From RSA to Ed25519: The Evolution of Digital Locks

Understanding Modern Public-Key Cryptography through Pictures and Intuition

—

1. The Big Picture: Why RSA Was Revolutionary

RSA (1977) was the first practical public-key cryptosystem. It allowed two people to communicate securely *without ever meeting to share a secret key.*

Core idea:

$$\text{Encryption: } C = M^e \bmod n, \qquad \text{Decryption: } M = C^d \bmod n.$$

where $n = p \times q$ for two large primes.

It's elegant and reliable — but it has one big weakness: **to stay secure, the numbers must be huge.** A modern RSA key is often 2048 or 4096 bits long!

—

2. The New Generation: Elliptic-Curve Cryptography (ECC)

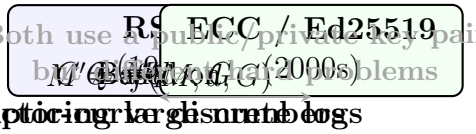
Elliptic-curve systems (like Ed25519, Curve25519, or ECDSA) keep the same basic *public/private key idea*, but they use geometry instead of multiplication and factoring.

$$\text{Public key} = \text{Private key} \times \text{Base point on the curve.}$$

You can think of it like walking along a strange mathematical landscape — the *elliptic curve*. Going forward along the path (multiplying by the private key) is easy, but figuring out how far you walked just by looking at the final spot is almost impossible. That's the "elliptic curve discrete logarithm problem."

—

3. RSA vs. ECC: A Side-by-Side View



— Concept — RSA — ECC / Ed25519 — ————-———————— — **Math base** — Multiplying primes, factoring — Geometry on a curve — — **Hard problem** — Integer factorization — Elliptic curve discrete log — — **Security growth** — Bigger keys → more security — Same security with much smaller keys — — **Key size** — 2048 bits typical — 256 bits typical — — **Speed** — Slower (big exponentiation) — Faster (smaller arithmetic) — — **Introduced** — 1977 — 2005 (modern Ed25519 in 2011) — —

4. A Visual Metaphor

cryptographic evolution

RSA: Huge steel v **Ed25519:** Tiny titanium lock.
Just as strong but compact and nimble.

—

5. What Ed25519 Specifically Does

Ed25519 is a special case of *Edwards-curve Digital Signature Algorithm (EdDSA)* built on the curve called `Curve25519`.

- It's used for signing and verifying messages, not encrypting them directly.
- It's incredibly fast, especially on modern CPUs.
- It avoids many implementation pitfalls of older algorithms.
- It provides about the same security as a 3072-bit RSA key — with only 256 bits!

$$\text{Private key: } k$$
$$\text{Public key: } A = k \times G$$
$$\text{Signature: } (R, S) = (r \times G,\ r + H(R, A, M) \times k)$$

All operations happen on points on the elliptic curve — not with giant integer exponents.

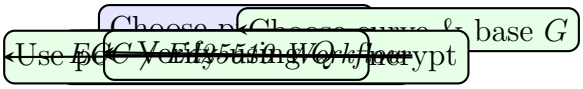
—

6. Why It Matters

- Smaller keys mean faster connections (think HTTPS, SSH, VPNs).
- Signatures are smaller — great for constrained devices or blockchains.
- The math is newer, but the logic is the same: one key to lock, one to unlock.
- Quantum computers may one day threaten RSA; ECC lasts longer (though not forever).

—

7. Final Comparison Diagram



—

8. Epilogue

RSA is still everywhere — old, wise, and reliable. But Ed25519 is like its younger, athletic cousin: it does the same job, just faster and lighter.

The world keeps both around, because understanding RSA teaches us the bones of cryptography, and understanding Ed25519 shows us where the field is going next.

"RSA built the foundation. Elliptic curves built the house."