1.A Blockchain Intuition

1P.A Create a Blockchain

2.A Cryptocurrency Intuition

2.B Cryptocurrency Transaction

2P.A Create a Cryptocurrency.

3.A Smart Contract Intuition

3P.A Create a Smart Contract.

----------------------------------------

**What is a blockchain**

A blockchain is a continuously growing list of records, called blocks which are linked and secured using cryptography.

1. Data
2. Prev Hash
3. Current Hash

First block is called the genesis block. It doesn’t have a previous hash.

Prev Hash :0000 Prev hash: 5TY74g

Hash: 5TY74g Hash: 5UT78m

**SHA256 Hash**

It is the fingerprint for documents. It was developed by NSA (National Security Agency USA).

(Secure Hash Algorithm 256). It is 64 characters long. It is hexa-decimal

Requirements for a hash algorithm (Think of as fingerprint)

1. One way: can’t create the original from the hash
2. Deterministic: Same of content should produce the same hash
3. Fast Computation
4. Avalanche effect: even a small change results in entirely different hash
5. Must withstand Collisions.

**Immutable Ledger:** Making a chain so that it is resilient to attacks. One change in any block means you will have to change the next block, creating a chain reaction. (Which is very difficult).

**Distributed P2P Network:** Copies of blockchain is stored in multiple computers in the network.

**Mining:**

Block #3

Nonce: number used only once

Timestamp

Data

With Multiple transactions

Prev Hash

Hash of block

Whole computation power required is for Nonce. Predict a hash below the target.

**Byzantine Fault Tolerance:**

**Consensus Protocol:** If who computer mines block at the same time, which one should the network accept is solved by this.

Types of Consensus Protocol: Proof-of-Work (PoW), Proof-of-Stake (PoS).

This also checks if someone is not adding a malicious block to the chain

**Proof-of-Work (PoW):**

**Longest Chain is the King:** Who ever adds the new block first wins. The chain that loses creates orphan blocks.

So more hashing power results into higher probability of creating the next block and win.

Notes:

Blockchain is a decentralized immutable ledger that keeps on growing.

**Three types of Decentralization**

When people talk about software decentralization, there are actually *three separate axes* of centralization/decentralization that they may be talking about. While in some cases it is difficult to see how you can have one without the other, in general they are quite independent of each other. The axes are as follows:

* **Architectural (de)centralization**— how many **physical computers** is a system made up of? How many of those computers can it tolerate breaking down at any single time?
* **Political (de)centralization** — how many **individuals or organizations** ultimately control the computers that the system is made up of?
* **Logical (de)centralization**— does the **interface and data structures** that the system presents and maintains look more like a single monolithic object, or an amorphous swarm? One simple heuristic is: if you cut the system in half, including both providers and users, will both halves continue to fully operate as independent units?

Blockchains are politically decentralized (no one controls them) and architecturally decentralized (no infrastructural central point of failure) but they are logically centralized (there is one commonly agreed state and the system behaves like a single computer)

**What is Bitcoin**

3 layers: Technology, Protocol/Coin, Token {Blockchain, Bitcoin not just a coin it’s a protocol, BNB is token created on Ethereum: Smart contract, Bitcoin doesn’t have any token}

Ethereum is protocol and ether are the coin, most of the protocol have their own coin

**Bitcoin’s Monetary Policy**

The Halving: The number of bitcoins released every year is halved every four years, every time a block is created bitcoin is released.

Bitcoins are deflating currency.

Block Frequency: time taken for a new block to appear.

**Mining Difficulty**

Difficulty = current target/ max target.

**Mempool:**  A Mempool is a Storage are where transactions are stored before they are added to a block. Every participant of the P2P distributed network has their own mempool on their computer.

**51% Attack**

Not an attack to taper with a block.

Not illegal, exploits the 6-block confirmation, with more hash rate, they try to create double spent.

The 51% is a group of hidden participants with majority of the computation power conduct mining without announcing their version of chain to the rest of the network. The attacker can benefit from by leveraging the double-spend problem.

**Mining Pool**

**Nonce Range:** 32-bit unsigned number. (0-4 billion), It has a timestamp.

So, changing timestamp helps us to reuse the hash, if it was not so, once the 4 billion range is finished, and it might be possible we don’t find a nonce.

How miners pick the transaction

Before storing them in a block, transactions are stored in mempool.

The ones with maximum fees.

But when mining pool works, they can go through all the numbers from 1 to 4 billion within 1 second. So, they must wait 59 sec for timestamp change. So, they drop one transaction and add new one with a lower fee.

Once the timestamp increase, it will again start will all the max fee transaction.

### **CPUs vs GPUs vs ASICs**

CPUs are general.

GPUs are specific to solve matrix operation. For display.

ASIC: Application specific Integrated circuit. Specialized to calculate SHA256

Ethereum uses eth hash which is memory heavy. So ASIC for bit coin can’t be used for Ethereum.

**Mempool**

Every individual node has a mempool, it’s a staging area. Once a transaction is done it is relayed over to node’s mempool.

**Cryptocurrency Transaction**

**Transactions and UTXOs:** UTXOs (Unspent transaction Output)

No such thing as Account.

So, when we do a transaction, we need to send whole of it.

E.g: if I have {harsh 🡪 0.4 BTC, chinu 🡪 0.2 BTC, rishav 🡪 0.7BTC}

Now is want to spend 0.5 BTC to buy a bike

So only valid transaction is 0.7 BTC 🡪 {0.5 BTC to the shop and - 0.2 BTC back to myself}

**Where do transaction fees come from?**

So, the money coming back to me, a part of it will be transaction fees.

Out 0.2 to 0.16 you get back and 0.04 is used as transaction fees.

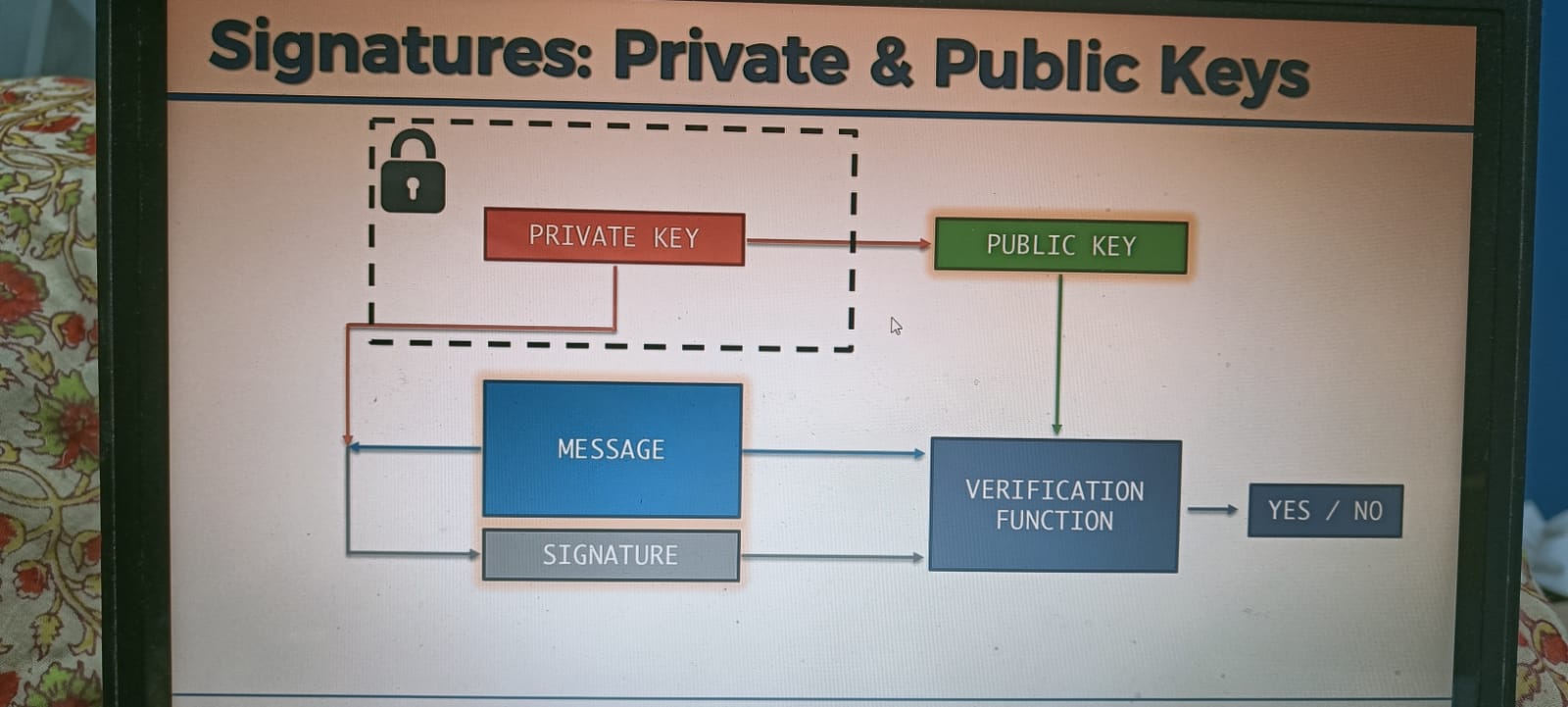
**How Wallet works**

It calculates all the UTXOs and shows you the balance.

**Signature: Private and Public Key**

When you start you are provided a private key, that is my unique identifier. (Password to bank account). From private key we create a public key(which is like my account number).

Message is a transaction, when I create a transaction the private is added to it creating a signature.



**What is segregated Witness (SegWit)?**

Size of block is 1MB in bitcoin, so load caused the network to slow. So, they did a soft fork.

In the transaction, 60% space was occupied with Signature and Public Key, so they separated it and transferred them as separate entity in the network. This allowed double the transaction amount in the network.

**Public key and Bit coin Address**

Public key 🡪 SHA256 we get Bitcoin Address. (Bank account number). Additional layer of security.

**Hierarchically Deterministic Wallets (HD) Wallets**

BIP: Bitcoin Improvement Proposal

Master private key 🡪 Private key 🡪 Public key 🡪 Bitcoin Address

+1 🡪 Private key 🡪 Public key 🡪 Bitcoin Address

+2 🡪 Private key 🡪 Public key 🡪 Bitcoin Address

Can be regenerated as we add deterministic value to master key, so need to worry about keeping all track of private key

**3.A Smart Contract Intuition**

**Etherium:** What if we create a block chain that not only stores transaction, but also store programs. What if we could run Facebook on everybody’s computer rather than hosting it in some server. Build worlds biggest super computer, that will be able to de-centralize any application.

**Smart Contract:** Basically, programs that run on blockchain

Here comes in **solidity:** the script to code in Etherium, bitcoin also has a script🡪 call the bitcoin script, but it’s not turing complete. Solidity is. Bitcoin script misses the loop. LOL. But intentional. If someone writes an infinite loop or a long loop, it will execute in all the nodes destroying the whole network.

Dapp : Contains a frond end and backend(Smart contract). For users to interact.

**Etherium virtual machine and Gas:** when we participate in the node, we get an EVM that encapsulates everything.

How to tackle the issue of infinite loop: For any computation we need to pay gas

DAOs: Decentralized Autonomous Organizations

**Hard Forks and Soft Forks**

Hard Forks: Loosen Rules, not backward compatible

Soft Forks: Tighten Rules, backward compatible

**ICOs (Initial Coin offering)**

**Whitepaper :**