Cryptographic Foundations

L02: What if someone could forge your digital identity?

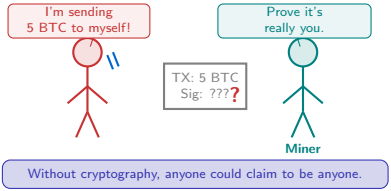
BSc Blockchain Course

What If Someone Could Forge Your Digital Signature?

Every time you send Bitcoin, the network must answer one question: **"Is this really you?"** There is no bank clerk to check your ID. There is no phone call to confirm. The only proof is a piece of mathematics called a **digital signature**. **Without cryptography, three attacks become trivial:**

- **Impersonation** – anyone claims to be you and spends your coins
- **Tampering** – a miner quietly changes "5 BTC to Alice" into "5 BTC to me"
- **Replay** – someone copies an old transaction and broadcasts it again

Cryptography is the invisible lock on every transaction. Without it, blockchain is just a shared spreadsheet anyone can edit.



Every blockchain transaction trusts mathematics instead of banks – but that trust is only as strong as the cryptographic primitives underneath.

Core tension: cryptography replaces institutional trust with mathematical proof. Break the math, and the entire system collapses.

What Makes a Cryptographic Hash Different from a Password?

Three cryptographic tools – three different jobs:

Property	Hash Function	Encryption	Digital Signature
Purpose	Fingerprint data	Hide data	Prove identity
Input	Any file or message	Plaintext + key	Message + private key
Output	Fixed-length digest	Ciphertext	Signature value
Reversible?	**Never**	**Yes, with key**	**Verifiable**
Blockchain use	Block linking, PoW	*Rarely used*	Every transaction

Pattern to notice:

Hashing is a one-way street – like putting a letter through a shredder that always produces the same confetti pattern for the same letter. Encryption is a lockbox with a key. Signatures are like signing a cheque that anyone can verify but nobody can forge. *Blockchain relies most heavily on hashing (for integrity) and signatures (for ownership). Encryption is surprisingly rare – Bitcoin transactions are public by design.*

The everyday analogy for each tool:

- **Hash Function:** "Fingerprint scanner"
  Feed in any document and get a unique fixed-length code. Change one comma, and the entire code changes. No way to reconstruct the document from its fingerprint.
- **Encryption:** "Lockbox with a key"
  You lock a message so only the keyholder can read it. With the right key, the original message comes back perfectly.
- **Digital Signature:** "Wax seal on a letter"
  Only you can stamp it (private key), but anyone can check the seal is genuine (public key). If the letter is altered after sealing, the seal breaks.

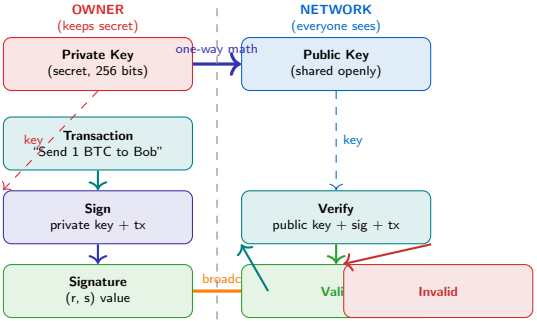
Passwords are secrets you share with a server. Cryptographic keys are secrets you never share with anyone.

Hashing proves data has not been tampered with. Signatures prove who sent the data. Together they give blockchain both integrity and authenticity – without a central authority.

Bitcoin uses SHA-256 for hashing and ECDSA for signatures. Ethereum uses Keccak-256 for hashing and the same ECDSA curve (secp256k1).

How Does a Blockchain Prove You Own Your Coins – Without a Bank?

The signing and verification pipeline:



Why this works without a bank: The private key is a secret number that only you know. The public key is derived from it using a one-way mathematical function – anyone can check your signature, but nobody can reverse-engineer your private key.

Four-step process in plain English:

1. You write a transaction ("Send 1 BTC to Bob")
2. You sign it with your private key – like stamping a wax seal
3. You broadcast the signed transaction to the network
4. Every miner verifies the signature using your public key

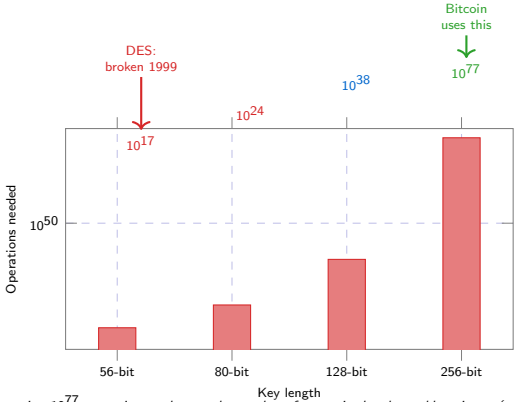
If the signature is valid, the transaction is accepted. If not, it is silently dropped. No human judgment is involved – the math decides. **Key insight:** The private key never leaves your device. Only the signature travels across the network.

Asymmetric cryptography lets anyone verify a transaction's authenticity without ever learning the sender's secret – this is the mechanism that eliminates the need for a trusted third party.

This is ECDSA (Elliptic Curve Digital Signature Algorithm) on the secp256k1 curve – the same scheme used by both Bitcoin and Ethereum.

How Hard Is It to Break These Cryptographic Puzzles?

Computational cost to brute-force different key sizes (operations):



A 256-bit key requires 10^{77} operations – close to the number of atoms in the observable universe ($\approx 10^{80}$).

What the numbers mean:

- **56-bit (DES):**
  Cracked in 22 hours in 1999. Old standards get retired.
- **80-bit:**
  A billion computers for a billion years would not finish. Safe today, but not future-proof.
- **128-bit (AES):**
  Current gold standard for symmetric encryption. No known attack comes close.
- **256-bit (Bitcoin):**
  Even a quantum computer (Grover's algorithm) only reduces this to 128-bit equivalent – still unbreakable.

Safety margin: Bitcoin chose 256 bits to buffer against unknown future attacks.

Blockchain security rests on the astronomical cost of brute-forcing a 256-bit key. The algorithms are public; the numbers do the protecting.

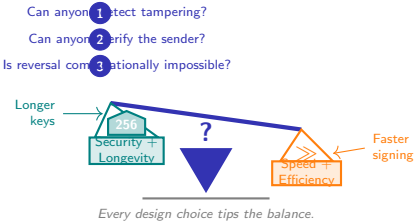
Source: NIST SP 800-57 (2020). DES crack: EFF, 1999. Grover's algorithm halves effective key length for quantum search.

Three Properties That Make Blockchain Cryptography Trustworthy

Before trusting any blockchain system, verify these three properties: **Property 1:**

Integrity – has the data been changed?
Hash functions detect any modification. Change one character in a block, and the hash changes completely – every block above it breaks. **Property 2: Authenticity – who sent this transaction?**
Digital signatures prove ownership without revealing the private key. Every node on the network can verify independently. **Property 3: Irreversibility – can the math be broken?**

One-way functions are easy to compute forward but astronomically hard to reverse. A 256-bit key would take longer than the age of the universe to crack. *If all three hold, security rests on mathematics. If any one fails, the chain is only as strong as its weakest human process.*



These three questions are a checklist, not a guarantee: cryptography protects against mathematical attacks, but human errors (lost keys, reused nonces) remain the real threat.

Bridge to L03 – Bitcoin Protocol: L03 shows how hashing, signing, and key derivation combine into a complete decentralised payment system.