BlockChain & CryptoCurrency

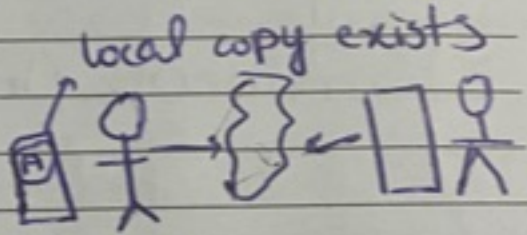
DATE:

from mid1 → mid2 → final.

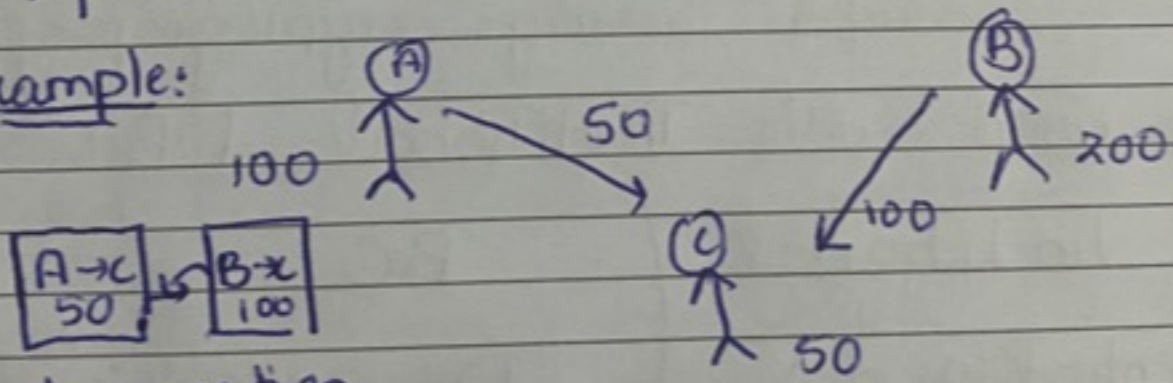
Why BlockChain?

- MicroSoft word → updation One by One in doc
  No simultaneous updation nature
- Google Docs → Shareable, simultaneous nature
  but was centralized, Single point failure

Blockchain
↳ De-centralized
↳ Down-proof
↳ each node has local copy.
↳ fault tolerant



Example:



every transaction
↳ new block

local copy → public ledger

each node will have local copies
any, in uptic updation, updation
all nodes ledgers

DATE:

*Characteristics:

- Open / Accessible to all
- Distributed / De-centralized
- Ledger (Public ledger)(local copy) (DB spread across nodes).
- Efficient (Security, Scalability)
- Verification (Validity by nodes) (Voting)
- Permanent (Immutable) (Tempes proof).

---

* Continuously growing list of records, called blocks which are linked and secured using cryptography

Comparison with Traditional DB:

Traditional	BC
• Centralized	De-centralized
• Security: firewalls access contr	Cryptography immutability
• Needs Trust in central authority	Built in trust

Vulnerable to attacks and manipulation
faster

Immutable
Slower but scalable

Block of a BlockChain

BN # 1	→ Block no
Nonce	→ No of only used once (changeable)
Data	→ Actual
Ph	→ hash of previous block
Hash.	→ hash of current block

↓ First block → Genesis Block

↓ hash value (Digital Signature)

↳ Takes input as (BN, Nonce, Ph, and gives hash

Normally → 64 char hex value (0-9, A-F)
uses different algos
SHA-256, SHA, 512.

(Mining).

DATE:

↳ secure hash algorithm

SHA-256

↳ 256 bits in mem, 64 char

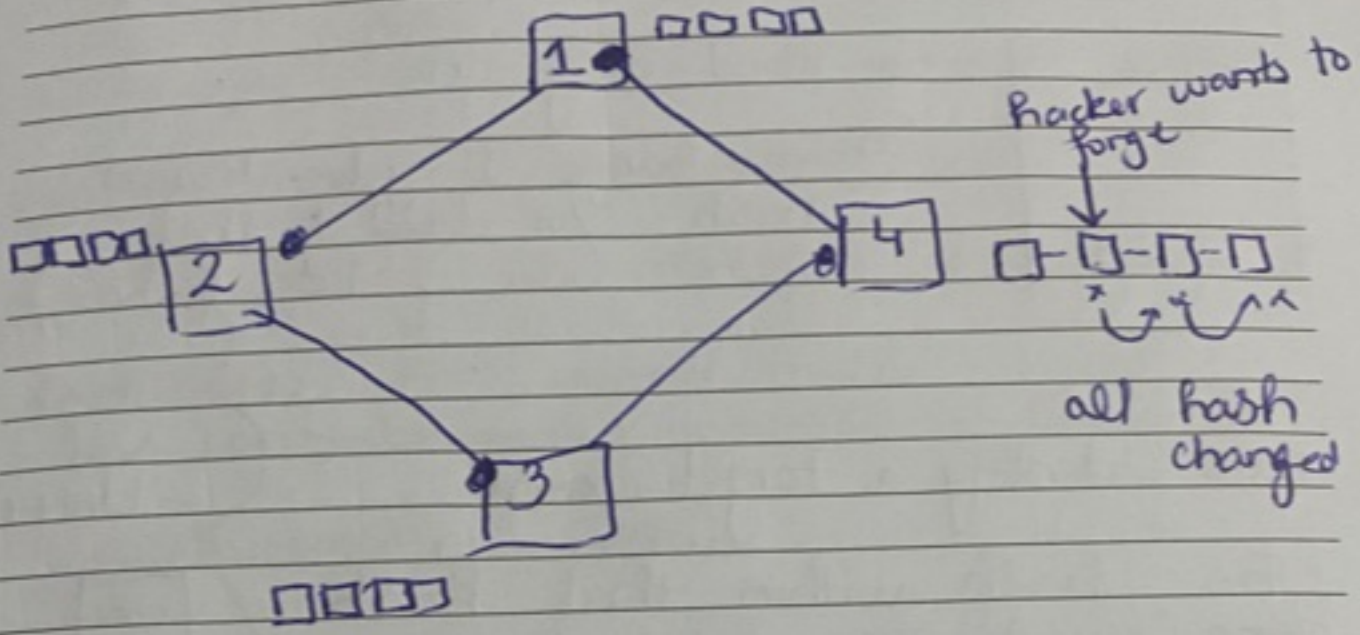
- If we change even 1 character, the whole hash will change

Requirements for a successful hash algo

1) One-way : Impossible to reverse engineer
2) Deterministic : Same output for same input
3) Fast Computation: Efficient for large data
4) Avalanche Effect: Small change in input should lead to vastly different hash
5) Collision resistance:
   ↳ Extremly unlikely for two inputs to get same hash

DATE: _____

Distributed P2P Network



→ Sol: Voting System (Checking blocks after time)

Correct sequence gets copied down into forged block chain due to majority votes, discarged forged block. (Security) ✓✓

Mining

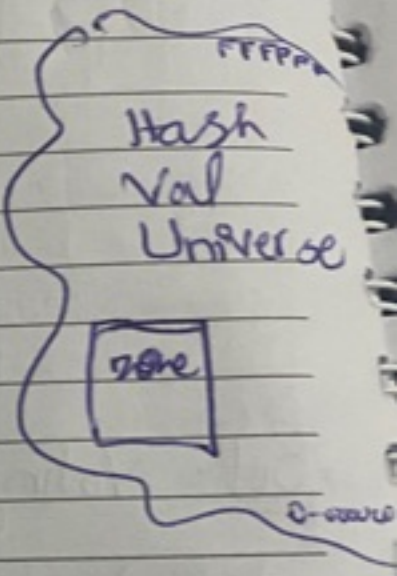
DATE:

How mining works

Nonce

→ no. used only one

↳ can change to get desired hash / or closed to that

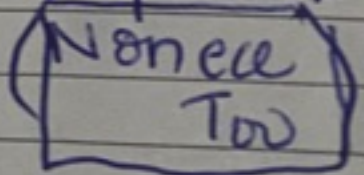


- We identify a target zone
- The hash within that hash zone will be accepted

- Performed by miners
  - ↳ Add new block in BC
  - ↳ Hit & Try
  - ↳ Need high computation power

When you get desired hash, you get reward

(Avalanch effect)
↓
hash value is unpredictable



Satoshi Nakamoto

DATE:

Miners → who validates the transaction and add them to blockchain by solving complex mathematical puzzle.

Nonce addes extra power? how
↳ flexibility as miners can adjust
↳ ensures that miner must invest computational power to find block

Cryptographic Puzzle:

- System generates a hash, miners have to generate any hash equal or less than that system hash

Whoever generates the same hash first, wins & is able to add new block. Winner nonce = Golden Nonce

Byzantine fault Tolerance:

↳ property of a system to remain secure even if some of its nodes behave fail.

Tolerance factor = 33%.
Uses concensus algorithm

DATE:

Tolerance factor: After a certain no. of false votes from a specific node, that node is disabled

Merkle - Damgard Transform: (Tree)
Ralph Merkle 1979

- Used to build collision resistant cryptographic hash function from one way compression func
- Binary tree which stores the collective hashes of child nodes

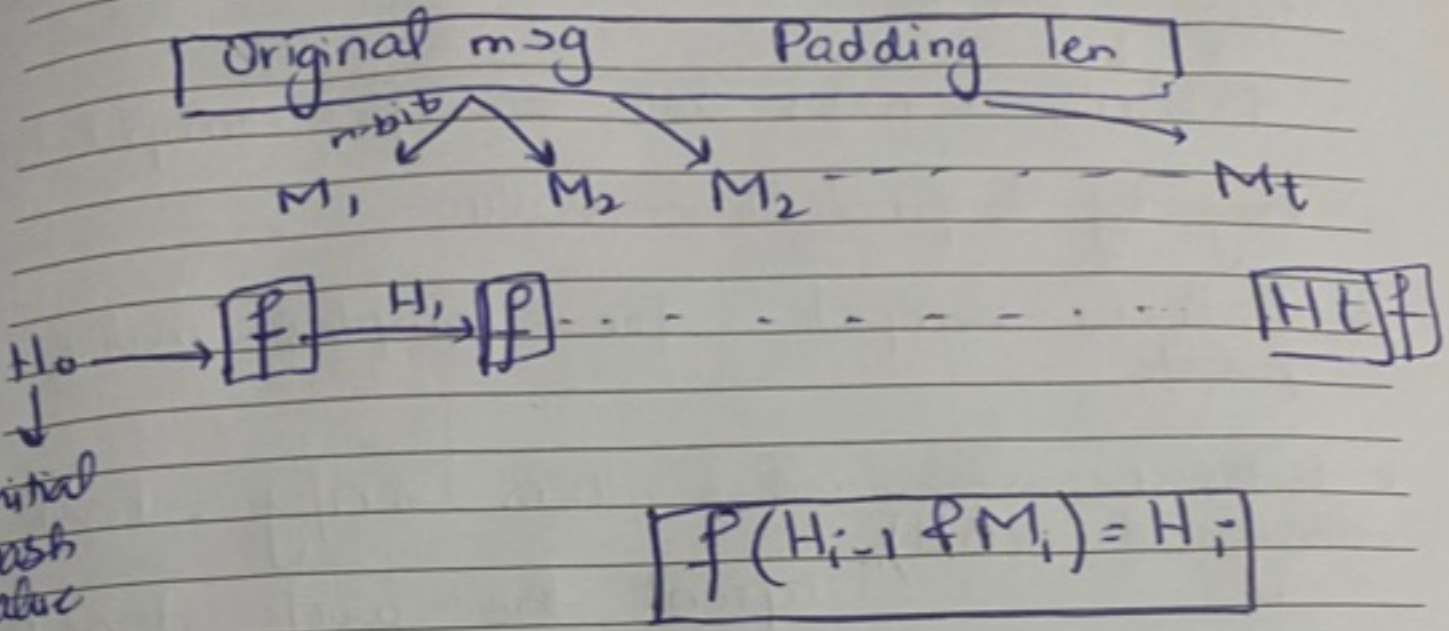
Steps:

① Msg length & padding are appended to message as multiple of (512/1024)

② Msg is then divided into blocks of 'n' bits

③ Compression func at each i^{th} operates on (H_{i-1})(hash) and M_i (msg) to creat (H_i).

DATE:



$$f(H_{i-1} \& M_i) = H_i$$

Consensus Algorithms

- mechanisms that enable distributed systems to achieve agreement on single data value depite presense of faulty nodes

PoW, PoS

↙ Bitcoin

↘ Ethereum 2.0

DATE:

supervision algo

Proof of Work:

Miners → nodes with powerful computation

- They try to find hash value
- Miners compete for complex maths puzzle
- Whoever wins, all others verify, hash assigned (BH). ↳ proof the work done by miner

Rewards: Bitcoins

Competing chain

↳ when the network contains two or more valid chains. The block which losses is called orphan block,

2008

longest chain wins ✓✓

DATE:

Bitcoin → cryptocurrencies
↳ no physical ~~existance~~

→ Decentralized digital currency, working on P2P BC

→ Satoshi Nakamoto

Capped at 21 million coins, digital gold

Uses PoW

→ Halving events
↳ After 4 years, the reward for miners is halfed, reducing the rate at which BTC are created

3 Important layers

→ set of rules guiding how network nodes comm

Tech Protocol/Coin — Blockchain BTC

Token

2009 ✓ 2008 ✗ DATE:

Bitcoin protocol

* Set of rules participants agree on stuff of BTC comm.
* Dictates how the come to consensus
* How public keys & signs used for authentication
* How Agree to upate the protocol itself

Tech	BlockChain					
Coin	Waves	Ethereum	BTC	Neo	Ripple	
Token	WCT BT	TRX AE	X	ACAT, TNC	X	
	WCIR INTC	REP SNT		BC RPX		
		RHOC MKR		QLC, TKY		
		PPT BNB		ONT, IAM		

BTC ecosystem:

- Nodes
- Large Mines
- Miners
- Mining pools

50 25 12.5 6.25
2009 → 2012 → 2016 → 2020 → 2024

DATE:

Monetary Policy:

- Halving
- Block freqeng

Frenquencies

BTC	10 min	4th
ETH	15 sec	2nd
Ripple	3.5 sec	1st
Litecoin	2.5 min	3rd

MID 2:

Concensus: → agreement (Trustworthy) (Secure)

- If transaction is valid or not

10BTC
A → B
X not valid

POW & POS

Proof of work
Proof of stake
} Efficiency, Cost etc | Differences

CAMPUS FRANCE

DATE:

Simplified

- New transactions are broadcasted to all nodes
- Each nodes collect new transaction into a block
- In each round, a random node gets to broadcast its block.
- Other nodes accept the block only if all transactions in it are valid. (unspent, valid signatures)
- Nodes express their acceptance of the block by including its hash in the next block they create

Incentive for miners:

① Miners:

↳ participate in blockchain network

↳ Valides transactions & add them to blockchain

• Network security & integrity

Nodes automatically re-calculate the target every two weeks

DATE: _____

avg time b/w blocks ≈ 10 min

1) Block Reward:

↳ fixed amount of money (bitcoin) to the miner who successfully adds a new block to blockchain

2) Transaction fee:

↳ Those who send transactions can attach a small fee to it.

↳ Miners priortize transaction with higher fees, to maximize the earning

Proof of Work:

energy consumption →

↳ all other nodes check if its valid or not

1) Puzzle solving
2) First to solve (he broadcasts the solution to the network)
3) Verification
4) Block Addition

P1: Difficult to compute

2014, AUG: 10^{20} hashes/blo

Trivial to verify

Nonce must be published as part of block

CAMPUS FRANCE

DATE:

Drawbacks
↳ EC
↳ ASIC hardware gives advantage to miners

If mining reward (block reward + Transfee) > hardware & electric cost → Profit

PoW	PoS
① Miners (block creates)	② Validators
③ Block reward (result)	④ Transaction fee
Heavy equipment required (special units)	No such equirement (standard equipment units)
Solve the puzzle	No need
EConsumption ↑	EC ↓
Computational power	Cryptocurrency

Security

PoW Strength → 51% attack
→ Uncensorable & publically broadcasted

- Proven applicability, predictable block times
- Does not rely on other nodes being trustworthy

DATE:

Current PoW systems

1) BitCoin
2) Ethereum
3) Litecoin
4) BitCoin cash

Mining pools: (Team up)

→ Group of miners who combine their computing power to increase their chances of solving the puzzle

→ Rewards are shared among participating miners based on their power

---

Nonce Range

- 32 bit number

0 ⟶ 4 Billion

Total possible 64 bit = 10^{77}

Total valid hashes = 2×10^{25}

Prob that random hash is valid = 2×10^{-22}

CAMPUS FRANCE

DATE:

Max Nonce = 2^{32} = 4×10^9

No collisions

Probability that one of them will be valid = 10^{-12}

A modest miner does 100MH/s

so 4Billion in 40 sec if he doesnt still found the golden nounce, nao what?

Time Stamp:

BN
Nonce
Timestamp
Data
PH
H

489 1529533115

↳ Unix time

1 Jan 1970

exhaust in terms of time

Nonce will be valid again

DATE: _____

How miners pick Transactions

avg = 100M/H

Block
TS
Nonce
Data
PH
Hash

4B

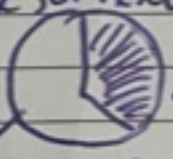
T_1
T_2
T_3
T_4

Transactions

- Best Case → Billion Trillian hashes/s

All Nonce will be exhausted

Time for next timestamp



← All Nonce exhausted

what to do here

We cant stay idle even for fraction of second. Then we change Date

We will change the config.

Changing combo of transaction

Can use None are valid

Mempools

T_1
T_2
T_3
T_4
T_5
T_6
T_7

← Miners pick Transaction from here

← Criteria
↳ Transaction fee

DATE:

Mempool:

↳ Temporary storage area for unconfirmed transaction

CPUs	vs GPUs	vs ASICs
Central processing unit	Graphic processing unit	Application specific integrated circuit
General	Specialist	Totally specialized
less than 10MH/s	less than 1GH/s	> 1000GH/s

Transactions & UTXO's

↓

Unspent Transaction outputs

DATE:

Seperate entities

A → Me (0.4) BTC ✓
B → Me (0.2) BTC ✓
C → Me (0.1) BTC ✓
0.7

} UTXO at your end. Unspend a Bitcoins

Bag (0.4), C (0.25)

I/P	O/P
A → Me (0.4)	Me → BagOwner (0.4)
B → Me (0.2)	Me → C (0.25)
C → Me (0.1)	Me → Me (0.05)
me → me (0.05)	

Using existing UTXO to make new UTXO.

Mempool → have local copy.
UTXO → leftover crypto currency
Waiting Area / Dynamic / Middle Area.

DATE:

input : source of funds
output : destination of funds
secured using digital signatures

BlockChain Wallet: → Digital wallet

- Ease of use, instant transaction, secure

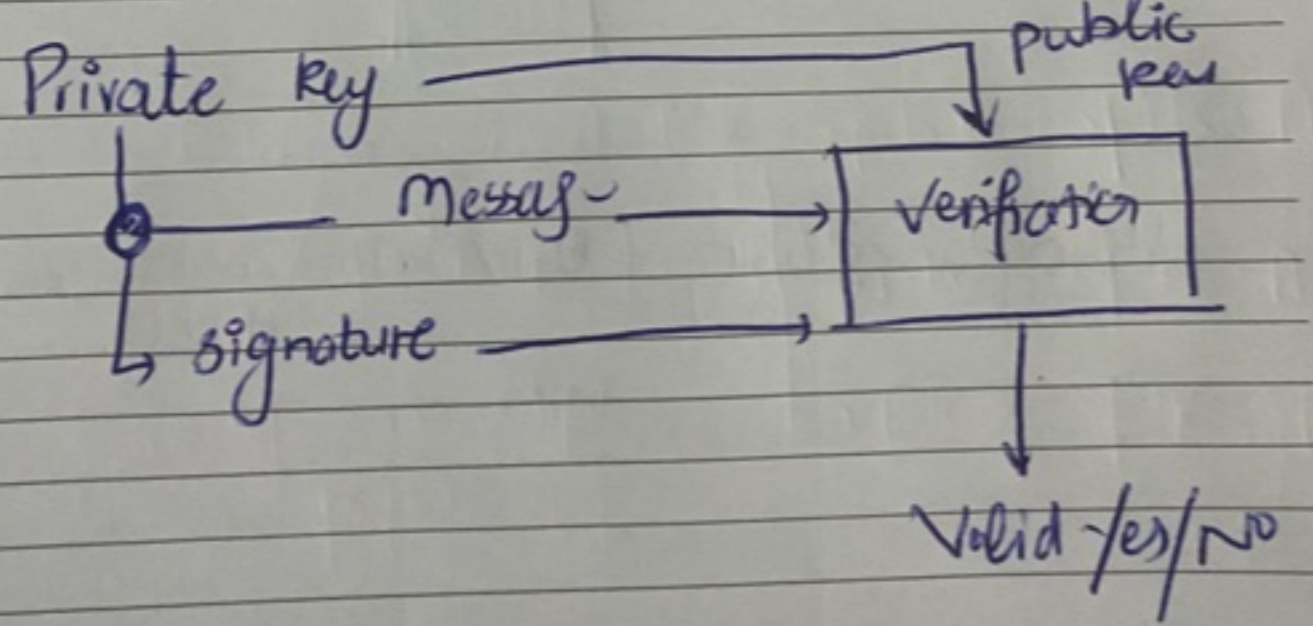
Keys:

Public Keys (Acc # No) → given to everyone

Private Keys (PIN) → you send crypto using this

Store private keys

Transaction can't be done without using private key.



DATE: _____

API for digital signatures

↳ set of functions & protocols that developers can use to incorporate digital signature capabilities in their apps

JCA, JCE

(sk, pk) : generate Keys (key size)

sk = secret key
pk = public key

sig = sign (sk, msg)

isValid : verify (pk, msg, sig)

Bitcoin uses (Assymetr.

ECDSA Standard

Elliptical curve digital sign Algo

CAMPUS FRANCE

DATE:

Decentralized Identity Management

Concept, empowering individuals to have more control over their personal info & digital identity without central authority

---

Hierarchical Deterministic wallets (HD) wallets

Transaction → Public key
Bitcoin accept → Bitcoin address
} pattern issue of Bitcoin address

Solution
↳ Master private key → pK1 → pK → A1 (public)
↳ Secure
↳ Check pK
↳ Give you multiple addresses public key

Desire pK

+1 → PK2 → PK → A2
+2 → PK3 → PK → A3

CAMPUS FRANCE

DATE: B chain ↳ 2008

Master public key
↳ public key generat ,nol for payment

Ethereum: → vitalik Butcin (2013)
↳ programs in blocks ↓19 y/o
not only transactions

Open-source blockchain-based platform
a Decentralized, immuteable ledger
"platform to let other to create/build programs on top of it =>

for Bitcoin
↳ purpose to create a cryptocurrency to eliminate banks and allow us to trade borderless

Ethereum → Smarts contracts would be running for making Dapps

Full Node → Miner (validate verify) stores copy of BC
Archieve Node → complete BC (Historical data)
Light Node → only transactions (store only block header depend on full node)

DATE:

What is a Smart Contract
↳ A program containing if's
↳ Runs on Ethereum blockchain
↳ condition for different stuff.
once program is written it will be executed. (de-centralized)

EOA → externally owned account (when wallet is made need privateKey)
Contract → controlled by contract code
account
↳ deployed on Block on Contract code

Smart Contract:
↳ program prolly in C/py.

Solidity incase of Ethereum
Contains If-else conditions
Turing complete → you code what you think
↳ loops → can be implemented

for Program to run, you will have to pay

DATE:

will be making D-apps - since SC will be running on all apps.

Each Node has

1) History of all Smart contracts
2) History of all transactions
3) Current state of all SC.

D-Apps
↳ Apps run in P2P network

Smart contract + Front end (js, react, etc)
(Backend)
(Presearch, LBRY, D·tu

Properties

↳ Trustworthy (Data)

No-censorship
Can never go down
They pay

CAMPUS FRANCE

DATE:

Security issue

1. Viruses & Access to private files
2. Infinite loop / Heavy computations

Solution

1. Ethereum Virtual Machine (EVM)

▲ What if hacker wrote harmful code of SMC.

↳ Run the network on Virtual Machin

D-apps

A system inside the system.

2. Ethereum Gas

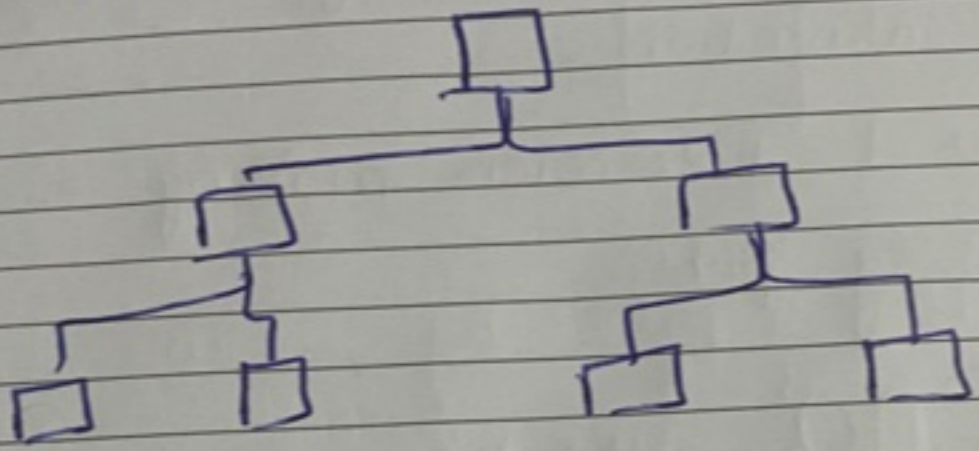
↳ You pay for what your code

↳ your code will stop running when gas runs out

Lecture 19 remaining

DATE:

- DAO's
  - Decentralized automous Organization
    - Different from traditional org (heirarchy).
    - Smart contract like an org



fully democratized
voting required
Transparent & public
Automatically handled

Example:
- NGO donation example

21K units

Gas price = 1 gwei = 10^{-9} ETH

DATE:

Gas Limit / Gas Limit ↳ maximum gas transaction can use — set by sender

DAO:

↳ Different from traditional org.

→ Smart contract as an org.

→ No human intervention
→ No CEO
→ Voting based Decision making.
→ fully public

The DAO attack

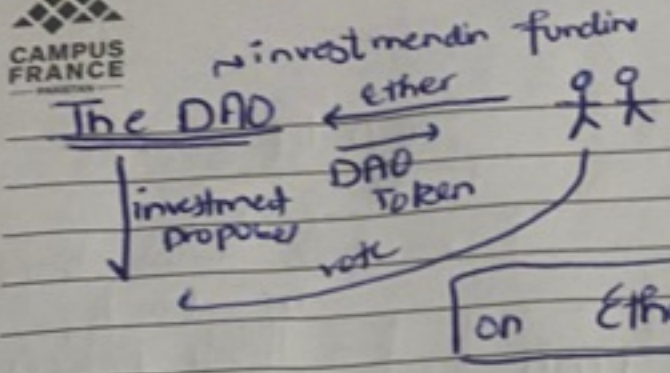
1) 2016 on Ethereum — $150M from 11K investors
vulnerability in DAO code

$60M stolen funds

↙ Hard Fork ↘

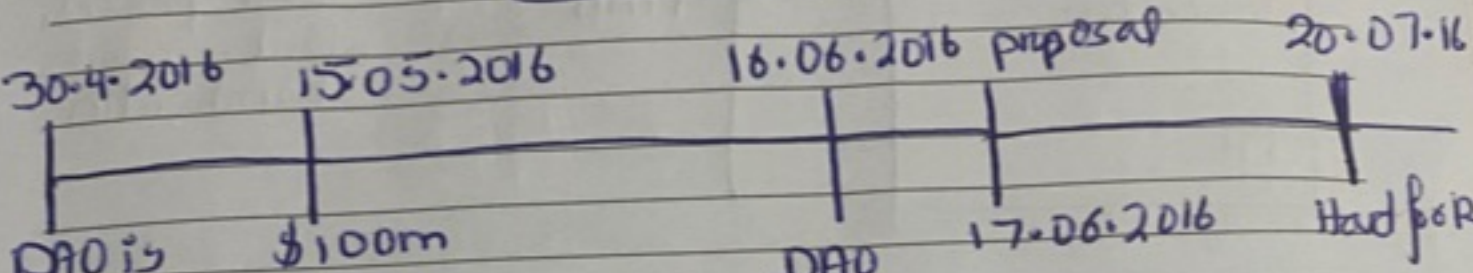
Ethereum classic — Ethereum (Improved)

DATE:



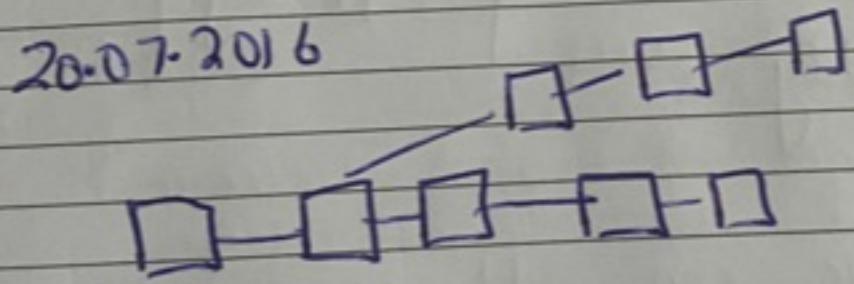
May 2016 crowdfunded 15M

June 2016 → 50M



(issue in code)
5M
60-65M

will be transfer to hacker account



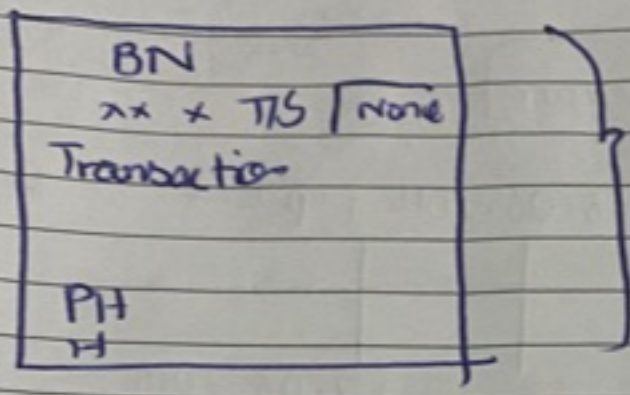
↓
(money returned to owner/DAO)

CAMPUS FRANCE PAKISTAN

DATE:

Segregated witness

Segwit:



more transactions in block, faster process. But more bandwidth

Issue:

Signature & public key contains data

60-65 % jagah/space

Script Sig

bcz

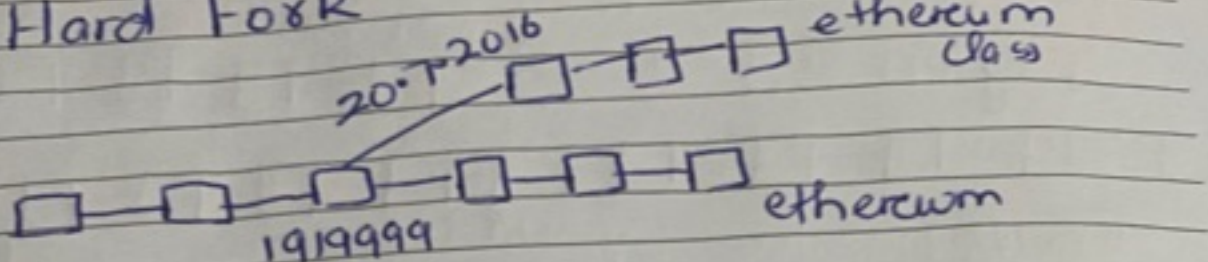
↳ send separately

↓

Now we can add more transaction in the block. Throughput ↑

DATE:

Hard Fork



ETC: Once SMC is deployed, no change is acceptable. No human intervention.

ETH: Upgradation / changes in SMC

During a hardfork, software implementing a protocol and its mining procedure are upgraded

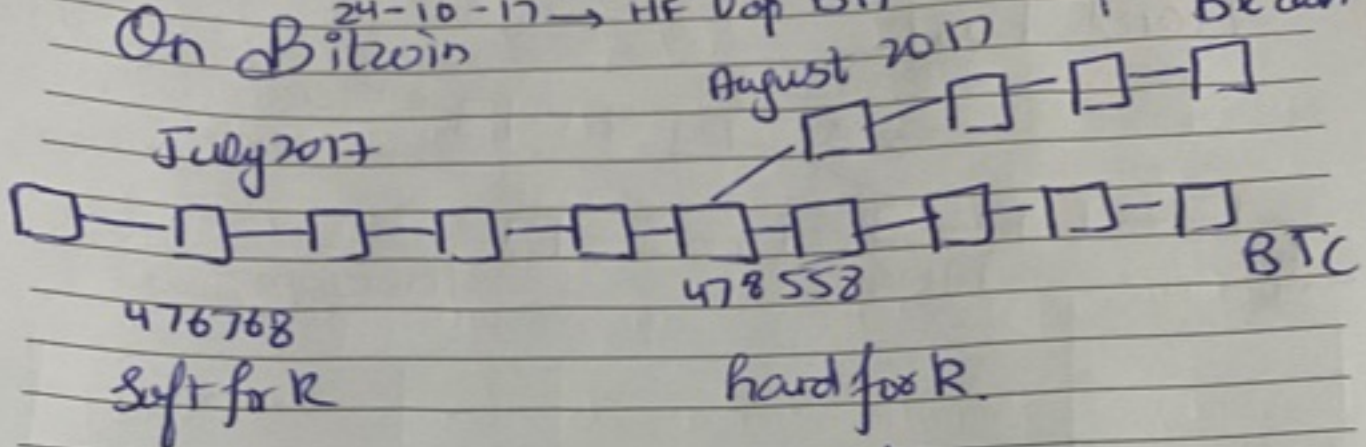
Once user upgrades their software, that version reject all transactions from old software. Creating a new branch.

However those who retain old software continue their progress to transaction.

20-7-16 → HF of Eth
20-7-17 → SF of BTC
1-8-17 → HF of Bit
24-10-17 → HF of Bit

DATE: 8 mb

On Bitcoin



↑ BTcash

Soft forK: → Tightening

↳ change in protocol but end product remains same

↳ Backward compatible upgradte

upgraded nodes communicate with non-upgraded ones

Chain is not divide.

Old nodes could still validate blocks & transactions but they would not understand it.

CAMPUS FRANCE

DATE:

Initial Coin Offering

IPO → Initial public offering

↳ startup company needs money

↓

The company gives shares to the public in exchange of money. Central authority

in ICO: → Digital coupons

↳ you need to build a new coin, you need cryptocurrency so you give tokens in exchange of it

Static Token

↳ Count = fixed

price = fixed

Dynamic Token

↳ may or may not be fixed.

Before launching ICO, you need to fill up the white paper.

↓

Details

CAMPUS FRANCE

DATE:

semantic / Decentralized web
↑

- Blockchain & Web 3.0
  - ↳ BC being decentralized and secure ledger technology, while web 3.0 represents the next gen of web focused on decentralization & enhanced user control centric internet, leverages BC & smart contract.

Ripple:

- ↳ known as XRP
- ↳ prominent & popular crypto currency today
- ↳ Impact on banking sector, imp tech shaping industries
- ↳ Company → OpenChain CEO → Chris Larsen CTO → Jed McCaleb
- ↳ released in 2012
- ↳ written in C++
- ↳ Network can operate without ripple comp

XRP currency code

CAMPUS FRANCE

DATE:

NEO: (onchain) (Chinese ethereum)

↳ released in 2014
↳ designed to build scalable network of decentralized apps

- Green light from chinese govt in August 2017

Litecoin:

↳ in 2011
↳ identical to Bitcoin expect
↳ faster transaction computation
larger no. of coin
↳ Easier to mine with GPU

Second most popular in 2017

Cardano (POS)

↳ 2017, open source smart contract platform
→ multilayer architecture

Stellar:

↳ 2014, NPO stellar.org
→ Open Source (protocol) for value exchange

CAMPUS FRANCE

DATE:

Hyperledger:

↳ open source collaborative effort created to advance cross-industry blockchain technologies

↳ Global collaboration, hosted by Linux foundation

Note:

Hyperledger is a softwar which everyone can use to creat one's own personalized blockchain service

	Bitcoin	Ethereum	Hyperledg
Cryptocurrency	BTC	ETH	None, imple
Network	Public	Public	permissio
Consensus	POW	POW	PBFT
SMC	None	solidity	Yes (Chain
Langs	C++	Go, py	Go, Java

CAMPUS FRANCE

DATE:

Crypto Wallets:

↳ software used to store, send & receive crypto

wallet stores private & public keys

Types:

1) Software wallet
2) Hardware wallet
3) Paper wallet.

Crypto Wallet: (Digital wallet)

↳ have PR, PK.

used on Desktop

Hot wallet

- Desktop wallet
- Mobile wallets
- Online wallet → network
  - ↳ web wallet (runs on accessible cloud)
- Online everyday transaction
- Free/easy
- Online / Hacking

Cold Wallet

↳ paid, offline, secure

↳ Ledger.

Hardware wallet

↳ physical/electronic device

↳ secured/more used

Paper Wallet

↳ Address & private key

CAMPUS FRANCE PAKISTAN

DATE:

Wallet ↳ stores public & private Keys which allow shareable transactions send

3 Types

→ Software wallets (online based) binance, coinbase

→ Hardware (USB) unplug & connect to net. (Ledger)
↓ expensive

→ Paper (QR, pkey, PKEY, on paper) ↳ lose, destroy), most secure. least used.

Hot wallet (software)
↓
Connected to internet
↳ vulnerable to net/hackers

Soft. wallet (Hardware, app)
Computer-connected
or
written
↳
losing, forgetting