L03 — Cryptographic Foundations

Cheatsheet

Hash Functions

Definition

$H : \{0,1\}^* \to \{0,1\}^n$ Any-length input $\mapsto$ fixed n-bit digest.
SHA-256: $n = 256$; output = 64 hex characters.

Five essential properties:

1. **Deterministic**: same input $\Rightarrow$ same output, always
2. **Fixed output**: output length constant regardless of input size
3. **One-way (preimage resistant)**: cannot reverse $H(x) \to x$
4. **Collision resistant**: infeasible to find $x \neq x'$ with $H(x) = H(x')$
5. **Avalanche effect**: flipping 1 bit changes ≈50% of output bits

SHA-256 security: $2^{256} \approx 10^{77}$ possible outputs; collision expected after 2^{128} attempts ($\approx 10^{22}$ years at 1 TH/s).

Uses in blockchain: block linking (prev hash), transaction IDs, mining puzzle (H(header) < target), address derivation.

Public Key Cryptography

Symmetric (shared key): same key encrypts and decrypts; key distribution problem.

Asymmetric (public/private key pair):

- **Public key** pk: share with everyone
- **Private key** sk: keep secret
- Easy to compute pk from sk; computationally infeasible to reverse

Bitcoin/Ethereum: use ECDSA on the **secp256k1** curve:

$$y^2 \equiv x^3 + 7 \pmod{p}$$

where p is a 256-bit prime.

Key generation:

1. Pick random $d \in [1, n-1]$ (private key, 256 bits)
2. Compute $Q = d \times G$ (public key; G = generator point)
3. Hash Q to obtain Bitcoin address

Security: *Elliptic Curve Discrete Logarithm Problem* (ECDLP) — given Q and G, finding d is computationally infeasible.

Digital Signatures (ECDSA)

Core Properties

Authentication: proves who signed.
Non-repudiation: signer cannot deny signing.
Integrity: detects if message was altered.

Sign with private key d, message m:

1. $z = H(m)$ (hash of message)
2. Choose random k; compute $R = k \times G$, let $r = R_x \bmod n$
3. $s = k^{-1}(z + r \cdot d) \bmod n$
4. Signature $\sigma = (r, s)$

Verify with public key Q:

1. $u_1 = z \cdot s^{-1} \bmod n, \quad u_2 = r \cdot s^{-1} \bmod n$
2. $P = u_1 \times G + u_2 \times Q$
3. Valid $\Leftrightarrow P_x \equiv r \pmod{n}$

Key insight: anyone can verify using the public key; only the private key holder can produce a valid signature. Proves ownership without revealing sk.

Address Derivation

One-way pipeline (each step is irreversible):

Step	Operation	Output
1	Random 256-bit number	Private key sk
2	$Q = sk \times G$ (ECC mult.)	Public key (65 B uncompressed)
3	SHA-256(SHA-256(Q))	Intermediate hash
4	RIPEMD-160( )	20-byte hash
5	Add version + checksum	Raw address bytes
6	Base58Check encode	Bitcoin address ("1…")

Modern formats: SegWit ("bc1q…", Bech32) and Taproot ("bc1p…", Bech32m) use WITNESS programs; lower fees, better error detection.

Merkle Trees

Structure: binary hash tree where each non-leaf node is the hash of its two children. Merkle Root stored in block header.

Proof of inclusion (Merkle proof): to prove Tx_i is in a block, provide the $\lceil \log_2 n \rceil$ sibling hashes on the path from leaf to root. Verifier recomputes root and compares with header.

Proof size: $O(\log_2 n)$ — for 1 000 txns: 10 hashes (320 B); for 1 000 000 txns: 20 hashes (640 B).

Root formula (4 leaves):

$$\text{Root} = H(H(H(T_1) \| H(T_2)) \| H(H(T_3) \| H(T_4)))$$

SPV verification: light client downloads only headers (∼75 MB); requests Merkle proof from full node; verifies transaction inclusion without trusting the full node for all data.

ECC vs RSA Comparison

Security	ECC key	RSA key	Ratio
80-bit	160 bit	1024 bit	6×
112-bit	224 bit	2048 bit	9×
128-bit	256 bit	3072 bit	12×
192-bit	384 bit	7680 bit	20×
256-bit	512 bit	15360 bit	30×

ECC advantages: smaller keys, faster signing/verification, less bandwidth and storage — critical for blockchain. Both Bitcoin and Ethereum use ECDSA secp256k1. RSA is not used in blockchain protocols.

BIP-39 Mnemonic & HD Wallets

Motivation: human-readable backup for private keys.

Generation:

1. Generate 128–256 bits of entropy (CSPRNG)
2. Append checksum (first entropy/32 bits of SHA-256)
3. Split into 11-bit groups; map each to BIP-39 wordlist (2048 words)
4. Result: 12 words (128 bit) or 24 words (256 bit)

Seed derivation: mnemonic + optional passphrase $\xrightarrow{\text{PBKDF2-HMAC-SHA512, 2048 rounds}}$ 512-bit seed.

HD wallet (BIP-32/44): seed → master key → unlimited child keys via derivation paths, e.g., `m/44'/0'/0'/0/0` (BIP-44 Bitcoin first address).

Key insight: one mnemonic backs up an unlimited number of key pairs.

Post-Quantum Cryptography

Threat: Shor's algorithm (quantum) breaks ECDSA and RSA in polynomial time. Grover's algorithm halves effective symmetric/hash key size (SHA-256 effectively 128-bit security against quantum attacker).

NIST PQC Standards (2024):

Algorithm	Use & Basis
CRYSTALS-Dilithium	Signatures; lattice (Module-LWE)
CRYSTALS-Kyber	Key encapsulation; lattice
SPHINCS+	Signatures; hash-based (conservative)
FALCON	Signatures; lattice (NTRU)

Timeline: cryptographically relevant quantum computers estimated 2030–2040 (highly uncertain). "Harvest now, decrypt later" means migration planning should begin now.

Blockchain impact: Bitcoin/Ethereum addresses spending exposed public keys are at risk; addresses not yet spent (hash-protected) have extra time.

Key Formulas

Hash function: $H : \{0,1\}^* \to \{0,1\}^{256}$ (SHA-256)

ECDSA signature: $\sigma = \text{Sign}(sk, m)$; valid iff $\text{Verify}(pk, m, \sigma) = \top$

Key pair: $pk = sk \times G$ on secp256k1

Birthday bound: collision after $\approx 2^{n/2}$ hashes for n-bit output

Merkle proof: $\lceil \log_2 n \rceil$ hashes to prove inclusion in n leaves

BIP-39 entropy: 12 words = 128 bits; 24 words = 256 bits

secp256k1 curve: $y^2 = x^3 + 7 \pmod p$, $p = 2^{256} - 2^{32} - 977$

Preimage resistance: 2^{256} attempts to find x given $H(x)$

Terminology Quick Reference

Term	One-Line Definition
SHA-256	256-bit hash function; Bitcoin's core primitive
ECDSA	Elliptic Curve Digital Signature Algorithm
secp256k1	Elliptic curve used by Bitcoin & Ethereum
Private key	Secret 256-bit random number; proves ownership
Public key	$sk \times G$; shared openly; verifies signatures
Address	Hash of public key; where funds are sent
ECDLP	Elliptic Curve Discrete Log Problem; basis of security
Merkle root	Root hash summarising all transactions in a block
SPV	Light client; verifies via Merkle proofs only
BIP-39	Standard for mnemonic seed words (12/24 words)
HD wallet	Hierarchical Deterministic wallet from single seed
Shor's alg.	Quantum algorithm breaking RSA/ECDSA in poly time
Dilithium	NIST post-quantum lattice-based signature standard
Avalanche	1-bit input change ⇒ ∼50% output bits flip