Cryptography — Part 2

RSA & Elliptic Curve Cryptosystems

04.14.2026

Autumn Luzovich (they/them/theirs)

Quick Preface (again)

Cryptography is hard!

RSA

Bob: Choose 2 large primes p & q *(secret)*

- Shares $N = pq$ & e where $\gcd(\phi(N), e) = 1$

Alice: Choose message m

- Shares $C \equiv m^e \bmod N$

Bob: Compute $d \equiv e^{-1} \bmod \phi(N)$ then finds original message: $m' \equiv C^d \bmod N$

Integrity

Based on how easy it is to find p & q.

Elliptic Curve Cryptography

The build-up

Groups

- Sets closed under some binary operation

Rings

- Non-empty set $\mathcal{R}$ that is closed under 2 binary operations

Fields

- A commutative ring where every non-zero element has an inverse.

Elliptic Curve Cryptography (ii)

While any field can theoretically be retrofitted to be a cryptosystem, in modern cryptography we use elliptic curves.

Most elliptic curve use Short Weierstrass Form:

$$y^2 = x^3 + ax + b = x(x-1)(x-a)$$

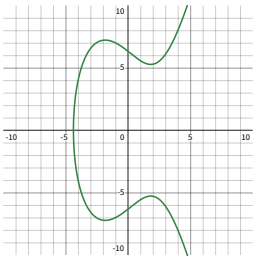

Figure 1: Elliptic curve in Short Weierstrass Form

Elliptic Curve Cryptography (iii)

In short, we perform binary operations on points on these curves to establish cryptographic integrity.

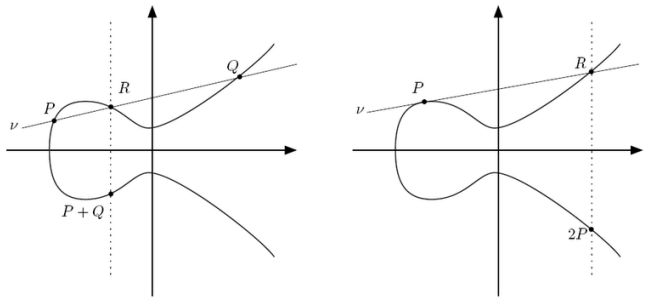

Figure 2: Elliptic curve operations defined over the real number field $\mathbb{R}$

But wait!
How are these resistant to modern computation?

Factorization

Factorization

In order to understand RSA's resistance, we must look at the two most common factorization algorithms: **Difference of Squares** and **Pollard's $p-1$ Method**.

Difference of Squares (Fermat Factorization)

Iterate through a loop of x, starting from 0 and incrementing by 1 to N such that the following holds:

$$N + x^2 = y^2$$

When we find this system we can split it apart into factors:

$$(y - x)(y + x)$$

And we're done!

Area of Efficiency

When done on composite numbers where their factors are close together; size does not matter.

Pollard's $p - 1$ Method

Using large composite numbers, find mutual divisors quickly:

$$\gcd(a^{r!}, N) > 1$$

- Typically for Pollard's it's common to start at $a = 2$.
- We keep increasing r by 1 (starting at 1 to N) until we find a gcd greater than 1 with N.
- If we get to N in the first iteration, we increase a by 1 and start over with $r = 1$.

Area of Efficiency

When done on numbers where their factors are smaller; distance apart does not matter.

The Crux

The Crux

So, to generate secure square-free composite integers, we must choose factors that are *not only* **big** but also **far apart**.

That way, they are resistant to both the Difference of Squares and Pollard's $p - 1$.

The Crux (ii)

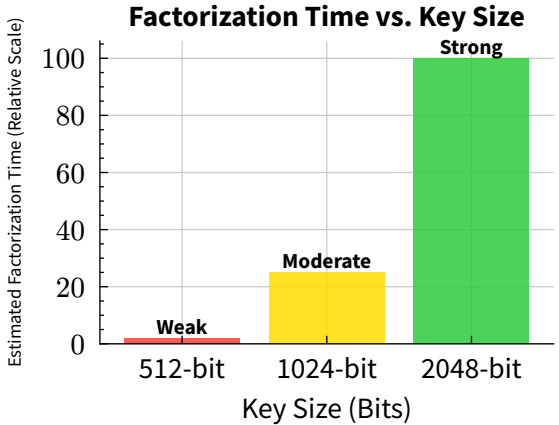

Figure 3: Factorization Time vs. Key Size (Mamatha and Sunitha)

Discrete Log Problem (for Elliptic Curves)

Given a large prime p & $g, h \in \mathbb{Z}/p\mathbb{Z}$, solving $g^x \equiv h \bmod p$ is hard.

We can expect similar exponential increases in integrity as key sizes increase.

Credits

[1] Policy-Based Cryptography: Theory and Applications - Scientific Figure on ResearchGate.

- https://www.researchgate.net/figure/Elliptic-curve-operations-defined-over-the-real-number-field-R_fig3_281013934 [accessed 13 Apr 2026]

[2] Mamatha and Sunitha. *Exploring the properties of prime numbers in cryptography*.

- https://doi.org/10.30574/wjarr.2022.13.1.0078