Unit-3 [Distributed consensus]

Distributed consensus.

Different computer (nodes) agree that a transaction is valid. It combines both Proof of work & longest chain rule.

Principles

* PoW

- Miner spend resources to solve tough maths puzzles to propose the next block (new block approx 10 min)

Process

queue pending transaction into a block → solve a puzzle by finding a nonce → share new block with → other nodes verify

* Longest chain rule

- The one that has spend the most computational resources and has the biggest chain is considered valid.

* Decentralised

Anyone can join in and participate

Fork resolution

two miners can mine same block (creating a fork). the new pills the longer one

Incentives

- Block rewards — new coin for miner
- Transaction fees — extra money for trxn processed.

Advantage

- security
- decentralised

limitations

- High energy
- scalability: big n/w → slow.

Proof of stake (To fix the energy wastage problem)

Process

users lock some of their coin → system randomly select validators based on how & since they have staked. → validators create new & verify blocks. → Bad behav causes to lose stake (penalty).

Advantage

- Energy eff — no heavy computation.
- Faster trxn — no mining, confirm faster

limitations.

- Wealth central. — rich ppl more stake
- security risk — Attack like "nothing at stake".

Proof of Burn (PoB).

[ppl destroy their coin to show they are serious abt supporting n/w]

Process

sends coin to a unusable address (burn) → Higher burn = Higher chance of validating blocks & earn rewards → Misbehaviour cause loss of validation rights

Advantage

- Energy efficient ← no mining
- long term commitment ← Burn coin = dedication

limitations

- Waste of resources ← paise jalana
- Centralisation ← Rich = more coin to burn

Difficulty level · (kitna mushkil puzzle mushkil kyunki block production steady).

If block banane jaldi create hoti = increase difficulty

if block slow create hoti = decrease diff.

steady production.

Har 2 weeks change nota hai difficulty (approx 2016 block)

Impact

- More competition ← tough puzzle = tough mining
- High energy usage ← tough puzzle & more energy

Sybil Attack

(ek attacker multiple fake identity (node banata hai)

How it works

- voting manipulation
  - zyada node = zyada vote
- Flooding
  - Fake node = Fake data
- Isolation
  - real node ko fake node se ghera lo.

How to prevent?

- PoW
- PoS
- Reputation system.

Energy utilisation (high puzzle : more electricity)

consumption

- More mining : More competition : More difficulty

How to solve

- PoS
- PoB

Environment pe impact

- Heavy mining = impact on env. ↑

Alternate solution = Smart contracts

(programs jo automatically agreement run karte hai)

Types

- Ethereum : Traditional smart contract
- Tezos, Cardano : new

Alternate

- Formal verification (Deploy se pehle check kro).
- off chain (Block chain se bahar calculate kro).

Challenges

- security
- scalability

Unit - 4 [cryptocurrency]

• Distributed Ledger Technology (database spread across multiple places w/o central admin.)

Blockchain
- Block linked w/ previous block hashes
- PoW & PoS
- Used in crypto, finance, voting

Directed Acyclic Graph (DAG)
- No block trxn linked directly
- confirmed by other trxn, no miner
- used in IOTA machine2machine w/ low fees

Holochain
- Each user own chain
- local validation no global
- used in scaled Dapps like social n/w

Hashgraph
- node gossip about trxn
- virtual voting
- used in financel gaming

Tempo (Radix)
- logical clock instead of block
- used in DeFi

Protocols

Peer2peer
- each node has copy of entire trxn
- direct comm^{n} w/o central server

PoW
- solve complex puzzle to validate

Cryptographic Algo
- SHA-256 (Hash) - secure trxn
- ECSDA (Digital sign) - ensure authenticity

Mining Strategy

- solo: full reward low chance
- pooled: shared resource : reward
- cloud: rent pe kro
- Merged: two bitcoin mine kro (Bitcoin + Namecoin)

Ethereum construction

Ethereum virtual machine (EVM)
- global computer that executes smart contracts

smart contract
- self executing code with conditions in it

Account Types
- Externally owned Accounts (EOA) [controlled by priv key]
- Contract account [controlled by smart contract code]

Ethereum Block chain structure
- use merkle & RLP encoding for eff. storage

Gas Limit (it limits excessive computation & pays for transactions)

- Types

Txn limit - ek txn ka limit
Block limit - pura txn in a block ka limit

★ Higher gas = faster txn inclusion

Decentralised Autonomous org (DAO) [org smart contract se chalta hai no leadership]

- Did not work as it was hacked due to smart contract vul.

Smart contract [stored on Block C run automatically when condition met]

Advantage
- trustless
- automatic
- cost eff

Application
- insurance claim
- voting

Challenges
- Bugs
- Difficulty updating once deployed.

GHOST Protocol – Faster block times [used in eth.]

[Greedy heaviest observed SMC.]

- chooses chain w/ cummalitive work [not just longest]

★ Attacks

Double spend attack - spend same coin twice

Sybil attack - single user = multiple nodes

Eclipse attack - isolate a node

51% Attack - control majority mining power

Rentracy attack - call a fn recursively

Replay attack - Reuse a txn in diff n/w

Dusting attack - spend tiny amount to deanonymise users.

DOA hack - Famous re-entrancy attack to hack DOA.

Sidechain

- seperate chain connected to main chain
- Allow users to experiment
- Tokens can be moved b/n both chains

Eg. Bitcoin
Ethereum

namecoin

- first fork of bitcoin at decentralised domain names
- Mapping .bit domains to address securely
- Enhances censorship but limited adoption

Unit 5 [Crypto regulations]

Stakeholders

- **Government**
  - prevent money laundering
  - Fincen → USA
  - FCA → UK
- **Crypto exchange**
  - must follow AML (Anti-money laundering)
- **Investor & traders**
  - must follow tax laws & report earnings
- **Blockchain dev & miners**
  - Maintain infra. & ensure n/w sec

Legal Aspects

- **Licensing**
  - exchanges must register w/ auth.
  - License ensure all regulation are followed
- **AML & KYC**
  - AML - Prevents illegal money into system
  - KYC - Forces exchanges to store customer identity.
- **Tax policies**
  - capital gain on trading profit
  - Income tax on mining / staking reward
- **Security & consumer protection**
  - Use strong cybersec measures
- **Cross Border**
  - Banned in some country.
  - No global regulation

Challenges

- Anonymity - Hard to track private coins
- Decentralisation - No central auth.

Blackmarket usage -

1. Buying illegal goods
2. Laundering money
3. Paying ransom.

Impact on Global economy

Positive

- Financial Inclusion - Bankless ppl
- Remittance - Faster, cheaper money transfer

Negative

- volatility : sudden price swings
- speculative bubble : overhyped crypto crash horribly.

∀ Applications of Blockchain

IoT

- Tamper proof for device2device comm
- Smart contract automate tasks
- remove single point of failure
- Ensures data integrity.

eg - Shipment Tracking
Smart home

DNS

- Stops DDoS on domain servers
- Allows anonymous domain registrations

eg - ĐDNS (eth-dns).

medical record

- decentralised, secure storage
- controlled access for doctor, patient

eg - Prevents hacking / tampering

Future Appl^n.

- voting
- Digital id.

Case study : Mining puzzle

concept : Mining Puzz ensure sec. by requiring miner to solve problems

Imp : Makes the n/w hard to attack, diff. & expensive

Unit-2 [Intro to block chain]

Advantages / Features of Blockchain over Traditional DBs.

1. Decentralisation: No single controlling authority
2. Security: Use strong crypto like SHA-256
3. Immutablity: Data once recorded can't be changed.
4. Transperency: All participants can see history.
5. Trustless env: Trust is gained through math & code.
6. Consensus mechanism: Ensure all participant agree on the data.

Types

- **Public**
  - open to everyone
  - (Bitcoin, Eth)
- **Private**
  - only authorised participants
  - (Hyperledger Fabric)
- **Consortium**
  - controlled by a group of orgs
  - (R3 corda)
- **Hybrid.**
  - mix of priv + pub
  - (Dragonchain)

Pipeline of Blockchain

Transaction creation (sender initiates txn) → Propogation (Txn spread into n/w) → Validation (node check txn rules) → Consensus & block addition (Agreement reached block added) → Ledger update (Every node updated into db)

→ finalisation (Txn perm. part of blockchain)

Patricia Tree (combines tree & a hash table)

Structure
- compressed tree that stores key-pair efficiently.

Appl^n
- used in eth to manage db
- stores account balance, smart contract & trxns

Advantages
- fast lookups
- eff. storage
- easy to verify data consistency.

Transaction & fees

Structure
input : where coin come from
output : " " go to
signature : prove senders ownership

Lifecycle
Created → Broadcast → validated → Mined → confirmed

★ · Higher fees = faster confirmation
· Fees vary on n/w congestion.

- Anonymity - (identity hidden behind random addresses)

techniques - Mixing service, Stealth address, Ring signatures

challenges - conflicts with regulatory compliance

Chain policy in Blockchain [Rules for updating the chain]

★ · Forks

Soft

(Backward compatible blockchain)

- maintains one chain

Hard.

(creates a new chain).

- splits into two

Decisions - voting is done on updates & changes

Security - Policy to prevent attack & ensure integrity.

Lifecycle of Blockchain App

concept & use case development
↓
Design & Architecture planning
↓
Implementation & planning
↓
Testing (security, perf.)
↓
Deployment to the n/w
↓
operation & Maintenance
↓
end of life (decomissioning)

Public	Private
• open to anyone • community driven • Full transperency • slow • High security	• Restricted. • central auth • controlled • fast • lower sec (centralised risk)

Unit-1 [Introduction to cryptography]

Cryptography Features.

- **Confidentiality** — only authorised can read the info.
- **Integrity** — Information remains unchanged during transmission
- **Authentication** — verify the sender
- **Non-repudiation** — the sender cannot deny sending the msg.

Types of crypto.

- **Symmetric**
  - same key
  - eg - AES
- **Assymetric**
  - 2 keys
  - eg - RSA, ECC
- **Hash Fn.**
  - Take Input data & return a fixed-size string.
  - unique for each input

* Hash function.

(mathematical f^n takes input & returns fixed hash value).

properties

- Deterministic - same input gives same o/p
- fast computation - Hashing is quick.
- Pre-Image resistance - Hard to guess i/p from o/p.
- Avalanche effect - Tiny change drastically changes o/p.

common hash functions

- MD5 — 128 bit
- SHA-1 — 160 bit
- SHA-256 — 256 bit
- SHA-3 — New standard.

. Collision resistance - Hard to find 2 i/p w/ same hash.

App^{n}

- password storage
- Verifying digital signature.
- Ensure file/data integrity

* Hash Pointer. (A data structure that store hash of same data along with a pointer to where that data is stored).

how they work?

- like regular pointer but also verify data integrity

Any change in data changes the hash.

In blockchain?

·every block points to the previous one through hash pointer. If any block's content changes, the # pointer also changes breaking the chain's integrity.

Application

- Blockchain - Each block has hash pointer to previous block
- Merkle tree - Tree where each node is a hash.

one-way function. (easy to compute but hard to reverse)

Eg. SHA-256, RSA encryption

- Backbone of secure comms.

complexity classes - P, NP, NP-Hard

class P - Problems solvable in P (polynomial) time

class NP - Problems which can be verified quickly.

NP-complete - Hardest problems in NP. All can be solved quickly.

NP-Hard - Atleast as hard as NP, but not easily verifiable.

ECDSA (based on ECC providing security w/ shorter key).

Digital sign properties - Authentication, Integrity, Non-repudiation

Key generation - Priv key (d), Public key (Q)

Signing Process - Hash the msg → Pick random no k → compute (r, s) as sign using elliptic curve.

Verification problem - Use public key to check if sign matches the msg hash.

Directed Acyclic Graph (DAG) - Instead chain of blocks, DAG uses a graph where txns confirm each other.

Benfit	Challenge
· Faster	· Harder to maintain security & consensus.
· No mining fees	

Zero-Knowledge Proof

[where one party reveals that they know secret w/o revealing it.]

Characteristics:

1. Completeness - If true, convinces verifier
2. Soundness - If false, cannot convince
3. Zero-knowledge - No secret leaked.

Byzantine Problem

[Achieving consensus in a system where some may lie]

↙ [using algo that tolerate faults]

eg - 30 nodes present. How many faulty, how many min. req.

fault tolerance $n \geq 3f+1$

$30 \geq 3f+1$ $\quad f = 9.66 \approx 9$ nodes.

Min req to agree $2f+1$

$2 \times 9 + 1 = 19$ nodes.

Quantum computing.

[They can use 0 & 1 simultaneously]

Impact on Crypto

- RSA & ECC: Broken by Shor Algo.
- AES: weakend by Grover

Challenges

- Errors
- Instability - Qubits cannot survive in outer env.

Future outlook.

- Post Quantum Crypto.
- Hybrid (classic + quantum).

Diagrams

- Hash pointer

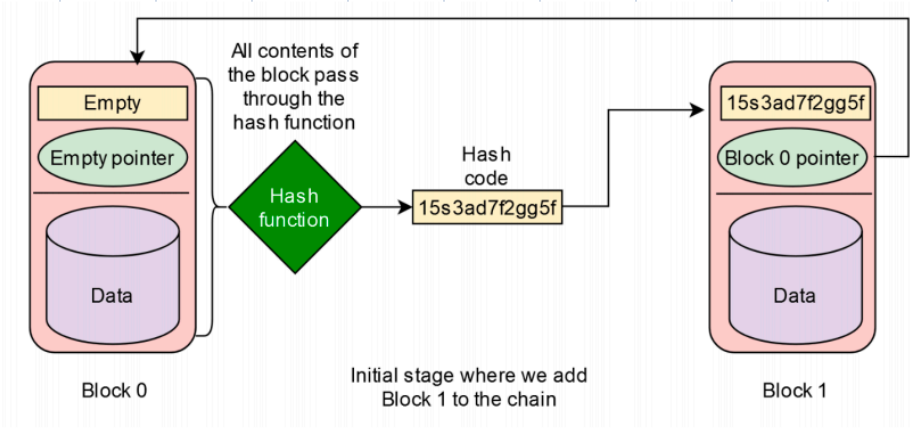

Figure 1.2: Hash Pointers in Blockchain

- Merkle tree

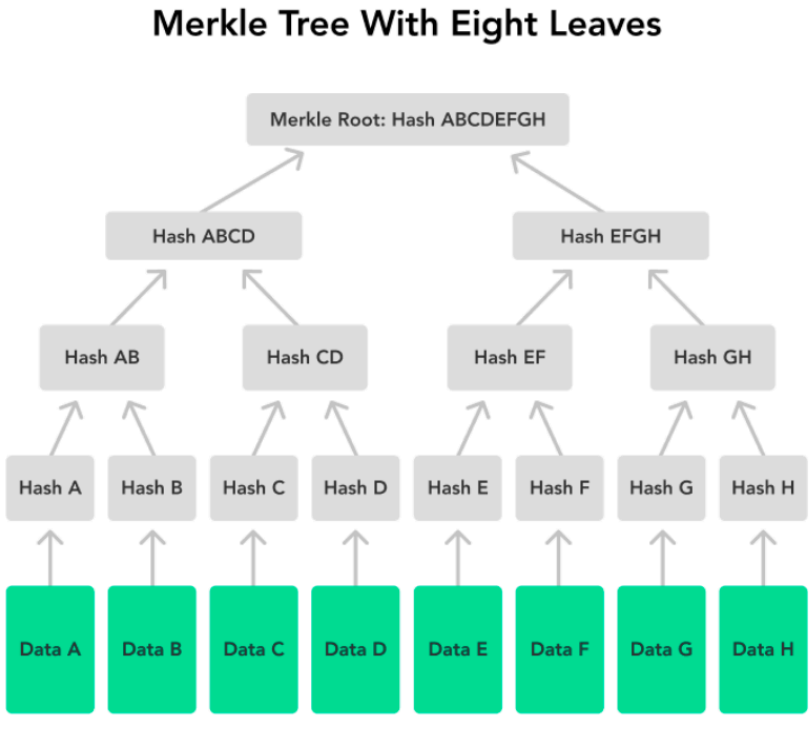

Figure 1.1: Hash Pointers in Blockchain