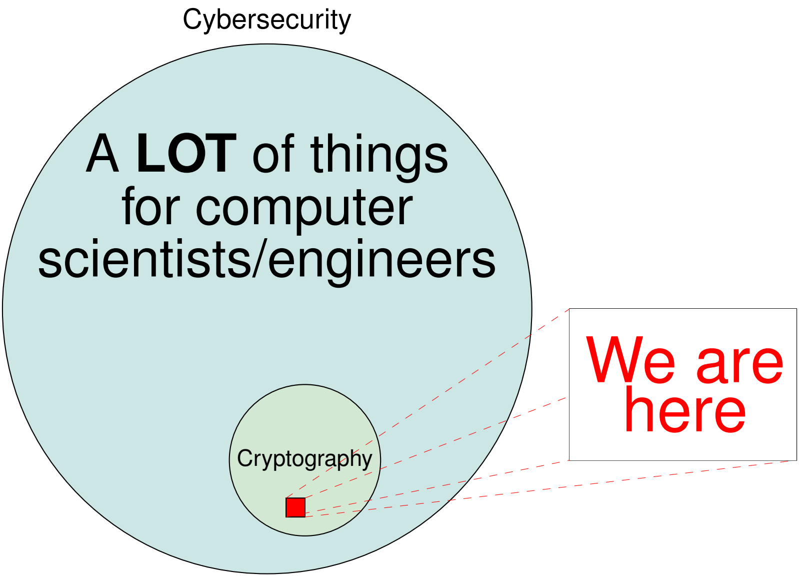
Cryptography on Magma

Marta Messia

31/03/2026


Cybersecurity
A **LOT** of things
for computer
scientists/engineers
Cryptography
We are
here

Properties of cryptographic algorithms

Functional requirements

1. Correctness
2. Termination
3. (Determinism)

Security requirements

An **hard** mathematical problem!

1. Confidentiality
2. Integrity
3. Authenticity

Resistance to chosen-ciphertext/chosen-plaintext/timing/side-channel attacks, etc...

What can we check with Magma?

Functional requirements

1. Correctness
2. Termination
3. (Determinism)

Security requirements

An **hard** mathematical problem!

1. Confidentiality
2. Integrity
3. Authenticity

Resistance to chosen-ciphertext/chosen-plaintext/timing/side-channel attacks, etc...

A simple encryption scheme - RSA

Alice

1. Chooses two big primes (in general $\sim 2^{1024}$) and computes $n = pq$, $\phi(n) = (p-1)(q-1)$.
2. Chooses $e \in \{1, \ldots, n\}$ such that $\gcd(e, \phi(n)) = 1$.
3. Computes $d \in \{1, \ldots, n\}$ such that $d \cdot e \equiv 1 \pmod{\phi(n)}$.

$$\xrightarrow{(n,e)}$$
$$\xleftarrow[c]{}$$

Bob

1. Chooses a message $m \in \{1, \ldots, n-1\}$.
2. Computes $c \equiv m^e \pmod{n}$.

Alice

Decodes the message c that she received as $c^d \equiv m^{d \cdot e} \equiv 1 \pmod{\phi(n)}$.

The **hard** problem

Factorization

`Cryptography` $\qquad$ `Number theory`

$\downarrow$ $\qquad$ $\downarrow$

Cryptosystems that are difficult to break. $\Longleftarrow$ Computationally **hard** problems.

Less talking!

HANDS ON!

$\downarrow$

With some number theory

The Dixon's random squares method

Ingredients

1. The odd integer n to factorize, must not be a prime power.
2. The smoothness factor v, an integer. *To be discussed later...*

Idea

Finding a series of congruences of type

$$x^2 \equiv y^2 \pmod{n}$$

such that at least one of those gives

$$\gcd(x+y, n) \notin \{1, n\} \quad \text{or} \quad \gcd(x-y, n) \notin \{1, n\}.$$

Then we got a proper factor of n.

Pseudocode

Algorithm Dixon(n, b)

Input: n: odd integer, not a prime power, to factorize.
v: an integer, smoothness factor.
Output: l: a proper factor of n.

Inizialization: Set $P \leftarrow \{p_1, \ldots, p_h\}$, the primes $p_i \leq v$. Initialize two empty lists B, Z.

1: Randomly choose $z \in \{1, \ldots, n\}$ and compute w the least positive remainder of $z^2 \pmod{n}$.
2: Factor $w = w' \prod_i p_i^{a_i}$, where w' has no factors in P. If $w' = 1$, then go to Step 3; else go to Step 1.
3: Add $(a_1, \ldots, a_h)$ to B and z to Z. If $\#B > h$, then go to Step 4; else go to Step 1.
4: Find the coefficients $f_b \in \{0, 1\}$ such that $\sum_b f_b b \equiv 0 \pmod{2}$, set $d \leftarrow \frac{1}{2}(\sum_b f_b b)$ and go to Step 5.
5: Let $x \leftarrow \prod_b z_b^{f_b}$ and $y \leftarrow \prod_i p_i^{d_i}$. If $x \equiv \pm y \pmod{n}$, then go to Step 1; else return $\gcd(x + y, n)$ or $\gcd(x - y, n)$.