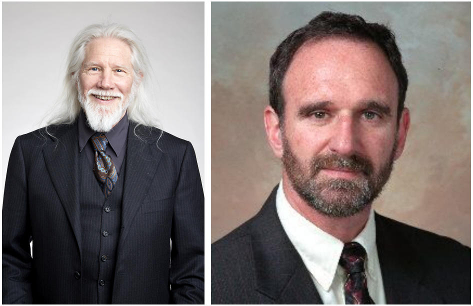


Figure: Whitfield Diffie (left) and Martin Hellman (right)

- They proposed a solution for managing **key exchange**!

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^{\times}$

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \leq a \leq p-2$
- Bob chooses a number b with $2 \leq b \leq p-2$

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^{\times}$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \leq a \leq p-2$
- Bob chooses a number b with $2 \leq b \leq p-2$

Then

- Alice computes $A = g^a$ and sends it to Bob

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \leq a \leq p-2$
- Bob chooses a number b with $2 \leq b \leq p-2$

Then

- Alice computes $A = g^a$ and sends it to Bob
- Bob computes $B = g^b$ and sends it to Alice

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \leq a \leq p-2$
- Bob chooses a number b with $2 \leq b \leq p-2$

Then

- Alice computes $A = g^a$ and sends it to Bob
- Bob computes $B = g^b$ and sends it to Alice
- Alice computes $B^a = \left(g^b\right)^a = g^{ab}$

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \le a \le p-2$
- Bob chooses a number b with $2 \le b \le p-2$

Then

- Alice computes $A = g^a$ and sends it to Bob
- Bob computes $B = g^b$ and sends it to Alice
- Alice computes $B^a = \left(g^b\right)^a = g^{ab}$
- Bob computes $A^b = (g^a)^b = g^{ab}$

Diffie-Hellman key exchange

Alice and Bob agree on this **public information**:

- a prime number p
- a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$

Then they choose integers that they keep **secret**

- Alice chooses a number a with $2 \leq a \leq p-2$
- Bob chooses a number b with $2 \leq b \leq p-2$

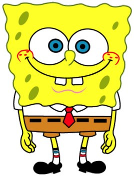
Then

- Alice computes $A = g^a$ and sends it to Bob
- Bob computes $B = g^b$ and sends it to Alice
- Alice computes $B^a = \left(g^b\right)^a = g^{ab}$
- Bob computes $A^b = (g^a)^b = g^{ab}$

Now they share the secret key g^{ab}!

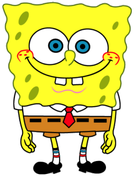
Small example

Public information: $p = 23$ and $g = 5$



Small example

Public information: $p = 23$ and $g = 5$



Chooses $a = 4$

Chooses $b = 3$

SMALL EXAMPLE

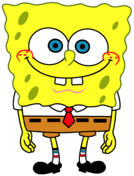
Public information: $p = 23$ and $g = 5$

Computes and sends $A = g^a = 4$

$\longrightarrow$

$\longleftarrow$

Computes and sends $B = g^b = 10$



Chooses $a = 4$

Chooses $b = 3$

Small example

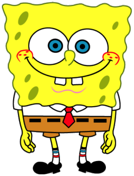
Public information: $p = 23$ and $g = 5$

Computes and sends $A = g^a = 4$

$\longrightarrow$

$\longleftarrow$

Computes and sends $B = g^b = 10$



Chooses $a = 4$

Chooses $b = 3$

Computes $s = B^a = 10^4 = 18$

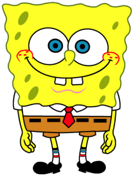
Computes $s = A^b = 4^3 = 18$

Small example

Public information: $p = 23$ and $g = 5$

Computes and sends $A = g^a = 4$ $\longrightarrow$

$\longleftarrow$ Computes and sends $B = g^b = 10$



Chooses $a = 4$

Chooses $b = 3$

Computes $s = B^a = 10^4 = 18$

Computes $s = A^b = 4^3 = 18$

They now share the secret $s = 18$!

IS IS SECURE?

- In the Diffie-Hellman key exchange protocol, an adversary knows g, g^a and g^b and wants to know $s = g^{ab}$.
- If the adversary knows one the secret value a or b, he can recover s

Discrete logarithm problem: it is **hard** to recover x from the data of g and g^x.

- We do not know any efficient technique (except in particular rare cases) to solve the discrete logarithm problem
- There is still research on the discrete logarithm problem!

Still, can't we guess?

- For $p = 23$, we can find x from g^x by computing all the powers of g by hand
- But for $p = 1031$, $g = 615$, and $g^x = 599$, would you do it?
  - A computer finds x in a few ms
- For $p = 1048583$, a computer finds x in 0.5 s
- For $p = 1073741827$, it takes 9 minutes
- For p very big, even a supercomputer cannot find x in a reasonable time

Conclusion on Diffie-Hellman

- In order to use **symmetric cryptography**, Alice and Bob need a common secret key
- Diffie-Hellman protocol allows them to **exchange** a key on a **public communication channel**
- Anyone can listen to the information they exchange, but nobody can recover the key

Asymmetric cryptography

Asymmetric cryptography

- Diffie and Hellman introduced a brilliant idea that enables the use of **symmetric cryptography**
  - In the Diffie-Hellman protocol, Alice and Bob still have symmetric roles
- In fact, it is possible to encrypt messages **without the need to exchange a secret key**!
  - Thanks to **asymmetric cryptography**!

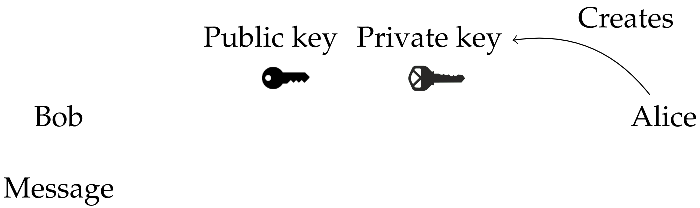
PUBLIC-KEY ENCRYPTION

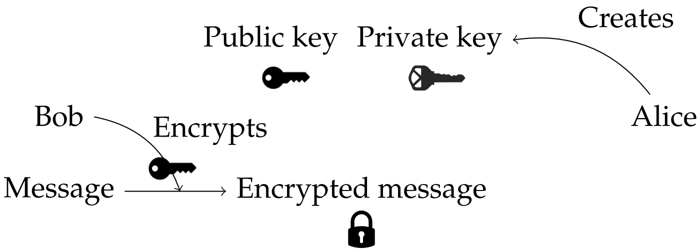
- In **asymmetric cryptography**, the roles of Alice and Bob are not the same anymore
- There are **two kinds of keys**
  - **public keys** used to **encrypt**
  - **private keys** used to **decrypt**
- Alice creates a **pair** of keys: a private one and a public one
- If Bob wants to send a message to Alice, he can use her public key to encrypt his message
- Alice can then use her private key to decrypt the message

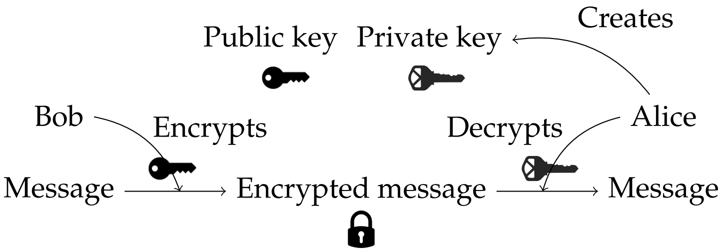
Bob

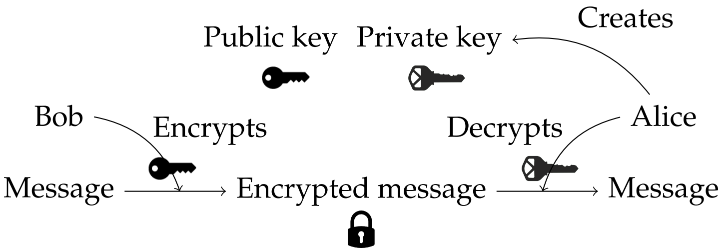
Alice

Message


Creates
Public key
Private key
Bob
Alice
Message


Creates
Public key
Private key
Bob
Encrypts
Alice
Message
Encrypted message


Creates
Public key
Private key
Bob
Encrypts
Decrypts
Alice
Message
Encrypted message
Message



- Before presenting an asymmetric encryption cryptosystem, we need a little bit of mathematics again.

Definition (Euler's totient function)

We let $\varphi(n)$ be the number of positive integers up to n that are coprime with n, *i.e.* the numbers x such that $\gcd(x, n) = 1$. The function φ is called **Euler's totient function**.

Example

- $\varphi(10) = 4$
  - $\gcd(1, 10) = \gcd(3, 10) = \gcd(7, 10) = \gcd(9, 10) = 1$
  - $\gcd(2, 10) = \gcd(4, 10) = \gcd(6, 10) = \gcd(8, 10) = 2$
  - $\gcd(5, 10) = 5$
  - $\gcd(0, 10) = \gcd(10, 10) = 10$
- $\varphi(7) = 6$
- $\varphi(35) = 24$

Properties of φ

Proposition

Let $n \geq 2$ be an integer. Then the set $(\mathbb{Z}/n\mathbb{Z})^\times$ has exactly $\varphi(n)$ elements.

Proof.

We said that $x \in (\mathbb{Z}/n\mathbb{Z})^\times \iff \gcd(x, n) = 1$ and the function φ counts the number of elements x such that $\gcd(x, n) = 1$. □

Proposition

The function φ is multiplicative: i.e. if $\gcd(x, y) = 1$, then $\varphi(xy) = \varphi(x)\varphi(y)$.

Example

- $\varphi(35) = \varphi(5 \times 7) = \varphi(5)\varphi(7) = 4 \times 6 = 24$

Fermat and Euler

Theorem (Fermat's little theorem)

If p is a prime number and x is a positive integer that is coprime with p (i.e. $\gcd(a, p) = 1$), then we have

$$x^{p-1} = 1 \mod p.$$

Theorem (Euler's theorem)

If x and n are coprime positive integers, then we have

$$x^{\varphi(n)} = 1 \mod n.$$

- Euler's theorem is a **direct generalization** of Fermat's little theorem, because $\varphi(p) = p - 1$

Rivest, Shamir and Adleman

- In 1977, Ron Rivest, Adi Shamir and Leonard Adleman were the first to describe an asymmetric encryption cryptosystem called **RSA**.

Figure: Ron Rivest (left), Adi Shamir, and Leonard Adleman (right)

The RSA cryptosystem

- Still **widely used today**
- works in $\mathbb{Z}/n\mathbb{Z}$, with $n = pq$ product of two primes
  - $\varphi(n) = (p-1)(q-1)$
- Alice chooses an integer $e \in \mathbb{N}$ such that $\gcd(e, \varphi(n)) = 1$
  - The pair (e, n) is her public key
- Alice computes an integer $d \in \mathbb{N}$ such that $e \times d = 1 \mod \varphi(n)$
  - The pair $(d, \varphi(n))$ is her private key

Encryption and decryption

- **Encryption** of a message x is done via

$$E(x) = x^e \mod n$$

- **Decryption** of a encrypted message y is done via

$$D(y) = y^d \mod n$$

Proposition

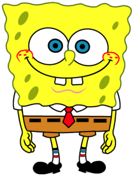
The RSA cryptosystem works: for any message $x \in \mathbb{Z}/n\mathbb{Z}$, we have $D(E(x)) = (x^e)^d = x^{ed} = x \mod n$.

Proof.

By Euler's theorem, we have $x^{\varphi(n)} = 1 \mod n$. We also have $ed = 1 \mod \varphi(n)$, hence there exists $k \in \mathbb{Z}$ with $ed = 1 + k\varphi(n)$. Thus $x^{ed} = x^{1+k\varphi(n)} = x \mod n$. $\square$

RSA TOY EXAMPLE

Wants to send $x = 7$



RSA TOY EXAMPLE

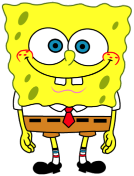
Public key: $(3, 15)$

Chooses $(p, q) = (3, 5)$

Computes $\varphi(n) = 8$

Chooses $e = 3$

Wants to send $x = 7$



Computes $d = 3$

RSA TOY EXAMPLE

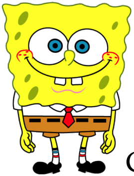
Public key: $(3, 15)$

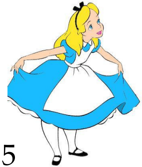
Chooses $(p, q) = (3, 5)$

Computes $\varphi(n) = 8$

Chooses $e = 3$

Wants to send $x = 7$





Computes and sends $7^3 = 13 \mod 15$

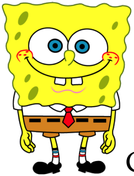
Computes $d = 3$

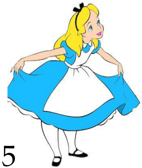
RSA TOY EXAMPLE

Public key: $(3, 15)$

Chooses $(p, q) = (3, 5)$
Computes $\varphi(n) = 8$
Chooses $e = 3$

Wants to send $x = 7$



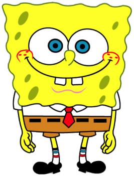


Computes and sends $7^3 = 13 \mod 15$

Computes $d = 3$
Computes $13^3 = 7 \mod 15$

RSA TOY EXAMPLE II

Wants to send $x = 42$



RSA TOY EXAMPLE II

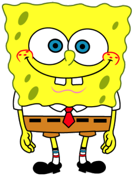
Public key: $(13, 77)$

Chooses $(p, q) = (7, 11)$

Computes $\varphi(n) = 60$

Chooses $e = 13$

Wants to send $x = 42$



Computes $d = 37$

RSA TOY EXAMPLE II

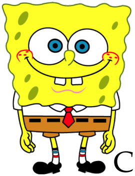
Public key: $(13, 77)$

Chooses $(p, q) = (7, 11)$

Computes $\varphi(n) = 60$

Chooses $e = 13$

Wants to send $x = 42$



Computes and sends $42^{13} = 14 \mod 77$

Computes $d = 37$

RSA TOY EXAMPLE II

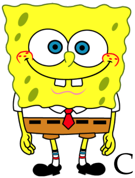
Public key: $(13, 77)$

Chooses $(p, q) = (7, 11)$

Computes $\varphi(n) = 60$

Chooses $e = 13$

Wants to send $x = 42$



Computes and sends $42^{13} = 14 \mod 77$

Computes $d = 37$

Computes $14^{37} = 42 \mod 77$

RSA SECURITY

- The best technique that we know to break the RSA cryptosystem is to find the factorization

$$n = pq$$

- Knowing p and q, we can recover $\varphi(n)$
- With $\varphi(n)$, we can recover d
- Factorization is believed to be **really hard** in practice, for large p and q

Open questions:

- We do not know if factorization is the best way to break the RSA cryptosystem
- We do not know if factorization is hard

One-way functions

Definition

A function $f : X \to Y$ is called **one-way** if $f(x)$ is easy to compute for all $x \in X$, but for essentially all elements $y \in Y$, it is hard to find any $x \in X$ such that $f(x) = y$.

Example

Let us take the function $f : \mathbb{Z}/17\mathbb{Z} \to \mathbb{Z}/17\mathbb{Z}$ such that $f(x) = 3^x$.

x	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$f(x)$	3	9	10	13	5	15	11	16	14	8	7	4	12	2	6	1

Trapdoor one-way functions

Definition

A **trapdoor one-way** function is a one-way function $f : X \to Y$ with the additional property that given some extra information (called **trapdoor information**) it becomes easy to find a preimage for any given $y \in Y$, *i.e.* to find $x \in X$ with $f(x) = y$.

Example

If $n = pq$ and e is an integer such that $\gcd(e, \varphi(n)) = 1$ then the map $f : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ given by $f(x) = x^e$ is a trapdoor one-way function! The trapdoor information is the factorization of n.

Conclusion

- One-way functions and trapdoor one-way functions are the basis for asymmetric cryptography
- It is unknown if there are "truly" one-way functions
  - often we cannot prove that a problem is "difficult"
  - the **discrete logarithm problem** and the **factorization problem** are two examples of such difficult problems
- There are still a lot of **open questions** in (asymmetric) cryptography
- Modular arithmetic ($\mathbb{Z}/n\mathbb{Z}$) gives challenging problems in cryptography, while being simple

There is more to discover!

- There are many other applications of cryptography
  - digital signatures
  - authentication
  - homomorphic encryption
  - ...
- based on many other mathematical objects
  - finite fields
  - elliptic curves
  - isogenies
  - lattices
  - systems of multivariate equations
  - ...