Blockchain Technology (Unit II Deep Dive)

Blockchain is a **decentralized, append-only ledger** that securely records transactions across a peer-to-peer network [1] [2]. Data are grouped into *blocks*, each containing recent transactions and a cryptographic reference to the prior block. For example, in Bitcoin each block header includes the SHA-256 hash of the previous block's header, as well as the Merkle-root hash of its transactions[3]. This linking of hashes means **every block depends on all prior ones**: "blockchain is a chain of blocks where each block is linked to its previous block by referencing the previous block header's hash" [2]. Once added, blocks cannot be altered without redoing the chain from that point onward. Because **every full node** in the network holds a copy of the blockchain, there is no central point of failure or authority [4] [5]. Data integrity is ensured by cryptography: each transaction is time-stamped and signed by the sender (see *Cryptography* below), providing a single "source of truth" that cannot be changed without detection[6] [4].

- **Decentralized Ledger:** All participating nodes keep a copy of the entire blockchain, synchronizing updates by consensus. There is *no central server*; anyone can run a node and verify the ledger. (Full nodes download all transactions and blocks and enforce the protocol rules [7]. Light or pruned nodes download only recent data and rely on full nodes for history.) The more nodes in the network, the more **decentralized and resilient** it is: "nodes are the source of truth… The more nodes a blockchain hosts, the more decentralized it will be. A high node count ensures resilience… and increases the difficulty level for infiltration" [5].

- **Immutable Chain:** Each block's header contains metadata (e.g. timestamp, version) and hashes that tie the chain together [8] [9]. By including the prior block's hash and Merkle root of transactions [3], the blockchain forms an unbroken sequence: any change to a past block would break the hash links in all following blocks. As a result, once a block is confirmed by consensus it becomes effectively permanent in the ledger .[2] This **append-only** property ensures a verifiable history: all transactions are time-stamped and recorded in order, making the ledger tamper-evident[10] [11].

- **Security Guarantees:** Blockchain uses cryptography and economic incentives to secure the network. Cryptographic hashes (like SHA-256) provide integrity and link blocks; digital signatures verify identities (see *Cryptography* below). An attacker would need to control a majority of the network's computing power (a "51% attack") to tamper with transactions. In large public blockchains this is extremely difficult. As one analysis notes, taking over the network "requires a massive amount of computing power," making fraudulent changes **highly impractical**[12]. Furthermore, consensus mechanisms (see below) ensure that honest nodes will always accept the majority chain, protecting against double-spends and other attacks.

- **Blockchain Diagram Illustration:** *Figure: Example of a blockchain diagram with linked blocks and distributed nodes.* Each block contains transaction data and references to previous blocks, forming a chain that is replicated across all nodes. This peer-to-peer structure makes the ledger fault-tolerant and transparent (all participants see the same data) [2] [10].

Key Concepts: A *block* is a batch of validated transactions. The **blockchain** is the sequential chain of all blocks, secured by cryptographic hashes [13] [2]. A *node* is any device running the blockchain software; nodes store the ledger, validate new data, and broadcast information. There are also specialized nodes called *miners* (in PoW systems) or *validators* (in PoS systems) that create new blocks. The **block header** typically includes fields like the previous-block hash, Merkle root of transactions, timestamp, difficulty target, and a nonce for mining [14] [15]. These elements (especially the hash pointers) ensure that blocks remain immutable once added.

Characteristics & Benefits: Because blockchain is distributed and cryptographically secured, it provides a **trusted, auditable, and single source of truth** without needing a central authority [1] [4]. All parties can verify the entire history of transactions. This transparency (all transactions on a permissionless blockchain are visible to participants) can reduce fraud, errors, and the need for intermediaries [10] [16]. The system is also **fault-tolerant**: if one or many nodes go offline, others continue to maintain the network without interruption [17] [5].

Exam Question: *Explain how the blockchain's peer-to-peer architecture and cryptographic design ensure data integrity and decentralization. Why is it said that blockchain provides a "single source of truth" without a central authority?*

Network Nodes and Consensus

Blockchain networks are **peer-to-peer**. Nodes communicate directly, relaying transactions and blocks among themselves. There is **no master node**; every node follows the same protocol rules. In practice, nodes perform three main functions: **maintenance** (storing and propagating the ledger), **validation** (checking incoming transactions/blocks against consensus rules), and **accessibility** (serving blockchain data to users) [18] [7]. Full nodes keep a complete copy of the blockchain and validate all new transactions/blocks [18] [7]. Some nodes (miners in PoW or validators in PoS) go further by creating new blocks when they win the consensus puzzle.

- **Roles of Nodes:**
- **Full Nodes:** These are the backbone of the network. They "preserve a blockchain's transaction history, sync, store, copy and distribute data while also validating new blocks" [7]. Full nodes enforce the protocol rules: they reject invalid transactions/blocks and add valid ones to their copy of the ledger.
- **Mining (PoW) / Staking (PoS) Nodes:** These nodes actively participate in the consensus mechanism. For example, Bitcoin miners use Proof-of-Work to verify transactions and solve cryptographic puzzles. As one source explains: *"mining nodes verify transactions using a proof-of-work consensus model…miners…compete to solve complex mathematical problems* [19]*"*. The winning miner adds a new block and is rewarded with newly-minted cryptocurrency (the "block reward") [19] [20].

- **Light/Pruned Nodes:** These nodes download only recent parts of the chain (block headers or recent transactions). They rely on full nodes to validate and request older data as needed, conserving storage and bandwidth.

- **Decentralization and Resilience:** There is **no single point of control**. As Nicholas Edmonds (Topl) notes, *"Nodes are the source of truth for a blockchain… The more nodes a blockchain hosts, the more decentralized it will be."* A high node count spreads trust and prevents any one party from controlling the network [5]. In effect, consensus is achieved by majority agreement: new blocks are accepted only if most nodes follow the same chain of valid blocks. This makes the network highly tolerant of failures or attacks on individual nodes.

- **Consensus Mechanisms:** To agree on which blocks to add, blockchains use consensus protocols. In Bitcoin (the original and a widely studied model), consensus is reached through **Proof-of-Work (PoW)** combined with the **longest-chain rule**[21] . Under PoW, all mining nodes race to solve a difficult hash puzzle: whoever finds a valid solution first gets to publish the next block. This is effectively a “grand competition between all nodes” [22]. Proof-of-Work creates randomness in block creation (each winning miner is hard to predict or coerce) and prevents spam/attacks by making block creation costly [23] [24] . The network then adopts the longest valid chain; any fork that is shorter is discarded.

Other blockchains may use different consensus methods. For example, **Proof-of-Stake (PoS)** systems select validators based on the amount of cryptocurrency they hold (“stake”) rather than computational work [25] . Some newer designs even use Proof-of-Elapsed-Time (PoET) or delegated/proof-of-authority schemes. These alternatives aim to reduce energy use while maintaining security. Regardless of method, the goal is the same: to allow a decentralized network to come to consensus on transaction order and to prevent double-spending or Sybil attacks [21] [23] .

- **Attacks and Security:** The network’s security depends on honest consensus. In PoW systems, an attacker would need to control over 50% of the hashing power to rewrite history. Because this requires enormous resources, it is generally infeasible on large networks[12] . As more nodes join and more computational power is added, the cost of such an attack grows. Thus, “the larger the network gets, the more difficult it is to alter any transactions stored on its blockchain”[12] . In PoS networks, similar majority (“51% of stake”) attacks are theoretically possible but are also very costly for a rational adversary.

Exam Question: *Describe how blockchain nodes reach consensus in a public blockchain like Bitcoin. Contrast Proof-of-Work with Proof-of-Stake and explain how consensus rules prevent double-spending.*

Blocks and Data Structure

Each **block** in a blockchain contains a batch of transactions and metadata. A typical block has two parts: a **header** and a **body**. The header (often 80 bytes in Bitcoin) includes: - A *version number* indicating protocol rules [26]. - The *previous block’s hash* (a 32-byte double-SHA256 hash)[26]. - The *Merkle root* (another 32-byte hash summarizing all transactions in the block) [27]. - A *timestamp* (when the miner started hashing) [15]. - A *difficulty target* (expressed in compressed form). - A *nonce* (a counter the miner adjusts to find a valid hash) [15] .

The body contains all the transactions included in that block. The **Merkle tree** allows the block header to commit to every transaction efficiently: transactions are hashed pairwise up the tree until a single Merkle root is obtained [28]. This enables quick proof that a given transaction is in the block (via a Merkle branch) without revealing all data.

Once the header is set, the miner computes the block hash (double SHA-256 of the header in Bitcoin). For a valid block, this hash must meet the network’s difficulty target (e.g. have a certain number of leading zeros).

The miner changes the nonce (or other fields) until a qualifying hash is found. The block is then broadcast; other nodes verify its validity by checking the proof-of-work and the transactions inside.

Blocks are **chained** by having each block header include the previous header's hash. As one source explains: *"blockchain is a chain of blocks where each block is linked to its previous block by referencing the previous block header's hash… no transaction can be modified unless the block… and all blocks that follow it are also modified."* [2] . The first block in the chain, called the *genesis block*, has no predecessor and is hardcoded into the software. Each new block increases the *height* of the chain by one.

Key points about blocks:
- **Transaction Recording:** Each block permanently records a set of new transactions. Once a transaction is in a confirmed block, it is effectively immutable.
- **Block Reward & Fees:** The block header also includes a special "coinbase" transaction that grants newly-created tokens (and any collected fees) to the miner. In Bitcoin, the coinbase reward started at 50 BTC and halves roughly every 210,000 blocks (currently 6.25 BTC)[20] . This minting of tokens is what incentivizes miners to secure the network.
- **Orphan Blocks:** Occasionally two miners find valid blocks at the same height simultaneously. One block will become orphaned when the network ultimately adopts the longer chain; the orphan block is dropped (its transactions go back to the pool).
- **Data Integrity:** Because each block's hash depends on all transactions (via the Merkle root) and the previous block's hash, tampering with any transaction would change that block's hash and break the chain. Nodes would immediately reject a tampered block since it would not match the recorded hash[2] [13] .

Exam Question: *Outline the structure of a blockchain block (including header and transactions) and explain how these elements ensure the immutability of the ledger. How does hashing link the blocks?*

Cryptography in Blockchain

Cryptography underpins blockchain security. Two main primitives are used:

- **Hash Functions:** A cryptographic hash (e.g. SHA-256 in Bitcoin) takes any input data and produces a fixed-size string (a digest) that is practically impossible to invert or collide. Blockchains use hashing in several ways:
- *Linking Blocks:* Each block header includes the hash of the previous block header [3] . If any data changes in an earlier block, that block's hash would change and invalidate all following blocks.
- *Proof-of-Work:* Miners compute the hash of the block header repeatedly (by varying the nonce) to find a value below the network's difficulty target.

- *Merkle Trees:* The root hash of the Merkle tree commits to all transactions in a block [28] . This allows a compact proof that a transaction belongs in that block without revealing all other transactions.
  In short, hashing provides **integrity**: any alteration of input data leads to a completely different hash, which nodes will detect.

- **Public-Key Cryptography:** Blockchain wallets and transactions rely on public-key (asymmetric) cryptography for identity and authorization. Each user has a key pair: a **private key** (kept secret) and a **public key** (shared publicly). The public key (often hashed further into an address) identifies the account, while the private key is used to sign transactions. As Warburg et al. explain, *"public-key* *cryptography… uses two mathematically related, but not identical, keys: a public key and a private key”*[29]. Only the holder of the private key can create a valid digital signature for a transaction. Other nodes use the corresponding public key to verify that the signature is genuine and that the transaction hasn’t been altered. Because the private key cannot feasibly be derived from the public key, this scheme is unforgeable. In Bitcoin, addresses are simply public keys (or hashes of them) in human-readable form [30].

In practice, when you send cryptocurrency, your wallet software creates a transaction and signs it with your private key. Every full node then verifies that the signature matches your public key (address) and that you had sufficient funds. This **authentication** step prevents unauthorized spending. The combination of hashing and digital signatures means that the network can verify both *what* the data is (integrity) and *who* authorized it (authenticity) without trusting any single party.

Exam Question: *Explain how cryptographic hash functions and public-key signatures are used in blockchain. How do they contribute to the security and verifiability of transactions?*

Wallets and Addresses

To interact with a blockchain, a user runs a **wallet** – software (or hardware) that manages cryptographic keys. A blockchain wallet generates one or more key-pairs. The **public key** (or a derived address) is the account’s identity on the network, while the **private key** is the secret that authorizes spending. As one slide notes: *“Blockchain wallets provide individual users the ability to transact on a blockchain network… Each account is assigned an immutable account-based identity (a public key) through which to interact with…other users.”*[31] [32]. In other words, *having a wallet means having an account on the blockchain*.

- **Public Addresses:** These are often shorter, user-friendly forms of the public key. For example, a Bitcoin address is a 34-character alphanumeric string derived from the public key. Anyone can send funds to your address; the network will record that “coins” are assigned to the corresponding public key. All transactions involving that address are publicly visible on the blockchain (though they don’t contain personal names). Warburg et al. describe it as akin to a credit account number that is visible to all: *“public keys are also known as wallet addresses… [they] can be seen exactly how many digital tokens are in everyone’s wallets (without knowing exactly who controls each wallet)”*[30] .

- **Private Keys:** This is a secret 256-bit number (usually encoded as 64 hex characters) that allows you to spend coins from the corresponding address. Wallet software uses the private key to create digital signatures on transactions. Because of asymmetric cryptography, *“no one can reverse engineer your public key in order to find out your private key”*[30] . If you lose your private key, you effectively lose access to any coins at that address. Conversely, anyone with the key can spend the funds, so securing your private key (often by encryption or storing offline) is critical.

- **Wallet Types:** Wallets come in many forms. Software wallets run on computers or phones and manage keys there. Hardware wallets (USB devices) store keys offline for extra security. “Cold storage” wallets keep keys on paper or offline devices with no network connectivity. All these are simply different ways of keeping the private key safe while allowing you to sign transactions when needed.

Importantly, the blockchain **does not store personal data or actual currency in your wallet**. Instead, it stores balances linked to addresses. The wallet itself *does not hold coins*; it holds keys. Coins exist only as entries in the ledger. When you send funds, your wallet creates a transaction saying “send X coins from my address to another,” signs it, and broadcasts it to the network.

Exam Question: *Describe how blockchain wallets use public and private keys for transactions. Compare this system to a traditional bank account: what are the similarities and key differences?*

Digital Tokens and Incentives

Most blockchains come with a built-in digital currency (a **native token**) that serves two roles: *currency* and *incentive*. For example, Bitcoin’s native token is BTC and Ethereum’s is Ether (ETH). These tokens *only exist as entries on the blockchain ledger*[33] . When you run a full node, you’re actually tracking who holds how many tokens (e.g. by address balances or UTXOs).

- **Token as Value & Payment:** Users transfer value by sending tokens to each other via transactions. These tokens are what give the blockchain **monetary value**. Many new blockchains include tokens in their design to enable decentralized economies. Some tokens are “stablecoins,” pegged to fiat (like USDC or DAI)[34] , used to minimize volatility.

- **Mining Rewards:** Blockchain protocols often mint new tokens to reward the nodes that secure the network. In Bitcoin, each new block currently awards 6.25 BTC to the miner (plus any transaction fees). This reward started at 50 BTC in 2009 and halves roughly every four years [20] . These tokens are created out of thin air by the protocol, and assigning them to the miner provides a financial incentive to contribute compute power. This is the “block subsidy” part of Bitcoin’s issuance.

- **Network Effects:** As one explanation notes, the *main value creation at the core infrastructure layer* is tied to **network usage**[35] . In simple terms, the more people use a given blockchain (more transactions, more apps), the more valuable its native token tends to become (similar to how a stock’s value can rise when a company does well). This positive feedback (often called a network effect) is a key economic motive for launching a blockchain with its own token [35] .

- **Security Incentives:** Tokens align economic incentives with security. Honest miners/validators spend real resources (electricity, stake) to secure the chain. Because they earn rewards, they are motivated to follow the consensus rules. A malicious actor, on the other hand, would stand to lose its stake if it violated rules (nodes would reject the invalid blocks). In Warburg’s words, *“Proof of Work creates some randomness for articulating which node wins… it will be hard to predict and therefore hard to coerce a winning participant to change parts of the block of transactions.”* This makes the system trustless and self-enforcing[23] .

Exam Question: *Explain the role of cryptocurrency tokens in a blockchain network. How do block rewards and token economics motivate participants and protect the blockchain’s integrity?*

Consensus and Security Details

Blockchain consensus combines *Sybil resistance* (preventing a single entity from pretending to be many) with protocols for agreement. In Bitcoin, Sybil control is achieved via Proof-of-Work, while agreement is via the longest-chain rule [36]. Other chains may use Proof-of-Stake or voting-based approaches for Sybil resistance.

A consensus protocol is the set of rules every node follows to accept or reject data. For example, nodes will only add a new block if: (1) its hash meets the difficulty target (in PoW), (2) all transactions in it are valid (no double spends, properly signed), and (3) it correctly references the prior block in the chain. Over time, nodes see many competing candidate blocks but will always choose the chain with the most cumulative work (or stake) behind it. This is sometimes called the *longest valid chain*.

Because blockchain is append-only, conflicting transactions can only be resolved by which one ends up in the chain. A **double-spend** attempt (spending the same coin in two places) is prevented by consensus: once one copy of the transaction is recorded in a block, all honest nodes will reject any later block that tries to double-spend that coin.

Security Guarantees: A correctly implemented consensus protocol protects against tampering and attacks. As described earlier, an adversary would need to control a majority of the consensus power (hashrate in PoW or stake in PoS) to unilaterally change history. In large public chains this is prohibitively expensive. Even if an attacker gained majority power, they could at best reorder or rewrite a small number of recent blocks; the network could still build a longer honest chain over time.

Consensus Variations:
- *Proof-of-Work (PoW):* Used by Bitcoin and many others. Requires heavy computation (energy) to solve puzzles. Provides strong security at the cost of energy use [24].
- *Proof-of-Stake (PoS):* Used by Ethereum 2.0, Cardano, etc. Validators "stake" coins and are randomly chosen to propose/validate blocks, with penalties for misbehavior. This requires much less energy but relies on different assumptions.
- *Others:* Some systems (e.g. Hyperledger, Ripple) use voting or permissioned setups where a consortium of known nodes reach agreement by round-robin or PBFT-like algorithms. These typically trade off some decentralization for higher performance and privacy.

Regardless of method, all blockchains share the same goal: **distributed consensus without trust**. Each node independently verifies all the rules, so the system is self-enforcing. In public blockchains, this creates a form of "peer-to-peer trust" – anyone can audit the chain at any time.

Exam Question: *Discuss how blockchain consensus algorithms (like Proof-of-Work and Proof-of-Stake) achieve agreement among nodes. How does the choice of consensus affect network security and resource usage?*

---

[1] [4] [6] [16] What Is Blockchain? | Oracle
https://www.oracle.com/blockchain/what-is-blockchain/

[2] [3] [8] [9] [14] [15] [26] [27] [28] Mastering Blockchain_full book.pdf
file://file-Xu5n2MfCb6xkL3YF8ZsK1g

[5] [7] [18] [19] [20] What Are Blockchain Nodes and How Do They Work? | Built In
https://builtin.com/blockchain/blockchain-node

[10] [11] [12] [13] [17] [21] [22] [23] [24] [25] [29] [30] [36] Bettina Warburg_ Tom Serres_ Bill Wagner - Basics of Blockchain.pdf
file://file-8YjDrp7oickvps6B2Trjs5

[31] [32] [33] [34] [35] Blockchain chap 2.pdf
file://file-WoF9rYnK5rfCfsTq8UduFR