END-TERM REPORT: BLOCKCHAIN TECHNOLOGY

Nandanmanjunath I 22B0920

Abstract

The topic I selected is blockchain technology and how this technology can be used in Bitcoin. I learnt these topics through the CS765 lecture video and have completed listening to 20 lecture videos till now.
I got a bigger idea on the blockchain concept.

Contents

1 WHAT IS BITCOIN

BITCOIN is a type of currency, a digital currency which came after 2009 which was invented Satoshi Nakomoto. Satoshi Nakamoto didn't like the idea of central government having control over monetary because of bad decisions made by government in the past which lead to inflation,Bubbles e.t.c.
Bitcoin currency system is decentralized that is all the members have control over the currency.

2 WHAT IS BLOCK CHAIN AND WHY IS IT USED

I will explain through example and its properties also

Case 1: Suppose a company need to verify your degree certificates. The best way is through offline verification when interview is taken offline and certificate is also checked offline for validity. What if verification should be done online// One way is going through the official website and verifying based on his name being in the list of graduates in a particular batch and department.

But is this legitimate?
One can easily bribe the person who maintains the website to add his name to the list. This is one of the problems that blockchain can solve. In blockchain, once information is entered into a block, it can't be removed or modified. Storing information in blockchain, it is difficult(Almost impossible to tamper the data), and it builds trust on data

Case 2: Suppose a person needs to buy property from another person. This can be done by exchanging the property documents and money.
In this case, the information should be known whether the person who is selling really owns the property. This can be checked by seeing when this property was bought by this person and checking in later years whether he sold his property or not.

The problem with normal currency is that both transactions cannot occur simultaneously. It is possible that after handing over the property, the other person might refuse to pay and vice-versa.
But by using blockchain technology, both transactions can occur spontaneously(Atomic transaction)

Case 3: Suppose a person has to send money to another person. Suppose this is an international transaction. In this case, many banks will be involved, and it is transferred through multiple banks to reach the other person finally. In this case, a lot of transaction fee is taken, and also so much time is taken.
But by using Blockchain, transactions can occur very fast, and it will eliminate all of the middlemen.

HOW?
All banks maintain some block-chain(A local copy of their own)and share a single block-chain containing multiple blocks. When a user creates a transaction, this transaction is broadcasted and all banks can view this transaction.Then all banks agree on the block where the transaction is placed. This process is called *Consensus* and after consensus this block cannot be changed permanently and new transactions will be placed in next block.

3 CURRENCY SYSTEM

Here I will be explaining about the olden days currency system and what were its problems and how they updated making better versions. Before starting I will once list all the basic requirements a currency should have

- **Acceptability** Should be accepted by everyone.
- **Durability** Should be lasted longer.
- **Divisibility** Should be easily divisible
- **Fungility** One unit of substance can be exchanged with any other 1 unit of substance.
- **Portability** Can be easily carried.

3.1 Barter System

This was the earliest method. The barter system is when people trade goods or services directly without using money. The problems with bartering were that you needed to find someone who wanted what you had and had what you wanted, there was no agreed-upon value for things, it was hard to divide or move certain items.

3.2 Coins

It built trust for money. They were made with gold and other precious metals. This became the basis unit of transfer. But in Roman Empire when the government needed money, they started to dilate the percentage of gold in coins to make more coins which led to inflation, and it was one of the reasons why the empire collapsed.

3.3 Notes

Newton observed that wear and tear occured to the coins (clipping) and the value of the coins were decreasing. So they decided to print notes(paper) and each note was equivalent to particular quantity of gold and all this gold is stored some where in reserve and instead of directly exchanging gold notes were exchanged.
But in WW1 after germany lost it hd to pay a lot of money to others. So government started printing more money than the gold they had and which lead to inflation(sky-rocketing of prices) and prices went upto 10^{12} times.

After WW2 Bretton Woods agreement took place. Us government had a large reserve of gold and they made an agreement than 35$ is one ounce of gold and they will be circulating 2.5 times the resrve they had.This system will fail if all people come together at once to withdraw gold in exchange of money.

Later during Vietnam War US government need more money and they started printing more money and this created tension to neighbouring countries as the reserve of gold in US is being completely utilised. Many countries decided to return back all the US dollars they had in exchange of Gold.
Currently all countries are using FIAT currency i.e. any ammount of currency can be created by government.
MMT(Modern monetary theory) includes FIAT and setting interest rates by central bank.

Let us come to the basic Question again. **Why currency has value** It has value because it satisfies all the 5 above mentioned properties. But the current currency has many problems. What about alternatives. Bitcoin is one of the alternative. The benifits are very fast transactions can occur and it is a decentralized currency system.

4 Basic Terms to be known

4.1 Inflation

Inflation is when prices of goods and services generally go up over time, causing the value of money to decrease. It's measured by tracking changes in price indexes. Factors like increased costs and changes in supply and demand contribute to inflation. Moderate inflation can be beneficial for economic growth, but high inflation can harm the economy and people's standard of living.

4.2 Bubbles

It is a situation where the prices of certain assets, such as stocks, real estate, or cryptocurrencies, become significantly inflated, detached from their underlying intrinsic value. This price surge is often driven by speculative buying and market optimism, rather than the fundamental value of the asset. As the bubble grows, more investors join in, expecting prices to continue rising and hoping to make a profit. However, bubbles are unsustainable, and eventually, the prices reach a point where they can no longer be justified, leading to a sharp decline or crash in prices, often resulting in significant financial losses for investors.

4.3 Proof of work

Proof of Work (PoW) is a consensus mechanism used in blockchain networks, such as Bitcoin, to validate and secure transactions. It involves miners competing to solve complex mathematical problems to add new blocks of transactions to the blockchain. The process requires significant computational power and energy consumption.
To participate in PoW, miners must invest computational resources and solve a mathematical puzzle by repeatedly guessing a solution until they find one that satisfies the required criteria. The solution, known as the "proof," is then shared with the network and verified by other nodes. Once verified, the miner's block is added to the blockchain, and they receive a reward in the form of newly minted cryptocurrency(Bitcoin).

4.4 Demand-Supply

Demand refers to the quantity of a product or service that consumers are willing to purchase at various price levels, assuming other factors such as income and preferences remain constant. It is influenced by factors such as price, consumer preferences, income levels, availability of substitutes, and overall economic conditions. Generally, as prices decrease, the quantity demanded tends to increase, while higher prices lead to a decrease in demand.

Supply, on the other hand, represents the quantity of a good or service that producers or suppliers are willing and able to offer for sale at various price levels. It is influenced by factors such as production costs, technological advancements,

resource availability, and market conditions. As prices rise, producers have an incentive to supply more of the product, leading to an increase in supply.

The interaction between demand and supply sets the equilibrium price and quantity in a market. When demand exceeds supply, there is a shortage, and prices tend to rise. Conversely, when supply exceeds demand, there is a surplus, and prices tend to fall. The equilibrium occurs when the quantity demanded equals the quantity supplied, resulting in a stable market price.

5 What are required for functioning of Bitcoin

The basic things for bitcoin to work are

- **DEMAND/SUPPLY**
- **Double spend** no one can use the same money more than one time.
- **Anonymity**All are connected to some network where they interact and each one knows how many bitcoins others have(decentralisation) but they don't know who in real life has those bitcoins. each person is identified by his public private key.
- Hash functions, P2P(peer to peer network), Public private key, Proof-of-work. I will explain all these in detail.

6 Peer to peer network

I will explain different type of networks with example.

Case 1: consider a case where all the people in a network are connected to a central server. Suppose a person A in this network has a songA and a person B wants that song.In this network first SongA must be sent to server and then to person B. But in this network many companies faed losses as A was directly selling the song to B.

Case 2: A Distributed Hash Table (DHT) is a way for computers in a network to find and share information with each other without relying on a central server. It works by dividing data into smaller parts and assigning unique labels to them. Each computer in the network keeps track of certain parts of the data based on their labels. When a computer needs to find or store data, it asks the other computers that are responsible for those specific parts. This helps distribute the workload and makes the network more reliable and efficient.
Suppose if there are N nodes in a P2P network then it takes log(N) steps time order to find a particular file from the network. It initially ask its neighbours and if it is not able to find it asks its next nearest tables and so on until it finds the file in an network.

Bitcoins peer-to-peer network

When a user initiates a Bitcoin transaction, it is broadcasted to all the nodes in the network. The nodes then validate and verify the transaction using consensus mechanisms like Proof of Work (PoW) .Once the transaction is confirmed by the network, it is added to a block, forming a part of the blockchain.

In a P2P network, each node has a copy of the blockchain, which contains a record of all Bitcoin transactions. This ensures transparency and prevents a single point of failure. If a node goes offline, the network remains functional as other nodes continue to operate.

The P2P nature of Bitcoin also extends to other aspects, such as the mining process. Miners, who are nodes with powerful computers, compete to solve complex mathematical problems to validate transactions and add new blocks to the blockchain. This decentralized mining process helps secure the network and maintain the integrity of the blockchain.

7 Structure of a blockchain

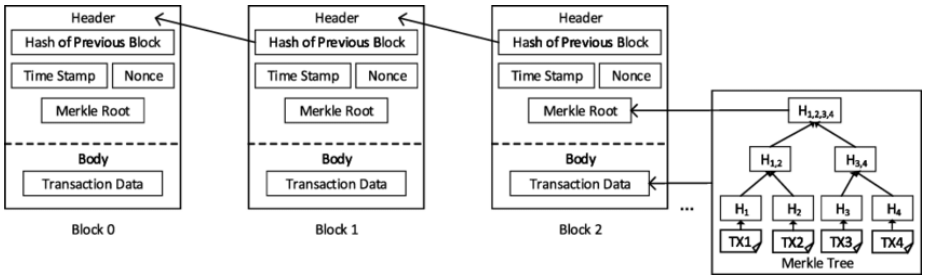

Figure 1: Structure of block chain

A blockchain is composed of multiple blocks. Each block typically contains a list of transactions or data, along with a unique identifier called a hash. The hash is generated based on the contents of the block and serves as a digital fingerprint, ensuring the integrity and immutability of the block. Each block in the chain contains a reference, known as a hash pointer, to the previous block in the chain. This creates a sequential link between the blocks, forming a chain-like structure. I will explain all these parts.

8 Hash Function

It is a function which on giving an input gives an output of fixed size. There are many types of hash functions but they should satisfy certain conditions in order to be used.

8.1 Properties of hash function

- Input is of any size
- Output is of fixed size
- Output is efficiently computed

8.2 Additional Properties

8.2.1 Collision resistance

A function (H) is collision resistant if it is computationally infeasible to find two inputs x_1 and x_2, which are unequal such that H(x_1)=H(x_2) Computationally infeasible means finding two such inputs takes so much time and resources.

8.2.2 Hiding

A hash function H is said to be hiding if when a secret value r is chosen from a probability distribution that has high min-entropy, then, given H(r||x), it is infeasible to find x. Here r||x stands for r concatenated with x. This can be used such that r is a key and x is a message and this can be used to encrypt the message.

high min-entropy: means that the distribution is very spread out, so that no particular value is chosen with more than negligible probability. min-entropy is defined as max of $-\log_2 P(r_i)$ where $P(r_i)$ represents probability of getting r_i from a probability distribution

8.2.3 Puzzle Friendliness

A hash function H is said to be puzzle friendly if for every possible n-bit output value y, if k is chosen from a distribution with high min-entropy, then it is infeasible to find x such that H(k||x) = y, in time significantly less than 2^n
If a hash function is puzzle friendly, then there is no solving strategy for this type of puzzle that is much better than trying random values of x

Proof-of-work
NONCE is a term that stands for "number used once." It is a random or arbitrary value that is used only once in a specific cryptographic operation, such as the mining process in a blockchain.
The Hash function will run until the output of the hash function will be less than some value known as the threshold value. The hash function can be changed by changing input by adding nonce to it and the hash function will run until the condition is satisfied

9 Working of Hash Function and Merkel Tree

Let the input given to the hash function be M bits.

It divides the input into groups of 512 bits. Each group of 512 bits is passed into a compressor function which has two inputs, initialisation vector and the 512 bit message and the output is 256 bit message. This output is again send into the compressor function with the 512 bit message be next part of the message and initialization vector be output obtained before. This compressor function will run until the complete message is sent through the compressor function.

What if the size of the message is not a multiple of 512. Then it will add 1 followed by zeroes till end and in next 512 bit group it will represent the message size.
Due to this way of sending message the input for all the possible different messages will be different and no two messages will have same output.

9.1 Merkel Tree

Suppose the message contains transaction information in it. Suppose we place all the transactions in the message side by side and the hash function is applied to this message. What if some information is wrong in it and we have to add or edit or remove some transaction? If we place all the information side-by-side we have to apply the hash function again to this new message even if there is a small change and it is time-consuming.
Merkel Tree is a type of data structure and by using this it reduces the time to find the output of a hash function if some data is changed
In a Merkel tree, data is organized in a hierarchical manner, where each level

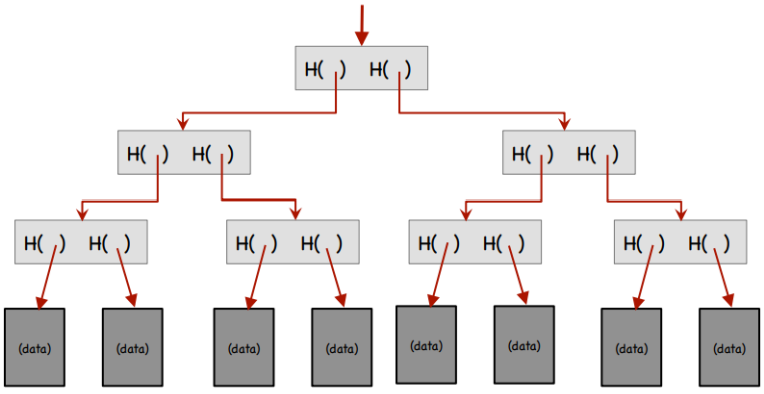

Figure 2: Merkel tree

consists of nodes that are formed by hashing the data from the level below. The bottom level of the tree contains the actual data blocks, such as individual transactions or pieces of data.
If there is a change in a data it take O(logN) to compute data instead of O(N)
The key advantage of a Merkle tree is its ability to efficiently verify the integrity of specific data within the tree. By comparing the hash of a specific data block with the corresponding hashes in the tree, it is possible to verify if the data block is unchanged or has been tampered with. This is accomplished by traversing the tree from the leaf nodes to the root, combining and hashing pairs of nodes along the path.

10 Block-Header

It shows what information is stored in a block

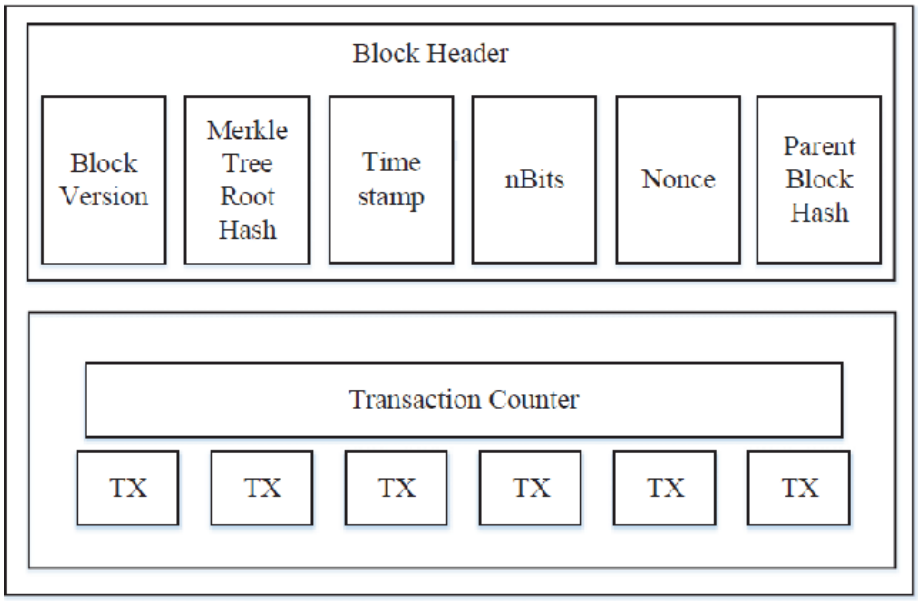

Figure 3: Block-Header

10.1 Bits

It represents the threshold value. It contains 4 bytes b_1 b_2 b_3 b_4 and threshold is given by $b_2 b_3 b_4 * 256^{b_1 - 3}$

10.2 Parent Block Hash

Output of the Hash function from the previous block.

10.3 Timestamp

Represents time taken to find a nonce value such that the output of hash function is less than threshold value.

Light Clients store only the value of block header.

11 Digital signature

A signature should be such that it is easy to create and verify but hard to forge Digital Signature Scheme consists of 3 algorithms:

- (sk,pk) := generateKeys(keysize) generates a key pair
- sk is secret key, used to sign messages
- pk is public verification key, given to anybody
- sig := sign(sk, msg) outputs signature for msg with key sk.
- verify(pk,msg,sig) returns true if signature is valid and false otherwise.

ECDSA (Elliptic Curve Digital Signature Algorithm) is a type of algorithm which generates public privare keys.

12 Bitcoin Transaction structure

Every block in the Bitcoin blockchain contains transactions made by the users. Every transaction records the transfer of Bitcoin from one entity/person to another. We can use these records to trace every Bitcoin to its point of creation. Inputs are the coins the user owns and wants to transfer to another person/entity. Every input has essential subfields to locate and unlock that coin in the Bitcoin blockchain. These fields are as follows:

Previous transaction ID: This is the ID of the transaction in which that Bitcoin was created as output and assigned to the current owner.

Previous transaction index: Every Bitcoin transaction can have multiple outputs (an array of outputs), and every output is identified by a unique index (index of the array). This index, along with the previous transaction ID, can be used to locate the transaction where that Bitcoin was created and to identify its real owner

Outputs are the newly generated Bitcoins locked to the hash of the public key of the payee.
Based on the input value number of Bitcoins a person has can be determined and the output bitcoins is less than or equal to input.
Multiple outputs can be generated from inputs.
Suppose a transaction of 3 bitcoins is done from A to B and A has 3.5 bitcoins. then the sum of inputs will be 3.5, and output contains 3 bitcoins transferred to B, and the remaining balance added back to the public key of A itself.

Transaction Propagation: When a user initiates a transaction, it is broadcasted to the network of nodes. The transaction contains the sender's address, the recipient's address, and the transaction amount.

Transaction Verification: Miners, who are responsible for validating and adding transactions to the blockchain, receive the transaction and verify its validity. They check if the sender has sufficient funds to cover the transaction and if the transaction adheres to the protocol rules.

Inclusion in the Mempool: Valid transactions are stored in the mempool, a temporary storage area for pending transactions. The mempool acts as a waiting area where transactions await confirmation.

Confirmation and Block Addition: Miners select transactions from the mempool to include in the next block they aim to add to the blockchain. They bundle several transactions together, perform proof-of-work calculations, and create a block.

Consensus and Block Validation: Miners propagate their newly created block to the network. Other nodes validate the block by ensuring that the transactions within it are valid and have not been previously spent. They also check if the proof-of-work calculations are correct.

Chain Extension: If a miner successfully validates and adds a block to the blockchain, it becomes a part of the longest chain. If all the transactions are valid then only the block will become valid and the corresponding chain will become valid. If a chain contains an invalid block complete blockchain becomes invalid. The other nodes in the network accept the longest chain as the valid version of the blockchain.

If an attacker attempts to double spend by creating conflicting transactions, the network will only accept one of them based on the longest chain principle. The rejected transaction will be considered invalid, preventing *double spending*.

12.1 Transaction Fee

It is the difference between the sum of inputs and the sum of outputs. To who is it paid. It is paid to the minor who creates a new block. creating a new block needs to solve the problem that is finding nonce to satisfy the threshold condition and this requires computers and electricity is required.

13 Confirmation Time

Time is taken for a transaction to get confirmed. This is important because from the same UTXO two transactions can be shown and whichever gets first into the blockchain is a valid transaction and the rest are not from the same UTXO. If a fork occurs transaction which is in the longest chain is a valid transaction. In order to confirm whether a person is getting his money he/she has to wait

until **n** blocks are passed after the block with the transaction.
More the value of **n** more he/she is sure that the transaction is valid.

13.1 memorylessness

memoryless means that whether an event has just occurred or that an event hasn't occurred in a long time will give us no clue about the likelihood that another event will occur soon. In comparison if we wait for a block to be generated after n mininutes the probability of forming the block doesn't change and remains the same. This property is present for Poisson distribution.

13.2 Attacker

An attacker tries to create a private blockchain with other transactions in order to try double spend and soon when his block gets longer he broadcasts his blocks and makes the other transaction invalid.

Let **p** and **q** be the fraction of hashing power of the attacker and honest miners. For the attacker to win eventually, **p** must be greater than **q**(skipping calculations to prove this)

$$p + q = 1$$

If **p** is greater than 0.5 the attacker will definitely win eventually.

14 Meni Rosenfeld's analysis

Assume network delay to be zero.
Let **z** is the number of blocks the honest chain is ahead of the attacker chain. Consider a block winning by an honest minor or attacker to be like an event of heads or tails with the probability of head **p** and tail **q**. Let s be the number of times head wins.

$$P_{z,s} = Prob(Honestaheadbyz/numberofhonestblocksiss) =$$

$$Q_z = Prob(Attackercatchesupgivenheisaheadbyz)$$

$$R_s = Prob(Attackercatchesupgivenhonestblocksisz)$$

$$R_s = \sum_{z=-\infty}^{s} P_{z,s} \cdot Q_z$$

On solving

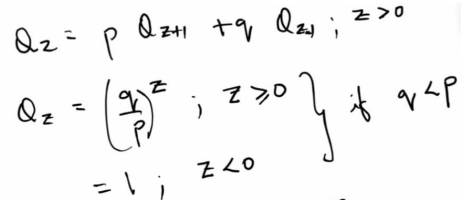

Figure 4: Solving equations

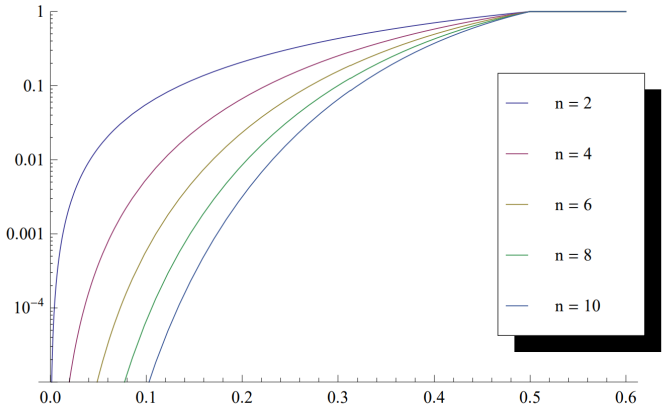

Figure 5: Graphs of R_s (here n is s)

As s increases R_s values decreases therefore the more we wait the less is the chance of double spend.

15 Selfish Mining

DON'T RELEASE THE BLOCK IMMEDIATELY; CREATE PRIVATE CHAIN

Initially, the forked blockchain will be shorter than the public blockchain. The private chain mines new blocks within its pool and hides any newly-generated blocks. The mining process is repeated until the private blockchain reaches a block height greater than that of the public blockchain.
There are many strategies such as release the chain when they attacker is 1 block, 2 blocks ... ahead of honest blockchain and many other techniques.
I will not be explaining the detailed math and graphs behind these strategies but I will explain some basic terms behind the strategy.

$\gamma_{pool} = Attackers\ block\ which\ entered\ the\ final\ chain/Total\ number\ of\ blocks\ generated$

$$\gamma_{pool} = \frac{A}{A_0 + H_0}$$

$$\gamma_{other} = \frac{H}{A_0 + H_0}$$

$\alpha = fraction\ of\ hashing\ power\ of\ attacker$
$\gamma = fraction\ of\ honest\ minors\ who\ attack\ attackers\ block.$

selfish mining gives more profit and more rewards to minor compared to honest mining.
selfish mining is more profitable than honest mining; multiple selfish miners or groups on a network would create a race between the forks and reduce profitability

16 Liveness and safety

Analogy of bank
In bank whenever you deposit cash A then the amount of cash used by the bank to circulate is A*CRR where CRR is cash reserve ratio.
transactions are accepted as long as bank runs.
If all people come together at once to withdraw their money then banks collapse.

16.1 Liveness

Liveness refers to the property that guarantees the continued progress of the blockchain network, meaning that the system can make progress and process transactions over time. Liveness ensures that the network remains active and doesn't get stuck or become non-functional. In the above case as long as bank is open it accepts transactions. In the case of Bitcoin, liveness means that new transactions can be added to the blockchain, and new blocks can be mined by miners. For example, when you send a Bitcoin transaction, you expect it to be eventually included in a block and confirmed by the network's miners. If the network is experiencing liveness problems, transactions might not get confirmed, and the system could become unresponsive.

16.2 Safety

Safety, on the other hand, refers to the property that guarantees the validity and integrity of the blockchain. It ensures that a transaction cannot be altered or reversed once it is confirmed and added to the blockchain. In other words, safety ensures that the information stored in the blockchain is accurate and trustworthy. In the above case, as long as the bank runs, it doesn't get bankrupted, and the transactions are updated correctly so that the amount in the bank account doesn't vanish or increase due to errors. In the case of Bitcoin, safety means that once a block is added to the blockchain and has a sufficient number of confirmations (additional blocks built on top of it), the transactions

contained in that block are considered final and irreversible. This is crucial for preventing double-spending and maintaining the entire system's integrity.

16.3 Why should miners be honest

There are a set of rules in order for a block in the blockchain to be valid. If an invalid block is added without following all the rules, the block and the blocks which are attached to it will become invalid.
Even though it becomes invalid, why is it a problem? In order for a block to be added to Proof of work must be done, and it takes so much time and consumes a lot of electricity and hardware is required. If the block is valid, the miner will be awarded a mining fee, which covers the electricity bill and gives profit to minor. The minor will be paid with bitcoins whose conversion will be more than the electricity bill.
Bitcoin consumes a lot of amount of electricity and therefore it is difficult for mining and common people cannot do mining. Consuming a lot of electricity is also a problem of Proof of work and people are searching for an alternative inorder to reduce power consumption.

17 Proof-of-stake

Proof-of-stake is an alternative to Proof=of-work it has some advantages like less power consumption, but it also has other types of problems. As in the case of Proof of Work (PoW) where miners compete to solve complex mathematical puzzles, PoS relies on validators who are selected to create new blocks and validate transactions based on the amount of cryptocurrency they hold and are willing to stake as collateral.

In a PoS network, participants are known as validators. To become a validator, a user must lock up a certain amount of the network's native cryptocurrency as a stake or collateral. This stake guarantees that the validator will act honestly because if they validate fraudulent transactions or attempt to attack the network, they risk losing a portion or all of their staked coins.
The validators are randomly selected randomly and deterministically and they validate a block and the validators are awarded mining fees where as in POW miners constantly mine and reward is based on a competition of who is able to create a block using POW.
If a validator does a mistake then the stake of coins will be taken as fine which is a loss for the miner.
You need to own enough coins or tokens to become a validator on a PoS blockchain. For PoW, miners must invest in processing equipment and bear hefty energy charges to power the machines attempting to solve the computations.
Eventhough there is less power consumption there are other problems like

17.1 Nothing at stake

Vadilators can create as many blocks as possible because there is no cost of making one which also leads to multiple forks in the blockchain. In order to stop this people are penalised in attempt of generating forks.

17.2 Long range attack

In a Long Range Attack, the attacker aims to create a longer chain (a chain with more accumulated blocks) than the current main chain. Once the attacker has built this longer chain in secret, they would reveal it to the network, causing honest nodes to consider the attacker's chain as the valid one due to its length. This attack can be especially problematic in PoS systems because validators are chosen based on their stake in the network, and attacker with an initial stake increases his balance by creating many blocks and gain reward for creating the block, as balance increases his stake increases and creating next blocks become faster and eventually his/her chain can become longer than the honest chain.

17.3 Initial Distribution Problem

A highly concentrated initial distribution could result in a few large stakeholders having significant control and influence over the network. This can lead to centralization of power, which goes against the decentralized principles that blockchain technology aims to achieve.
people with a higher stake has a higher chance of producing the block.To increase their stake, people may prefer not to spend their money, which is not suitable for a system.

17.4 Bribe attack

People may bribe other people to create blocks in their private chain to increase their chain length and this may cause loss to other honest people.

17.5 Pre-computing Attack

A pre-computing attack in the context of Proof of Stake (PoS) is an attack where an attacker attempts to pre-calculate or pre-mine a significant portion of the blockchain's stake before the network's official launch or before they become public validators.
By exploiting the lack of randomness in the slot leader election process, a slot leader can manipulate the frequency of them being elected in subsequent blocks. This issue can be solved by enforcing randomness to the process and minimising or even eliminating influence factors controlled by the validators.

17.6 Alternate Solution 1 : POW + POS

It solves electricity problems and also has the benefits of POW.
Let the mining fee per block be M and avg time between two blocks be T
Full POW: Energy cost per unit time $< \frac{M}{T}$
Alternate solution; POW after N-1 blocks of POS.
Energy cost per unit time $< \frac{M}{NT}$
In this method, Nothing at stake problem reduces, but the short-term nothing-at-stake problem can occur (lasts for N-1 blocks at max)
Long-range attacks will also become difficult but all other types of attacks are possible.

18 Slasher

Specific types of mechanisms are introduced to mitigate certain security concerns in Proof of Stake (PoS) blockchain systems. The Slasher mechanism aims to deter malicious behaviour, such as double-signing, and prevent validators from attempting to create multiple forks of the blockchain, which could lead to various security issues.

The primary purpose of the Slasher mechanism is to penalize validators who engage in harmful activities that undermine the security and integrity of the POS blockchain.
In this, each block must have POW and be signed by a group of persons owning stake.
At a level in the chain, validators must sign one block and avoid creating forks in the blockchain. If he/she fails to do so they will be penalized using the stake. Validators are decided well in advance and they must deposit some stake before they reach the block which they need to validate. After successful validation, this money will be returned back. If failed, some part of the stake goes as a transaction fee, some as a miner fee and the rest is burned. That is, this money is sent to public key 0.
Byzantine Fault Tolerence : If two-thirds of the validators are honest there is no chance of having two valid blocks at the same level.
In this, the mining reward is reduced 50 times every year.
When in block B_K, validators are selected with address a such that

$$Hash(a, Hash(B_k)) < bal(a) * Target$$

The target is adjusted such that there are minimum of 15 volunteers every time. If no block is selected then that level is completely skipped and jumps to next level and the target for the next block decreases by 2^{n+2} times where n is the number of blocks skipped.

19 Anonymity in Blockchain

Anonymity refers to the state of being anonymous or unidentified. In the context of cryptocurrencies like Bitcoin, anonymity is desirable for users who want to protect their financial privacy. Traditional financial systems often require personal identification information when conducting transactions, but cryptocurrencies offer higher pseudonymity.
When you transact with a cryptocurrency, your real-world identity is not directly linked to your wallet address. Instead, you are identified by a unique alphanumeric address or a public key. While your transactions are recorded on the public blockchain, your identity remains hidden unless you voluntarily reveal it.

19.1 Mixer

A mixer, also known as a cryptocurrency mixer or tumbler, is a tool or service designed to enhance the privacy and anonymity of cryptocurrency transactions. It achieves this by mixing or combining multiple users' funds to obfuscate the origin and destination of the coins.
When using a mixer, a user sends their cryptocurrency to the mixer's pool along with the coins from other users. The mixer then shuffles and redistributes the coins, making it challenging to trace the original sender and recipient of each transaction. As a result, the transaction history becomes less transparent, thereby enhancing privacy.
Mixers can be helpful in breaking the link between your wallet address and your identity, providing an additional layer of privacy.
However in this case some companies might not be honest and you have to elect trustworthy companies who offer mixer services.
Advanced forensic analysis techniques and blockchain analysis tools can sometimes deanonymize certain transactions, especially if users inadvertently reveal information that links their identity to their cryptocurrency activities.

19.2 How does mixer work

The user who wants to enhance their privacy initiates a transaction by depositing their cryptocurrency into the mixer's pool. This is usually done by sending the coins to a specific address provided by the mixer.
The mixer combines the deposited funds with those of other users who are also utilizing the mixer service. By pooling multiple transactions together, it becomes challenging to track individual transactions or identify the origin of the coins.
Once a sufficient number of transactions have been pooled, the mixer starts the mixing process. The mixer breaks down the pooled funds into smaller and more randomized amounts, splitting and recombining them multiple times.
The mixer creates new and random addresses for the recipients to receive their coins. These new addresses are not directly linked to the original sender or

recipient's addresses, further obfuscating the trail of the coins.
The users can now withdraw their mixed funds from the mixer. The coins they receive will have been mixed with those from other users, making it difficult to trace the origin of the funds.

19.3 Mixing fee

mixing fee is taken in such a way that if N people are there in the mixing pool then one of the N-1 people doesn't get money and others get back their exact money. This is one method and there are other methods like taking a percentage of the money.

20 RAFT

Suppose there is only one centralized server where the information is stored. What if the server stopped working suddenly? It is always better to back up all the information and store it on some other servers. The more the number of servers used for backup, the more the safety of the information in the server.
RAFT protocol is a consensus mechanism which means different servers agree on the same information.
Suppose let a client requests some information from the server. This information is first sent to a main server(Leader), and the rest of the protocol is as follows

- This information is sent to all other servers also
- Logs in the server get updated
- The information in the logs is stored in SM(state machines)
- then the reply is sent back to the client

The information to others (consensus) is done first so that others can work if the leader server fails.

WHAT IF THE LEADER FAILS
If a leader fails, the server crashes, other servers get to know, and the servers become eligible to become candidates for the next leader. When the leader server crashes the next candidate term will increase by one. A leader is selected based on a voting system.
A server votes only when the logs in the candidate server are a superset of the logs in the server voting. This way, the server with the most logs is selected as the leader.
Also, the term is considered to select the leader. Suppose the leaders' message took so long to reach other servers, and other servers thought the leader server was dead. Then on the election, if the old leader also participates, the leader with a higher term gets selected (the old leader's term would not be increased).

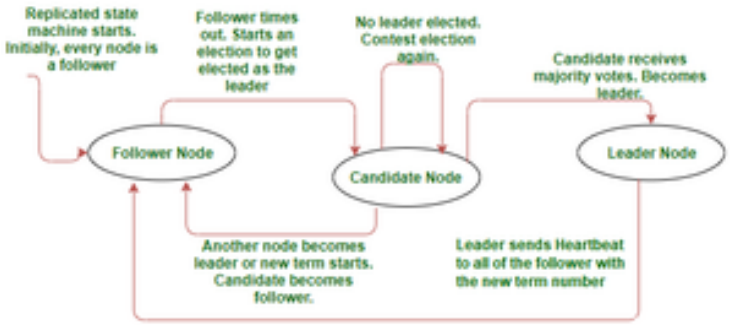

Figure 6: RAFT

To know the leader's term, every node puts term in its messages.

IMPORTANT POINTS FOR RAFT TO WORK

- Choose a new leader if the current leader fails.
- Leader accepts log entries from clients and replicates them across clusters, forcing others to agree to its log

Commiting inputs: If the leader commits an input and updates the local state, then effectively RSM(Recursive state machine, i.e. the group of servers) has committed this input.
HOW TO ENSURE IT WORKS
Rule: The leader commits an input only after ensuring that the majority of the nodes have received that input and that the log of those nodes is the same as its own up to this point.
Availability/Liveness(new inputs accepted and eventually committed)
The RSM is fully functional as long as the majority of servers are operational and can communicate with each other and the client.
RAFT is available if

$broadcastTime << electionTimeout << AverageTimeBetweenFailureEvants$

21 BYZANTINE FAULT TOLERENCE

Byzantine fault: A Byzantine fault is a condition of a computer system, particularly distributed computing systems, where components may fail, and there is imperfect information on whether a component has failed.
In a network, some nodes might be honest, and some are not(byzantine nodes) but how to be sure that consensus occurs?
The term "Byzantine" in Byzantine Fault Tolerance originates from the Byzantine Generals' Problem, a theoretical scenario in which a group of generals

surrounding an enemy city must coordinate their attack or retreat strategies
Consensus Requirements

- **Agreement**: All honest generals(servers) must reach the same decision
- **Termination**: All honest generals must eventually make a decision
- **Validity**: If a leader is honest, all other honest generals must agree to his command no matter what he decides.

Asynchronus system: No upper bound on the time a node takes to receive process and respond to an incoming message.
Let the number of nodes in a network be N and the byzantine nodes be f.
Suppose a client sends an input and only N-f servers respond saying they have updated the input. Should the client wait for more responses?
Again another client wants to read the input. Again he waits for N-f servers to respond.
In the worst-case scenario in the first f servers, which didn't respond all are honest but could not respond due to other reasons. And while responding let the other f servers who were honest and responded the first time but now could not respond.
This means there were N-2f servers that responded every time and in these servers, f might be a byzantine Fault. To always bring a correct output there must be at least one honest server in these N-2f servers i.e.

$$N - 2f > f$$

Which means $f < \frac{N}{3}$
So for a synchronous network, if Byzantine fault servers are less than $\frac{N}{3}$, then the network is Byzantine Fault tolerant.

FLP (Fischer, Lynch, and Paterson) result
It is impossible to have a deterministic protocol that solves consensus mechanism in a message passing Asynchronous system in which at most one process may fail by crashing.

22 Till what I have completed

I have done watching videos of the cs765 course till lecture 20 and learnt many concepts like what blockchain is, how does currency system work, the structure of a blockchain, consensus mechanisms like POW, and POS, how to keep transactions anonymous in blockchain using mixers, Raft and many more.