GUARDIANS EDTECH

Certified Ethereum Developer Level 1

A part of
Certification in Blockchain Technology
Conducted by Guardian Edtech

Trainer: Mantavya Jain

Table of Contents

....1

History of Money – Past, Present & Future 4

Bitcoin: The Bible for Blockchain Technology 7

Introduction to Blockchain Technology 9

Distributed Ledger Technology (DLT)..13

Cryptography Basics 14

Peer – To – Peer Network 18

Understanding Blockchain Structure 20

Node:...20

Block Header:...20

Block Data: 21

Addresses:...22

Wallets: 24

Consensus Mechanism...28

Two General Problem:..28

Byzantine General Problem..30

Byzantine Fault Tolerance 31

Types of Consensus Mechanisms...36

Working of a Blockchain System 39

Life Cycle of Transaction 44

Case Study of Bitcoin...45

Case Study of Ethereum...49

Types of Blockchains 53

Layers of Blockchains 55

Blockchain Forks..57

Blockchain Bridges 60

Blockchain Tokens 62
Blockchain Use Cases 65
Blockchain Fundraising Models...67

History of Money – Past, Present & Future

"Long ago, our ancestors relied on the **barter system** for trade, such as exchanging 3 sacks of grain for 2 sturdy goats."

But later, it was realized that this system had too many limitations:

1. Challenge of finding mutually agreeable exchanges
2. Portability issues
3. Indivisibility

So, they transitioned to more **universally acceptable, divisible, uniform, and portable forms of money** such as gold, silver, cowry shells, salt, and various other commodities.

Then China introduced **the world's first paper currency**, which was backed by gold, silver, or other precious metals, providing a more efficient and standardized medium of exchange.

Then came the era or the present time of **fiat currency**: Modern economies began using money that had no intrinsic value but was backed by the trust and authority of governments.

Examples of fiat money include the Indian Rupee, US Dollar, Euro, and Yen. Unlike tangible commodities, fiat money relies on trust and legal recognition.

Finally, with the advent of the internet. Money began to be **digitalized**. This includes:

1. Credit Cards
2. Debit Cards
3. Net-banking
4. Digital gold
5. Online Banking
6. And other electronic payment systems.

As we advanced into the 21st century, we had:

Bitcoin, the first electronic currency that was peer-two-peer. The first known cryptocurrency. Bitcoin completely changed the perspective on how money works.

Following this, the concept of **Central Bank Digital Currencies (CBDCs)** began to take shape, utilize blockchain or other DLT to represent the digital equivalent of physical money, regulated by the countries national Bank.

For Example: India's has its own CBDC by the name "e-rupee (e₹)" regulated by Reserve Bank of India.

Financial systems built on blockchain technology known as **Decentralized Finance (Defi)** that offer financial services without traditional intermediaries.

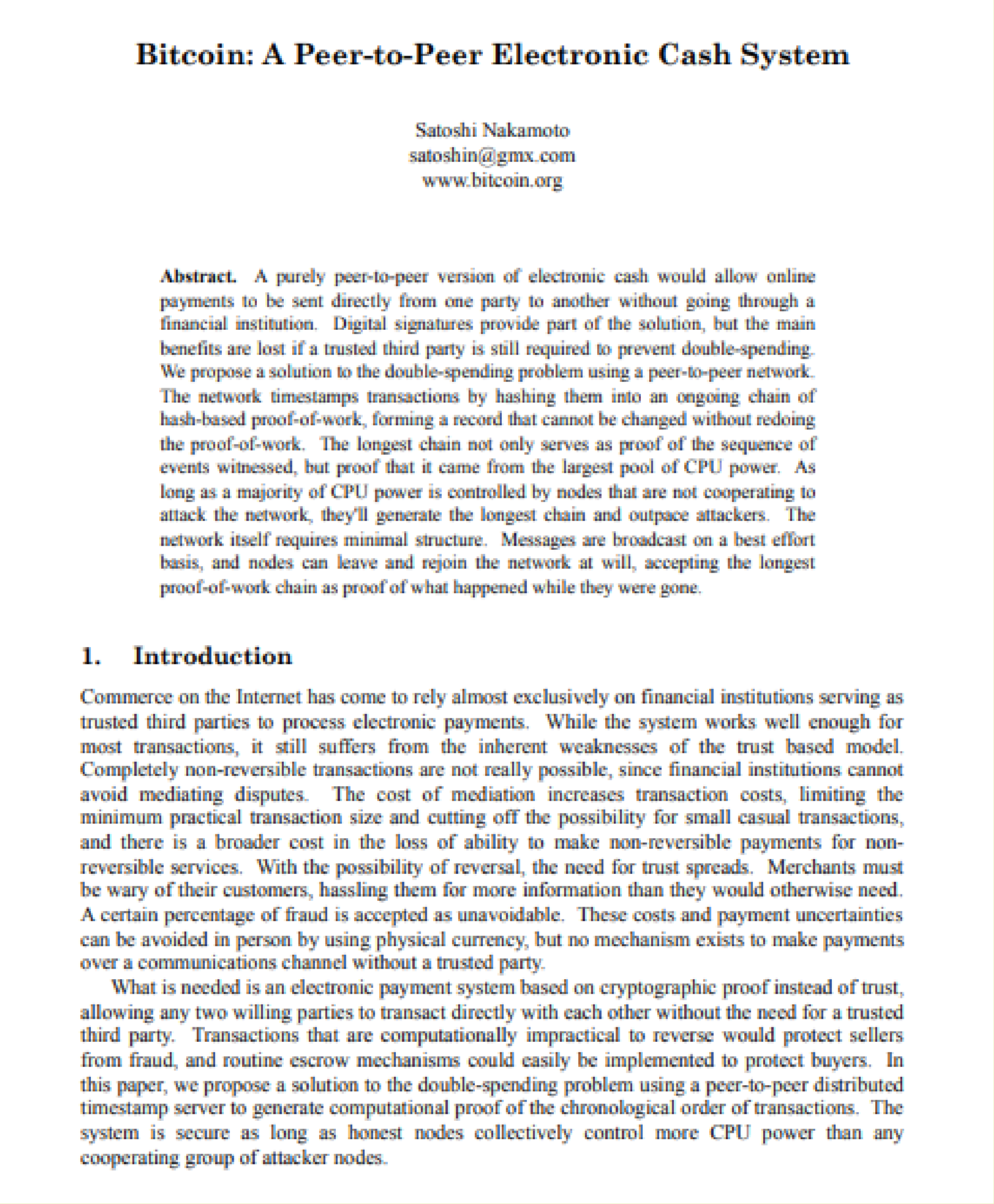
Bitcoin: The Bible for Blockchain Technology

The 2008 financial crisis exposed major flaws in the centralized financial system: lack of transparency, centralized control, inflation, and financial exclusion.
This crisis highlighted the need for a transparent, secure, and decentralized financial system where individuals could transact without relying on intermediaries like banks. This vision led to the creation of Bitcoin that disrupted the money.

Bitcoin was introduced in 2008 by an anonymous entity known as Satoshi Nakamoto.

On October 31, 2008, Nakamoto published a white paper titled *"Bitcoin: A Peer-to-Peer Electronic Cash System,"* detailing the concept and functioning of Bitcoin, the first decentralized cryptocurrency.

The **White Paper** of the Bitcoin introduced the concept of underlying technology known as Blockchain.



Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

1. Introduction

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers from fraud, and routine escrow mechanisms could easily be implemented to protect buyers. In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes.

https://bitcoin.org/bitcoin.pdf

Introduction to Blockchain Technology

So, in very easy terms, **blockchain is the Internet of Value**.

Unlike the internet we use today that is Internet of data, moves data across the world in seconds. These data can be in form of photo, video, documents, etc.

But blockchain, moves values across the internet in seconds. These values can be in form of cryptocurrencies or tokens.

Let's consider a **Real World blockchain analogy**…….

- Imagine a massive vault system from a bank in your society. Where the **data of residents** of the society is stored.
- No **Central authority** is responsible for this vault system. Instead every member are bind by the **rules** to look over each other's data.
- The vault is filled rows of deposit boxes.
- Each deposit box is **made up of glass**, allowing everyone to visualize the content of the deposit box, **but have access to their vault**.
- Suppose a new wants joined the society and opens a new deposit box. He/she get a key that is **unique** to that box.

This is the fundamental concept of blockchain. Anyone can see the contents (data) of all other addresses (residents), but can’t interpret them because they all are secured using **encryption and hashing**.

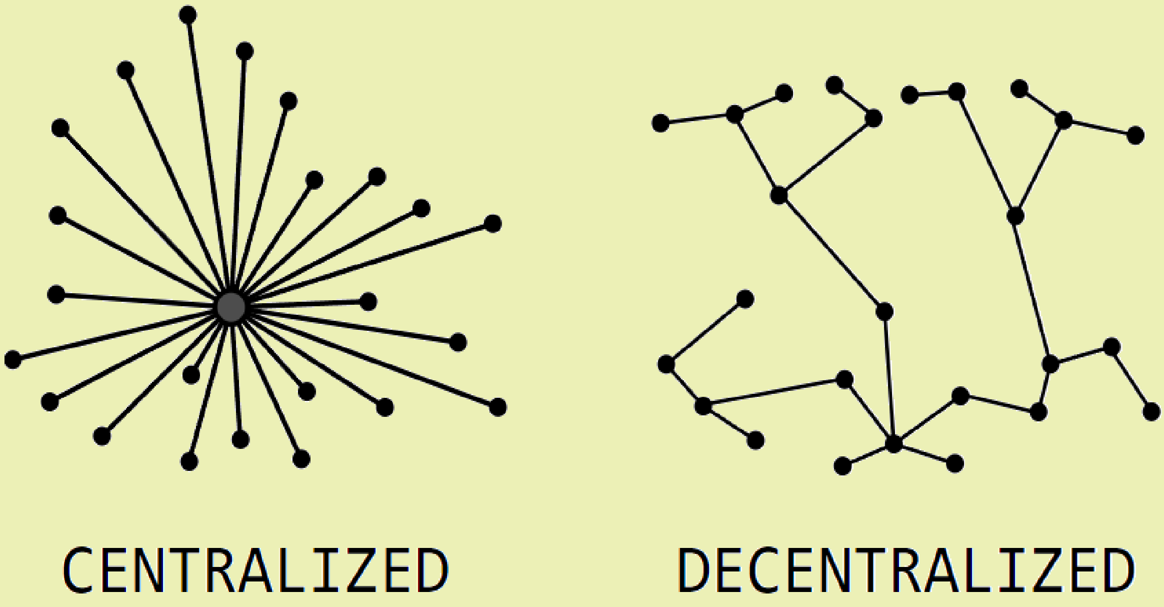
Translating to Blockchain -

1. **Blockchain System as the Vault:**
    - Think of the blockchain as this massive vault where data is securely stored.
2. **Decentralization:**
    - In blockchain, there is no central authority. Instead, everyone on the network (residents) follows a set of protocols (rules) to verify and oversee data.
3. **Blocks as Deposit Boxes:**
    - Data on the blockchain is stored in blocks, similar to the deposit boxes in the vault.
4. **Transparency and Security:**
    - The blockchain is transparent because everyone can see the contents of each block. However, only the owner of a block can access and manage its data using a unique cryptographic private key.

5. **Unique Keys:**
    - When someone adds new data to the blockchain, they get a unique cryptographic key that only they can use to access and control their specific block of data.

Features of blockchain:

1. Decentralization – It doesn’t have a single point of control which can led to single point of failure,

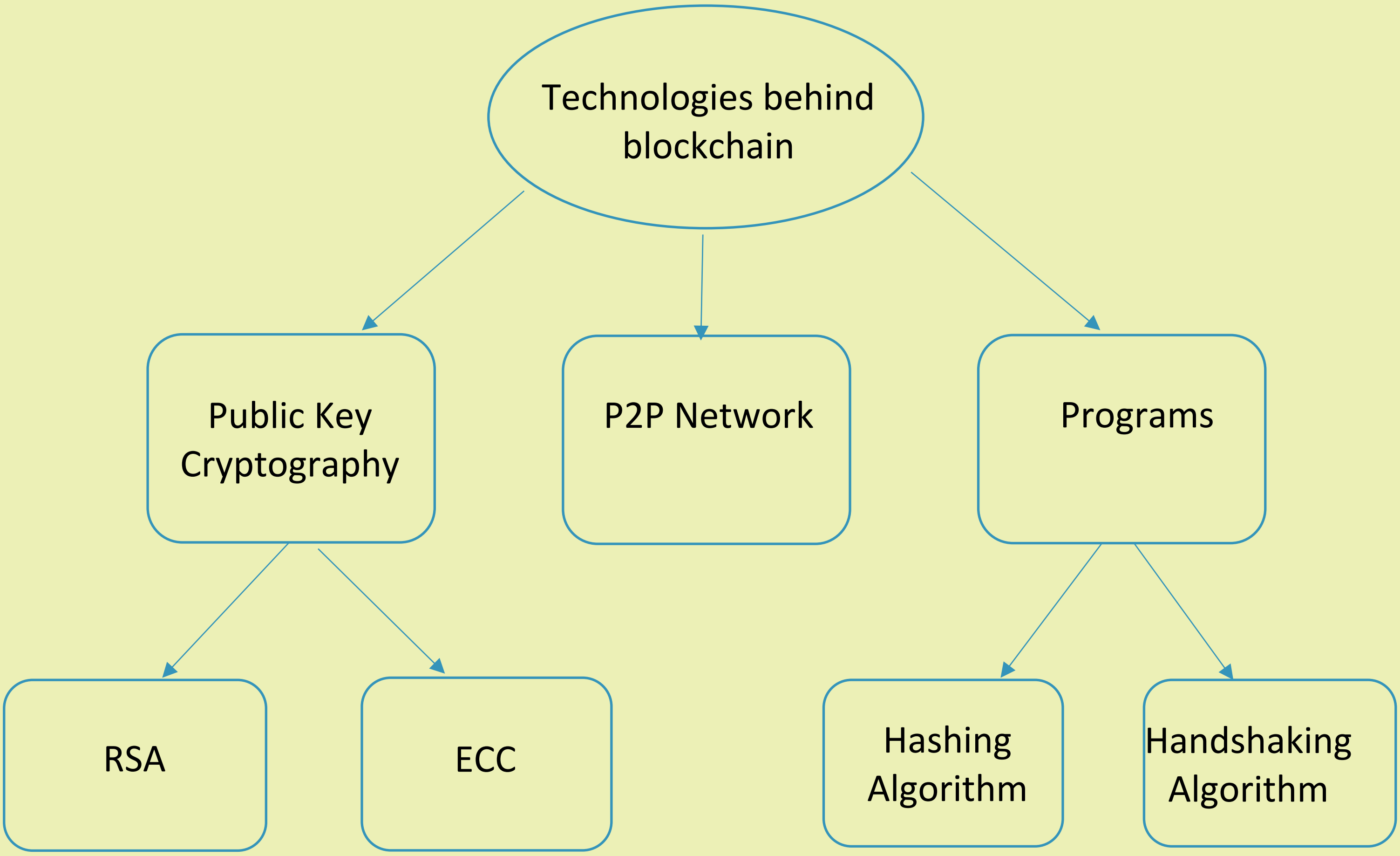


2. Immutability – Once the data has been recorded, it cannot be deleted/altered.

3. Transparency – Everyone can see what’s happening on the blockchain.

Technologies behind Blockchain:

Note: Blockchain is not something entirely new but rather an innovative amalgamation of decades-old, tried-and-tested technologies. It builds on the foundations of Public Key Cryptography, developed in the 1970s, Cryptographic Hash Functions, also from the 1970s, and Proof-of-Work, which emerged in the 1990s.



Distributed Ledger Technology (DLT)

Distributed Ledger Technology (DLT) is a database of information that's shared and duplicated across a network of computers in different locations. They are of many structures:

Blockchain which has a more linear chronological structure. Organized as a chain of blocks, each containing a list of transactions

Graph utilizes a Directed Acyclic Graph (DAG) called the Tangle

Mesh employs a decentralized, agent-centric framework where each participant maintains their own chain

Trees utilizes a tree-based architecture with unique notary services to ensure transaction consensus

Note: All Blockchain's are DLT's, but every DLT is not a Blockchain

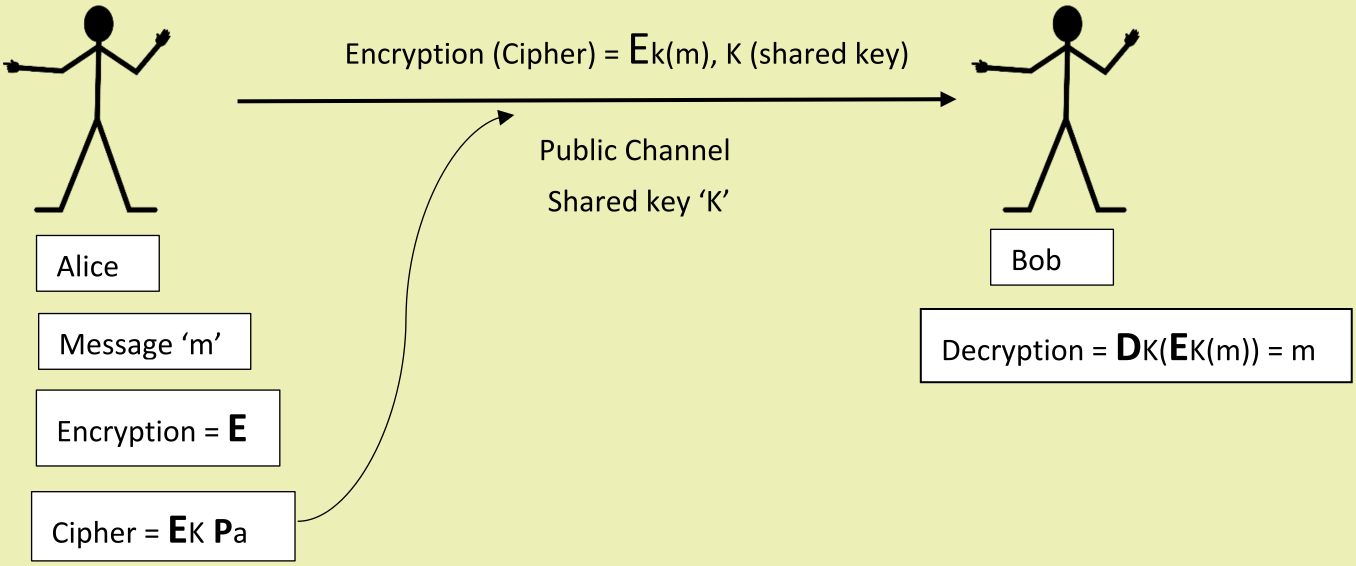
Cryptography Basics

Cryptography is the science of secret writing.

Cryptography is essential to blockchain technology, ensuring secure communication and data integrity. Here are the key cryptographic concepts:

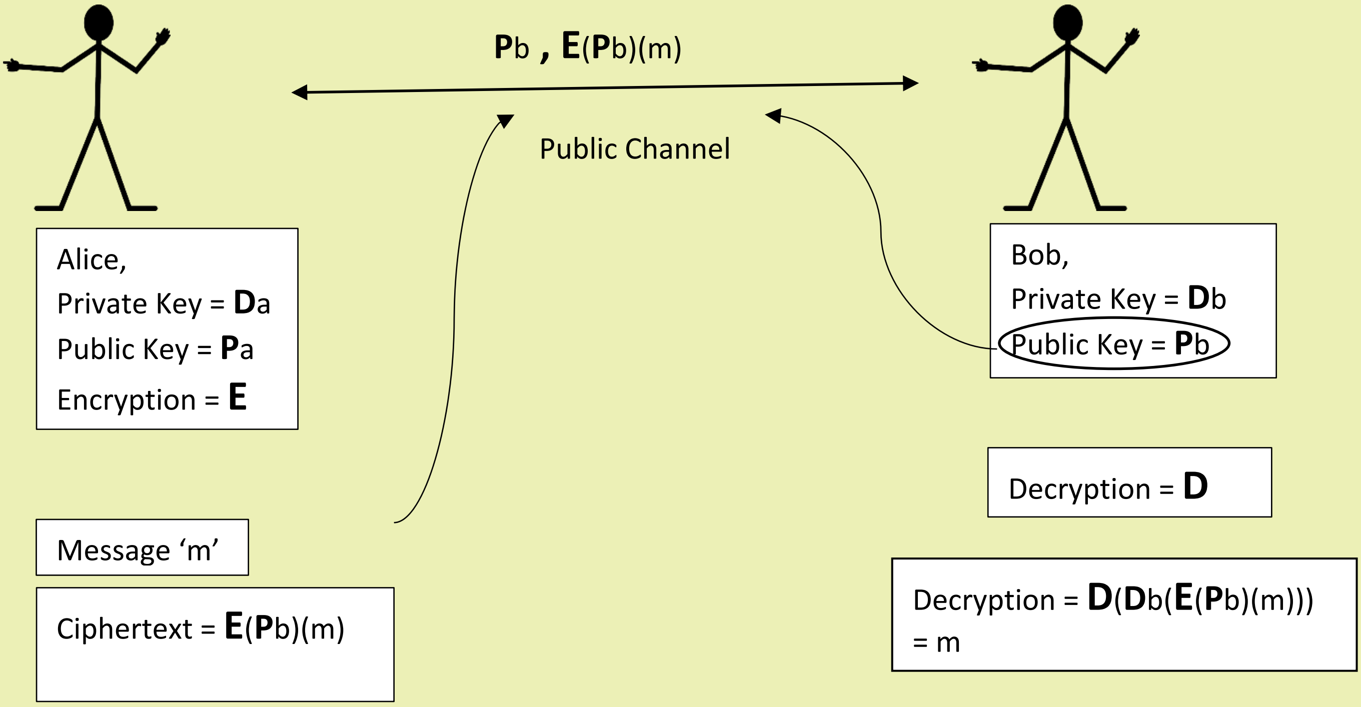
Symmetric Key Encryption

- **Definition:** The same key is used for both encryption and decryption.
- **Examples:** AES (Advanced Encryption Standard), DES (Data Encryption Standard).
- **Use Case:** Efficient for encrypting large amounts of data but requires secure key distribution.


Encryption (Cipher) = $E_k(m)$, K (shared key)
Public Channel
Shared key 'K'
Alice
Message 'm'
Encryption = **E**
Cipher = $\mathbf{E}_K \mathbf{P}_a$
Bob
Decryption = $\mathbf{D}_K(\mathbf{E}_K(m)) = m$

Asymmetric Key Encryption

- **Definition:** Uses a pair of keys, a public key for encryption and a private key for decryption.
- **Examples:** RSA (Rivest-Shamir-Adleman), ECC (Elliptic Curve Cryptography).
- **Use Case:** Secure key exchange and digital signatures.



Pb , E(Pb)(m)

Public Channel

Alice,
Private Key = Da
Public Key = Pa
Encryption = E

Message 'm'

Ciphertext = E(Pb)(m)

Bob,
Private Key = Db
Public Key = Pb

Decryption = D

Decryption = D(Db(E(Pb)(m)))
= m

RSA (Rivest-Shamir-Adleman)

- **Definition:** An asymmetric encryption algorithm based on the difficulty of factoring large integers.
- **Features:** Widely used for secure data transmission and digital signatures.

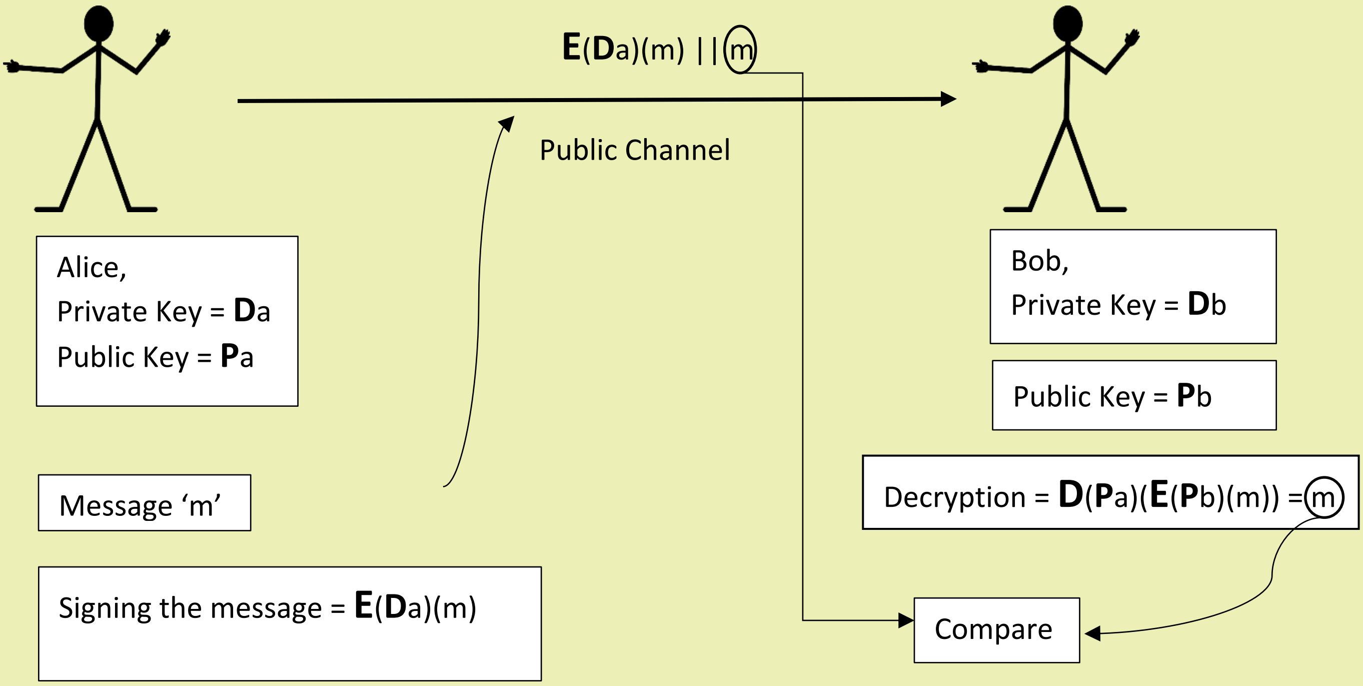
- **Strengths:** High security, well-established.

ECC

- **Definition:** Uses elliptic curves over finite fields for encryption.
- **Features:** Provides similar security to RSA but with smaller key sizes, resulting in faster computations.
- **Strengths:** Efficiency and security.

Digital Signature

- **Definition:** A cryptographic technique used to verify the authenticity and integrity of a message or document.
- **How It Works:** A message is hashed, and the hash is encrypted with the sender's private key to create the signature. The recipient can verify the signature using the sender's public key.



Hashing

- **Definition:** Converts input data into a fixed-size string of characters.
- **Examples:** SHA-256, SHA-3.
- **Use Case:** Ensuring data integrity, password storage, and digital signatures

Peer – To – Peer Network

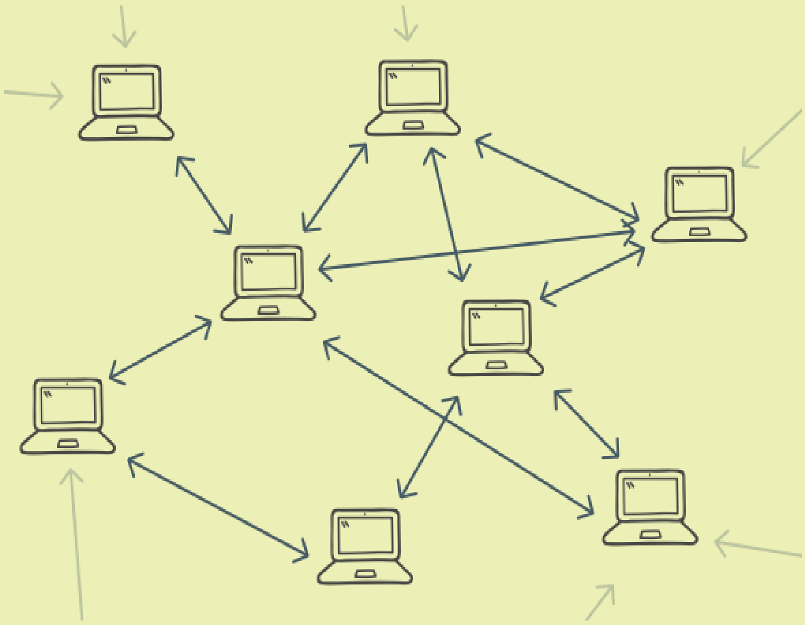
Imagine you're downloading a movie from a torrent. If you've done this before, you might have noticed something interesting: you can start watching parts of the movie even before the entire file is fully downloaded. Why is that?

This happens because each part of the movie file is independent of the others. As soon as a segment is downloaded, you can watch that portion of the movie. This is a key feature of peer-to-peer (P2P) networks.

Now, think about downloading a movie from a central server, like a traditional website. In this case, the movie file isn’t fragmented. You have to wait until the entire file is downloaded before you can start watching it. If there’s any interruption or slow connection, you have to wait for the whole file to finish downloading. This is because the data is stored as a complete unit on the server.

Here's where the beauty of P2P networks comes in. In a P2P network, data is fragmented and distributed among multiple peers (users). When you download a file using a P2P network like a torrent, you’re not just downloading from one source. Instead, you're downloading different parts of the file from multiple peers simultaneously. This makes the process faster and more efficient.

- **Fragmented Data:** The movie is broken into smaller parts, and each part is shared across the network.
- **Distributed System:** Instead of relying on a single central server, each peer contributes to the downloading and uploading process.
- **Resilience and Speed:** If one peer goes offline, you can still download the needed parts from other peers. This decentralization makes the network more resilient and often faster.



Understanding Blockchain Structure

Node:

A node is any computer that connects to the blockchain network. Each node stores a copy of the blockchain and participates in the network's processes, such as validating transactions. For example, in the Bitcoin network, each user running the Bitcoin software is a node.

Block Header:

The block header contains metadata about a block and includes the following components:

- **Index:** A unique identifier for each block in the blockchain. The first block with index 0 is known as genesis block.
- **Previous Hash:** The hash value of the previous block in the chain, linking the blocks together and ensuring integrity.
- **Nonce:** A random number used only once for the cryptographic hashing process during the mining of a block.
- **Timestamp:** The exact time when the block was created, ensuring chronological order.

- **Current Block Hash:** The unique hash value of the current block, generated based on its contents and ensuring data integrity.

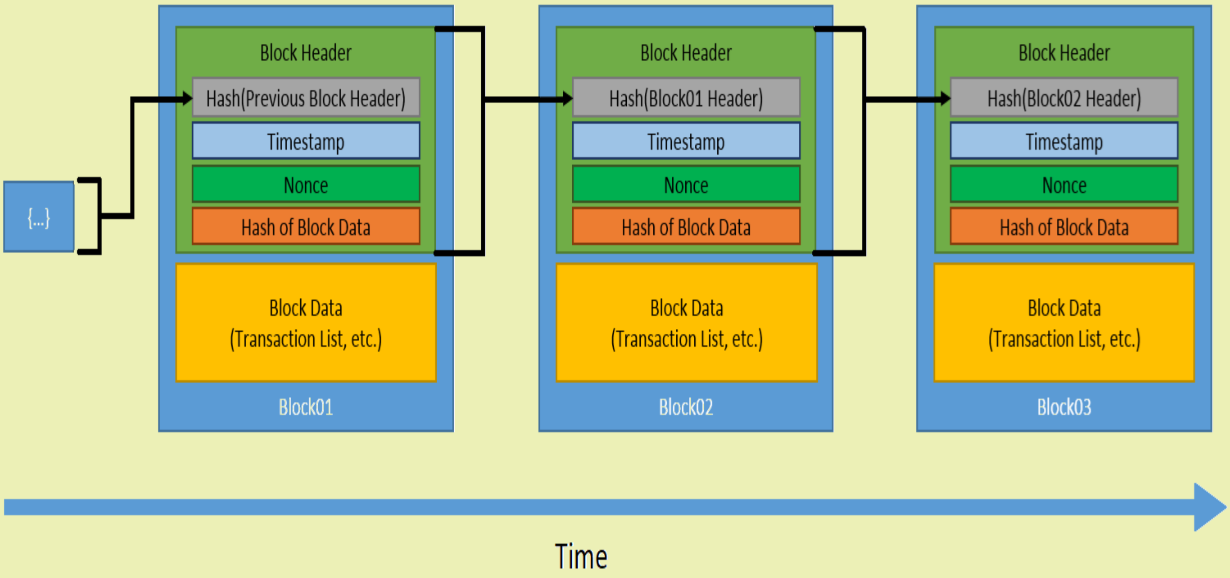
Block Data:

The block data section contains the actual transactions and a Merkle Tree:

- **Transactions:** A list of all transactions included in the block, recording the transfer of assets or information between parties.
- **Merkle Tree:** A binary tree used to efficiently and securely verify the integrity of large sets of data.

Why Merkle Tree in Blockchain?

Merkle Trees are chosen for blockchain because they provide a way to verify large amounts of data quickly and efficiently. Each leaf node in the tree represents a hash of a transaction, and each non-leaf node is a hash of its children nodes. This structure allows for quick verification of the data's integrity without needing to check the entire dataset. Compared to other data structures, Merkle Trees are highly efficient in verifying data integrity and detecting changes, making them ideal for blockchain's decentralized and distributed nature.



Addresses:

An address is similar to a *Bank account or an UPI Id*, where we can send and receive crypto's. In our blockchain it is a unique identifier that represents a possible destination for a cryptocurrency payment.

Sample Address: 1AAE1EDcCAUmyBfi46G3vpik8oCaKVoabT

It is derived from the user's public key through a series of cryptographic processes.

- **Public Key:** Generated from the user's private key, it is used to create the public address. The public key allows others to send cryptocurrencies to the user.

Sample Public Key = 03f87ecf12e0ea14cd666f3a125b612uuu46705575g45
7082714e1e61005792cb2

- **Private Key:** A secret key known only to the user, which is used to sign transactions and access the funds associated with the public address. Keeping the private key secure is crucial, as anyone with access to it can control the associated funds.

Sample Private Key = b4bb85bad78db692d16ec5b3d478a8b176b874f248c8
3e544d9fed18e6246d22

Note: Always remember to not share your private key with anyone. As allowing anyone to possess your private key directly means giving your crypto ownership to others.

That's why there is a saying in crypto space:
"**NOT YOUR KEYS, NOT YOUR COINS**"

Wallets:

A wallet in the context of blockchain is a digital tool that allows users to store, send, and receive cryptocurrencies and also monitor balances.

It doesn't actually store cryptocurrencies like a physical wallet stores cash; instead, it stores the cryptographic keys required to access and manage these digital assets.

There are many forms of wallets:

1. Paper Wallets
2. Software Wallets
3. Hardware Wallets
4. Exchange Wallets

There are also wallets like:

Hot Wallet: They are connected to internet and stores crypto online. They are not safe. They come under software or exchange wallets.

Cold Wallet: That stores your crypto's offline without the need of internet. They are much secured than the Hot Wallet. They come under paper or hardware wallets.

Paper Wallets: Physical printouts of the public and private keys, often in the form of QR codes. The most secured is by just writing down the private and public keys on a piece of paper.

Load & Verify
Bitcoin Address
14sScGvSjGtxNbqFUoStoXN7eXydaN1JMM
bitcoin
Amount:
Strength in Numbers
Private Key
5JGnozkYRHhpap6DeUrHRmceTHCh8f7wcszpHGepCxS998Hiqqh
Spend

Hardware Wallets: Physical devices designed to securely store private keys offline. They are little pricey. Examples include Ledger and Trezor.

Software Wallets: These are applications or programs installed on a computer or mobile device. Examples include MetaMask and Trust Wallet. They usually have a *mnemonic seed or seed phrase.* That usually looks like this:

Rabbit

London

Vegetable

Cartoon

Sofa

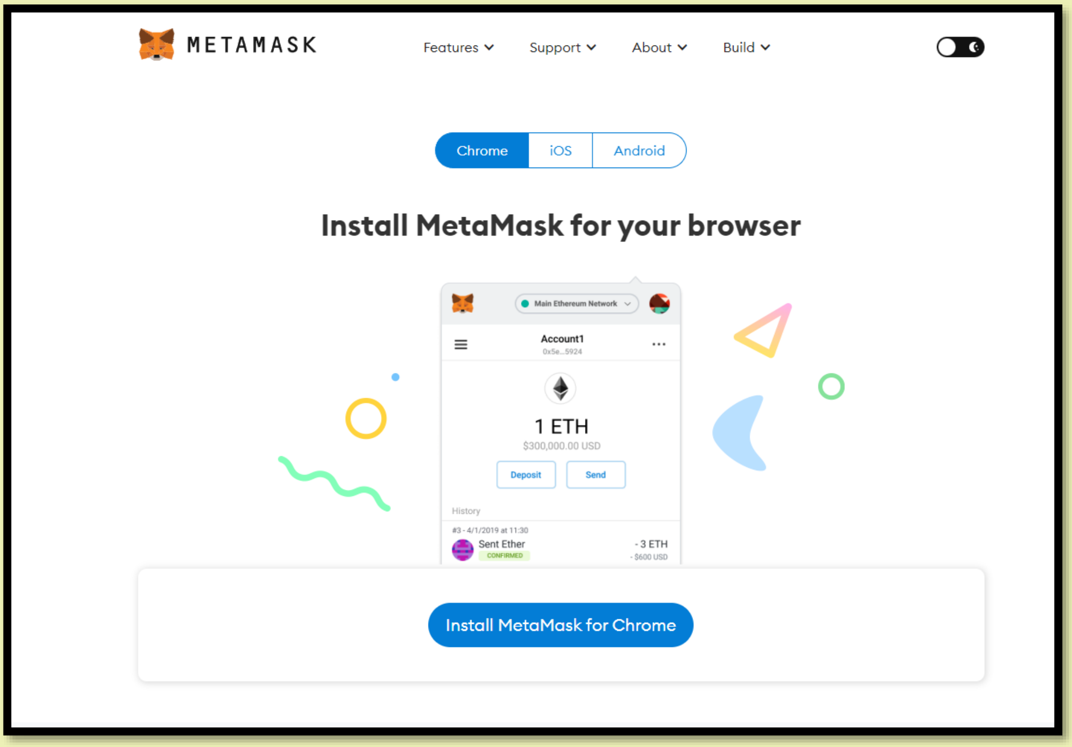
Music

Plastic

Quote

Achievement

Old



Install MetaMask for your browser

Exchange Wallets: These are also the types of software wallets, but are provided by 'Exchange Platforms'. For example Exchange wallet provided by CoinDCX, Coinbase, etc.

Note: If there is a centralized exchange, then they have the custody of your crypto's. Which is not good as if they go bankrupt, all your crypto's will be gone.

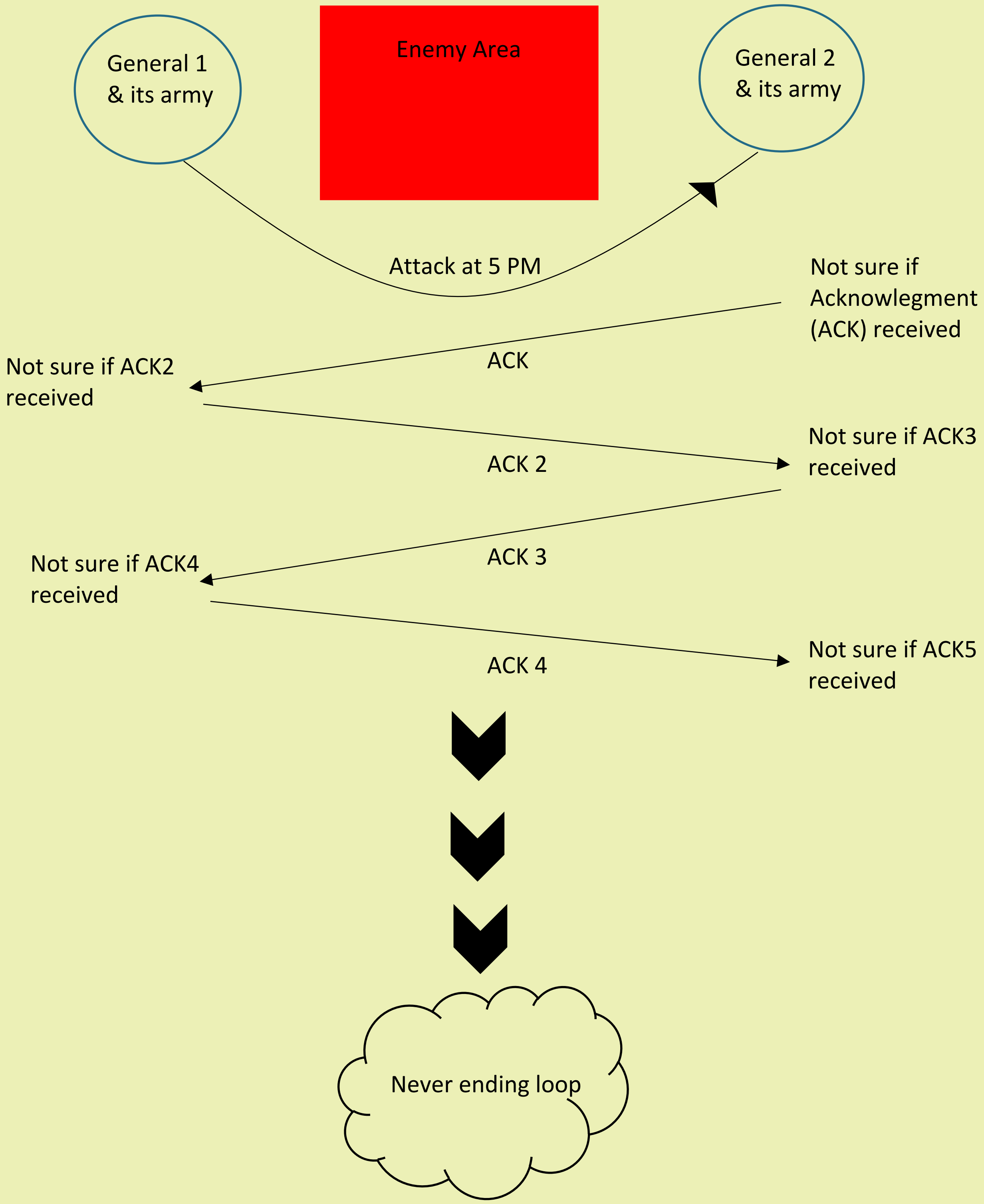
Consensus Mechanism

Consensus: When the majority of the nodes or participants in a network agree on a single state, consensus is achieved.

Two General Problem:

Imagine two generals planning to attack an enemy area. They must coordinate their attack but can only communicate over and via enemy area.

If one general does not receive confirmation from the other, neither will attack, leading to failure. This highlights the difficulty in ensuring reliable communication and synchronization.



General 1 & its army

Enemy Area

General 2 & its army

Attack at 5 PM

Not sure if Acknowlegment (ACK) received

ACK

Not sure if ACK2 received

ACK 2

Not sure if ACK3 received

ACK 3

Not sure if ACK4 received

ACK 4

Not sure if ACK5 received

Never ending loop

Byzantine General Problem

In the Byzantine Generals Problem, imagine an army led by one commanding general and several lieutenants. The goal is for the generals to agree on a coordinated plan of attack. However, some of these generals might be traitors, deliberately trying to mislead the others.

In this scenario, the commanding general issues an order to the lieutenants. If the commanding general is loyal, all loyal lieutenants should follow this order and attack at the specified time, ensuring coordination. However, if the commanding general is a traitor, they might give conflicting or false orders. For example they said *“Attack at 5 PM, and themselves don’t attack”.*

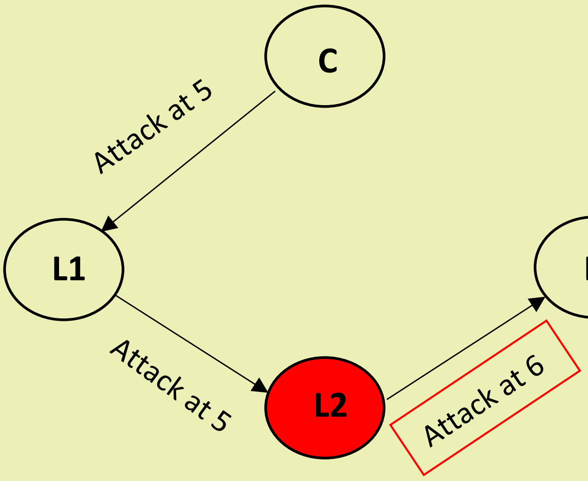
To reach a consensus despite potential traitors, the lieutenants must communicate among themselves and take a majority vote on the received orders.

For the army to successfully coordinate their attack, at least two-thirds of the generals (both the commanding general and the lieutenants) must be loyal. **If more than one-third of the generals are traitors, they can create enough confusion to prevent consensus, leading to a failed coordination and the enemy winning**.

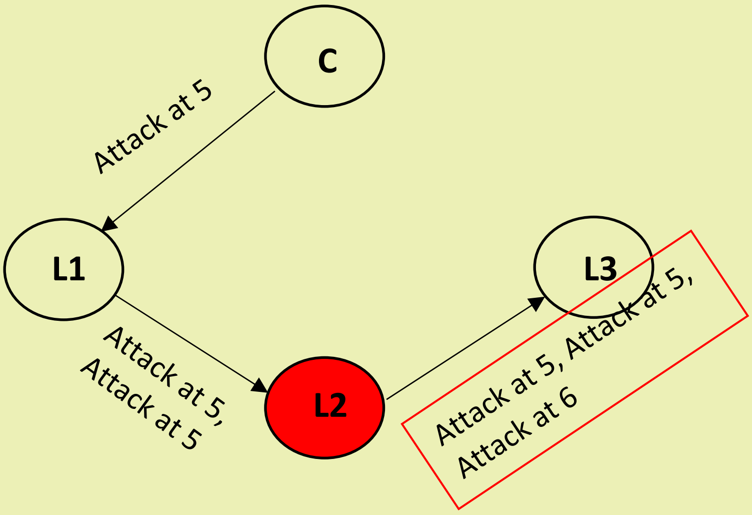
Byzantine Fault Tolerance

Case 1:

Imagine there are one Commander (C), three Lieutenants (L1, L2, L3). Suppose second Lieutenant (L2) is a traitor, then he can change the message of commander. In the below diagram he changed the message "*Attack at 5 PM*" to "*Attack at 6 PM*". **Then Consensus is not received.**



Solution: Appending the new message with old message.

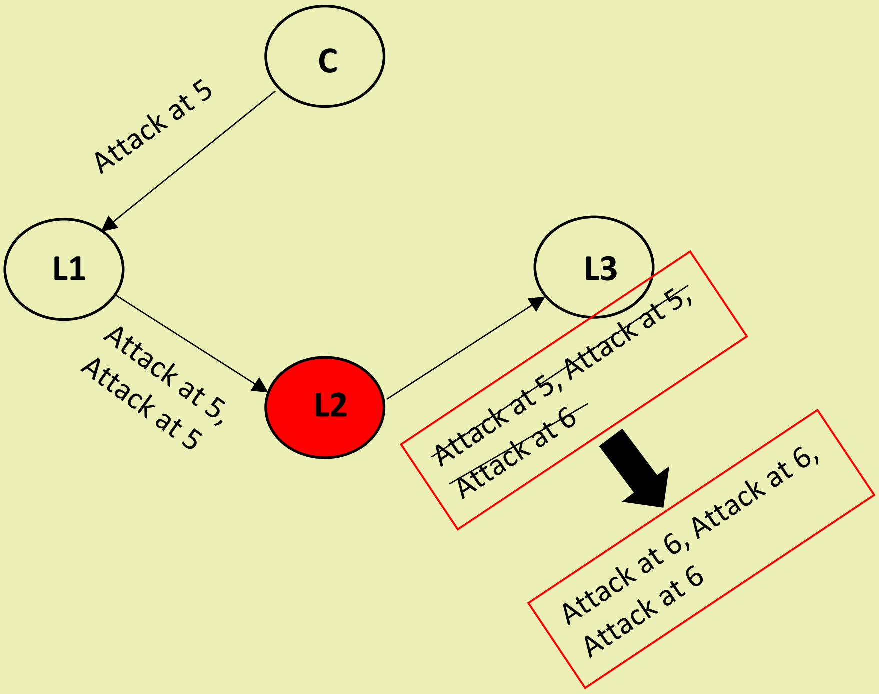


Now, L3 can predict that L2 must be a traitor, because the messages are not in the synchronized manner.

Now there's an another issue,

Case 2:

Suppose L2 have the ability to change the messages and he changed all the previous message that were appending every time. From "*Attack at 5, Attack at 5*" to "*Attack at 6, Attack at 6*", and then appendid his message to it "*Attack at 6, Attack at 6, Attack at 6*". In this way L3 will not be able to guess is this a valid message.

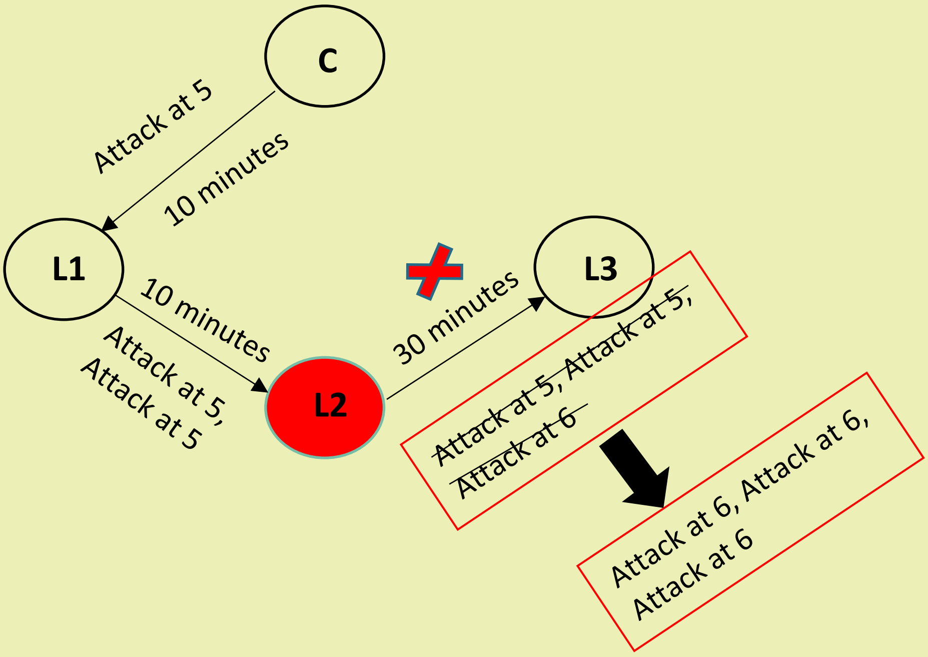


Solution:

We implement a time bound and a nonce {Discussed Earlier}. Due to this we'll implement a time bound for the formation of message which will be 10 minutes, meaning messages has to be delivered within 10 minutes and each soldier are expected to receive the message within 10 minutes.

Now with this time bound and nonce:

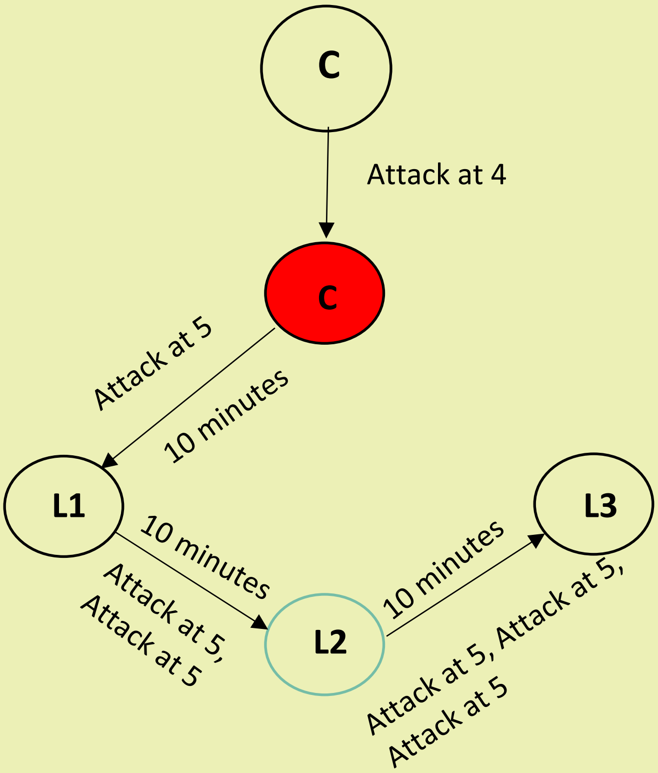
- L2 will take 30 minutes to send the new message, as to alter previous message will take him 20 minutes ("*Attack at 5, Attack at 5"*, which is **10+10 = 20**). And then will append his message which will also take 10 minutes. Therefor **20+10 = 30 minutes**.
- And L3 will be expecting the message within 10 minutes. Therefore L3, will get to know that L2 is a traitor.



Case 3:

Now suppose the commander is the traitor, then in such case consensus will reach, but it's not everyone wanted. Consensus might have reached but what at what cost? Lost the war just because their own commander misleads message.

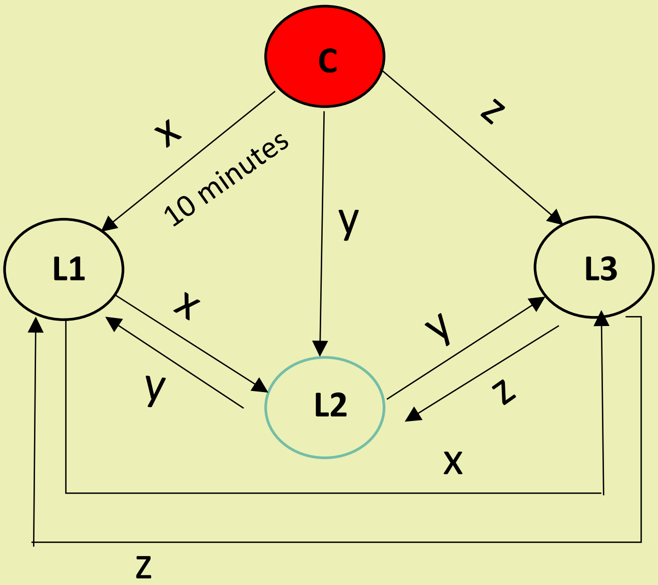
Suppose the other commander sends the message to this commander "*Attack at 4*". But he changed it to "*Attack at 5*" and sends to all lieutenant.



Solution:

Let's connect all them together, in such case if he tries to send every lieutenant different message. This time every lieutenant will get to know, that commander is a traitor.

Suppose the Commander sent "x" message to L1, "y" message to L2, "z" message to L3. Then –



In this, it can be interpreted, that the consensus will not reach as nobody have the same message.

Types of Consensus Mechanisms

Currently, there are more than 75 types of consensus mechanisms. Some of the widely used consensus mechanisms are:

Proof of Work (PoW)

Think of PoW like a big puzzle-solving competition. Miners (computers) race to solve a complex mathematical problem. The first one to solve it gets to add the next block of transactions to the blockchain and earns a reward. It’s like a lottery, but you need a lot of computing power to participate.

Proof of Stake (PoS)

In PoS, instead of solving puzzles, you get a chance to add the next block based on how many coins you own and are willing to "stake" or lock up as a guarantee. The more coins you stake, the higher your chances. It’s like having more tickets in a raffle—the more you have, the better your chances of winning.

Hybrid PoW/PoS

This combines both PoW and PoS. Miners first use PoW to solve puzzles and create blocks, but PoS is used to validate these blocks. It’s like having a competition to find a winner (PoW), and then a panel of judges (PoS) verifies that the winner played by the rules.

Delegated Proof of Stake (DPoS)

DPoS is like a democracy. Coin holders vote for a small number of delegates who will create and validate blocks on their behalf. These delegates are like trusted representatives. If they misbehave, they can be voted out and replaced.

Proof of Authority (PoA)

PoA relies on a few known trusted entities that are identified, and are credible to create and validate blocks. These entities are pre-approved and known to the network. Their reputation is at stake, so they are expected to act honestly.

Proof of Work Time (PoWT)

This is a variation where the blocktime scales with mining power, means the speed of blockchain increases with power increases. It helps blockchain to scale efficiently, thus increasing transaction speed with power.

Proof of Meaningful Work (PoMW)

Here, the work done should be meaningful or useful in some way. It’s like solving the real-world problem, such as scientific research or complex computations.

Proof of Elapsed Time (PoET)

PoET is like taking turns based on a random timer. Each participant waits for a random amount of time and the first one whose timer goes off gets to create the next block. It’s fair and efficient, relying on trusted hardware to ensure the timers are honest.

Working of a Blockchain System

Imagine there are five people in a blockchain network: *Alice, Bob, Charlie, David, and Eve*. They are part of a decentralized system where transactions occur and are recorded on the blockchain.

Let's say three transactions are initiated:

T1: *Alice sends 1 Bitcoin to Bob.*

T2: *Charlie sends 2 Bitcoins to David.*

T3: *Eve sends 3 Bitcoins to Alice.*

These transactions are initially stored in a **Transaction pool**, which is a temporary holding area for pending transactions waiting to be processed. This pool is maintained by all nodes in the network, but the transactions are not yet part of the blockchain.

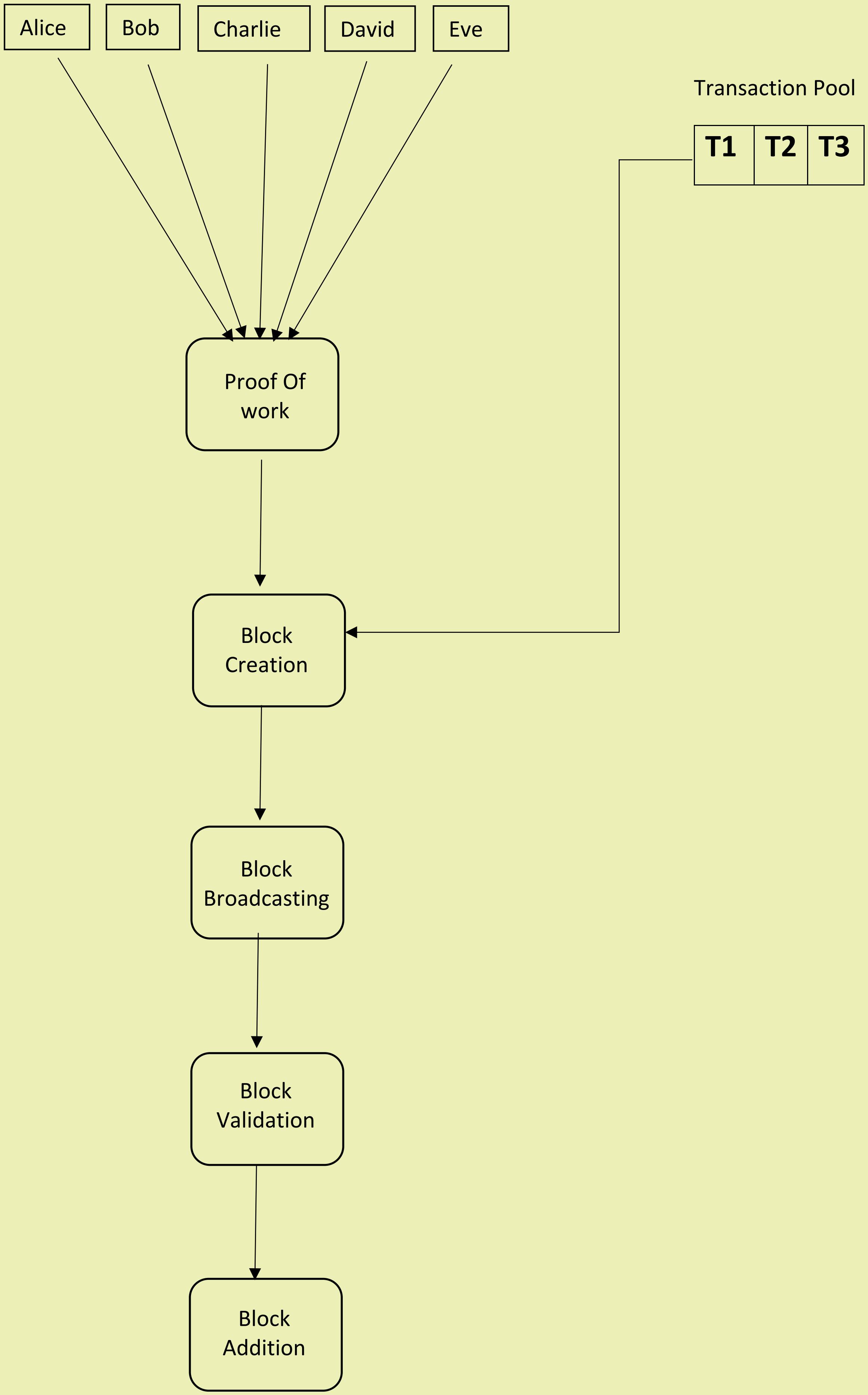
To add these transactions to the blockchain, the network participants compete to solve a difficult mathematical problem through a process called **Proof of Work**. The first participant to solve this problem within a set time frame, typically around 10 minutes, gets the right to create a new block.

Suppose Bob successfully solves the problem. Bob then creates a new block containing the transactions T1, T2, and T3 from the transaction pool. This is known as **Block Creation**.

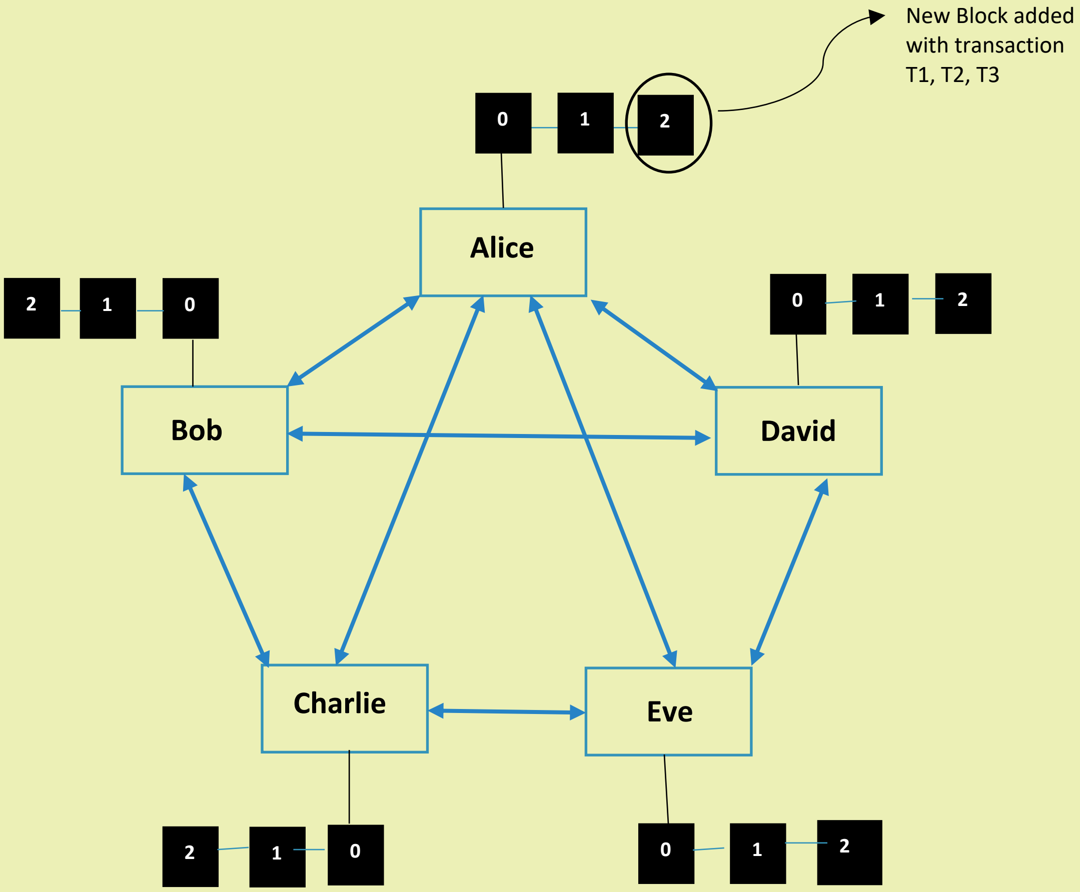
Once the block is created, Bob broadcasts it to all other participants in the network: Alice, Charlie, David, and Eve. This is known as **Block Broadcasting**.

Each participant then validates the new block by checking:
The Proof of Work**:** Ensuring the problem was correctly solved.
The Transactions**:** Verifying the legitimacy of T1, T2, and T3, ensuring that the sender has sufficient funds and that there are no <u>double-spending issues</u>. This is known as **Block Validation.**

If all validations pass, the block is considered valid. Each participant then adds the new block to their copy of the blockchain, ensuring consistency across the network. The block now becomes a permanent part of the blockchain, and the transactions it contains are confirmed. This is known as **Block Addition**.


Alice
Bob
Charlie
David
Eve
Transaction Pool
T1
T2
T3
Proof Of work
Block Creation
Block Broadcasting
Block Validation
Block Addition

Suppose this new block added is of the index 2,

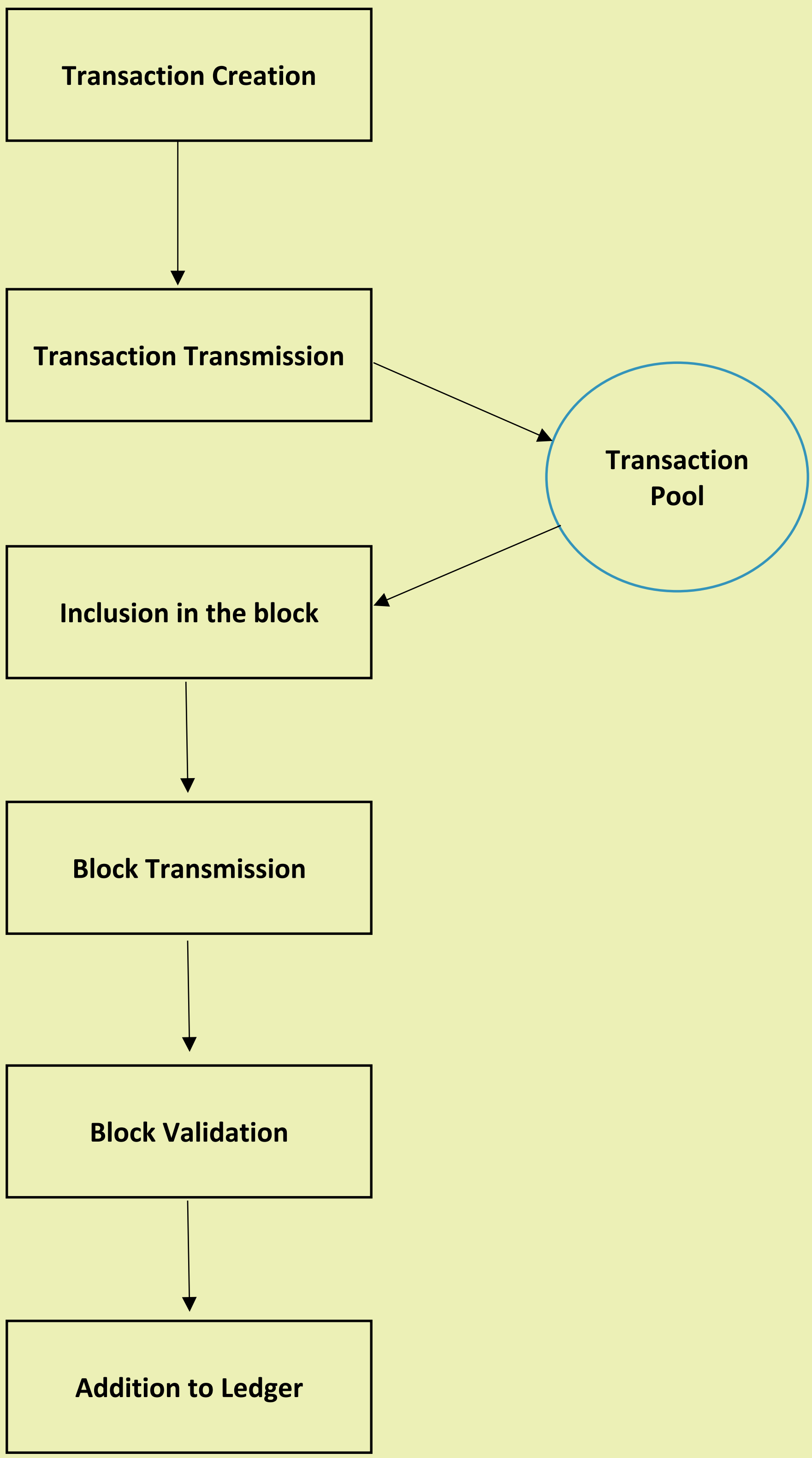


Summary of the Process:

1. **Transaction Initiation:** Transactions (T1, T2, T3) are created and placed in the transaction pool.
2. **Proof of Work:** Participants compete to solve a difficult problem. The first to solve it wins the right to create a block.
3. **Block Creation:** The winner (Bob) creates a block with the transactions.

4. **Block Broadcasting:** The new block is broadcasted to all network participants.
5. **Block Validation:** Participants validate the block and its transactions.
6. **Block Addition:** Once validated, the block is added to the blockchain, updating every participant's copy.

Life Cycle of Transaction



Case Study of Bitcoin

Origin

Bitcoin was introduced in 2008 by an anonymous entity known as Satoshi Nakamoto. In a white paper titled "*Bitcoin: A Peer-to-Peer Electronic Cash System,*" Nakamoto outlined a vision for a decentralized digital currency that would allow secure, direct transactions without relying on a central authority.

Purpose

The primary purpose of Bitcoin was to create a decentralized form of money that could operate independently of traditional financial institutions. This was particularly relevant in the wake of the 2008 financial crisis, which exposed the vulnerabilities of centralized banking systems. Bitcoin aimed to provide an alternative that was transparent, secure, and resilient against economic turmoil.

Native Cryptocurrency (BTC)

The native cryptocurrency of the Bitcoin network is *Bitcoin (BTC)*. It serves as the unit of account and medium of exchange within the network. BTC is used to reward miners who validate transactions and secure the network, and it is traded on various cryptocurrency exchanges.

Bitcoin Metrics and Market Capitalization

Bitcoin's value is determined by market demand and supply. Key metrics include:

Price**:** The current market price of one BTC. Which is **$68899.80 USD.**

Market Capitalization**:** The total value of all BTC in circulation, currently it is **$1,274.37B USD.** Calculated as (Price per BTC) x (Total BTC in circulation).

Volume: The total amount of BTC traded over a given period. The current volume is **$23.59B USD**.

Working of Bitcoin

Bitcoin operates on a decentralized ledger called the blockchain, where all transactions are recorded. The network uses a consensus mechanism called Proof of Work (PoW), in which miners compete to solve cryptographic puzzles to add new blocks to the blockchain.

Bitcoin Halving

Bitcoin's supply is limited to 21 million BTC. To control the issuance rate, the network undergoes a process called "halving" approximately every four years. During a halving event, the reward for mining a new block is cut in half. This reduces the rate at which new BTC are created, adding scarcity and potentially increasing value. The final BTC is expected to be mined around the year 2140.

Transaction Record Keeping Model: UTXO

Bitcoin uses the Unspent Transaction Output (UTXO) model for transaction record-keeping.

"Imagine you have a wallet full of coins. Each coin has a specific value. For example, you might have one coin worth $5, two coins worth $10 each, and one coin worth $20.

When you want to buy something that costs $15, you can't split a coin. Instead, you pick the coins that together make up at least $15. In this case, you could use the $20 coin. You give the $20 coin to the seller, and they give you back $5 in change."

In this model:

Inputs: Represent the sources of funds, referencing previous UTXOs.

Outputs: Represent the destinations of funds, creating new UTXOs.

A transaction consumes UTXOs as inputs and creates new UTXOs as outputs. This model ensures that each Bitcoin can only be spent once and simplifies transaction validation.

Primary Types of Bitcoin Transactions

Pay-to-PubKeyHash (P2PKH): The most common type, where the recipient's address is a hashed public key. The recipient must provide a signature and public key to spend the funds.

Multisignature (MultiSig): Requires multiple signatures to authorize a transaction. Used for added security or shared control.

Pay-to-Script-Hash (P2SH): Allows complex transaction scripts by specifying a script hash. The actual script is provided when spending the funds.

Case Study of Ethereum

Origin

Ethereum was conceptualized in late 2013 by a young programmer named Vitalik Buterin. The project officially started in early 2014, with the development team including Gavin Wood, Joseph Lubin, and others. Ethereum's initial coin offering (ICO) in 2014 raised funds to support its development, and the network went live on July 30, 2015.

Purpose

Ethereum was created to expand the capabilities of blockchain technology beyond simple financial transactions, as seen with Bitcoin. It aimed to provide a decentralized platform that could execute smart contracts—self-executing contracts with the terms directly written into code. This allows developers to build decentralized applications (dApps) that run on the Ethereum blockchain, providing new possibilities for decentralized finance (DeFi), tokenization, supply chain management, and more.

Native Currency (ETH)

Ether (ETH) is the native cryptocurrency of the Ethereum network. It is used to pay for transaction fees, computational services, and serves as the primary medium of exchange within the Ethereum ecosystem.

Developers use ETH to deploy smart contracts and interact with dApps, while users pay ETH to execute transactions and use services on the network.

Working of Ethereum

Ethereum operates on a decentralized blockchain where each node in the network maintains a copy of the blockchain. Smart contracts are executed by the Ethereum Virtual Machine (EVM), which ensures that contract code runs as intended. The network uses a consensus mechanism to validate transactions and add new blocks to the blockchain. Initially, Ethereum used Proof of Work (PoW) but is transitioning to Proof of Stake (PoS) with Ethereum 2.0.

Blockchain Metrics

Key metrics for Ethereum include:

- Price: The current market price of one ETH is INR **2,78,996.69**.Market Capitalization: The total value of all ETH in circulation which is **397.64B**, calculated as (Price per ETH) x (Total ETH in circulation)
- Volume: The total amount of ETH traded over a given period, as on 27th July it was **$10.28B**.
- Total Value Locked (TVL): The amount of assets locked in Ethereum-based DeFi protocols. Currently it is **42.74B US Dollars.**

Transaction Record Model

Ethereum uses an account-based model for transaction record-keeping.

"It works similar to credit or debit card"

There are two types of accounts:

Externally Owned Accounts (EOA): Controlled by private keys, these accounts can send transactions and initiate smart contracts.

Contract Accounts: Controlled by smart contract code, these accounts execute code when they receive transactions.

Transactions and Gas Fees

Transactions on the Ethereum network involve sending ETH or interacting with smart contracts. Each transaction requires computational resources, which are paid for using a unit called "gas."

Gas: A measure of the computational work required to execute operations, including transactions and smart contract interactions.

Gas Price: The amount of ETH a user is willing to pay per unit of gas. It is typically measured in Gwei (1 Gwei = 10^-9 ETH).

Gas Limit: The maximum amount of gas a user is willing to consume for a transaction. Users set a gas limit to prevent overspending on fees.

Example of a Transaction

When Alice sends 1 ETH to Bob, she creates a transaction specifying:

Recipient: Bob's address.

Amount: 1 ETH.

Gas Limit: e.g., 21,000 units (the typical amount for a standard ETH transfer).

Gas Price: e.g., 20 Gwei.

The total transaction fee is calculated as (Gas Limit) x (Gas Price). If the transaction consumes less gas than the gas limit, the remaining gas is refunded to Alice.

Types of Blockchains

A **permissionless blockchain** is a type of blockchain where anyone can join the network, participate in the consensus process, and read or write transactions. These blockchains are open to the public and do not require any permission from a central authority. They are also known as public blockchains.

Bitcoin and Ethereum are the most well-known examples of permissionless blockchains. Anyone can participate in mining (for Bitcoin) or validating transactions, and all transaction history is publicly accessible. Imagine a giant library where anyone can enter, read any book, or even add new books to the collection without asking for permission. Everyone can see what books are there and who has borrowed them.

A **permissioned blockchain**, on the other hand, is a type of blockchain where the participants need approval to join the network. These blockchains are typically used by organizations that require more control over who can access the network and who can read or write transactions.

Hyperledger Fabric is an example of a permissioned blockchain. It is used by businesses and organizations that need to maintain privacy and control over their blockchain network. Think of a private club where only members who have been approved by the club's board can enter. Inside the club, only members can participate in activities, and some information is only shared among members.

Layers of Blockchains

Layer 0 is the foundational layer that encompasses the underlying infrastructure of the entire blockchain network. It helps to create custom blockchain.

Polkadot **is an** example of Layer 0 platform that enable different blockchains (Layer 1) to interoperate and communicate with each other.

Layer 1 refers to the base layer of the blockchain itself. It includes the core protocols and consensus mechanisms and functionality like decentralization and security.

Ethereum and Bitcoin are Layer 1 blockchains. Ethereum's Layer 1 provides a secure and decentralized platform for running smart contracts and decentralized applications (dApps).

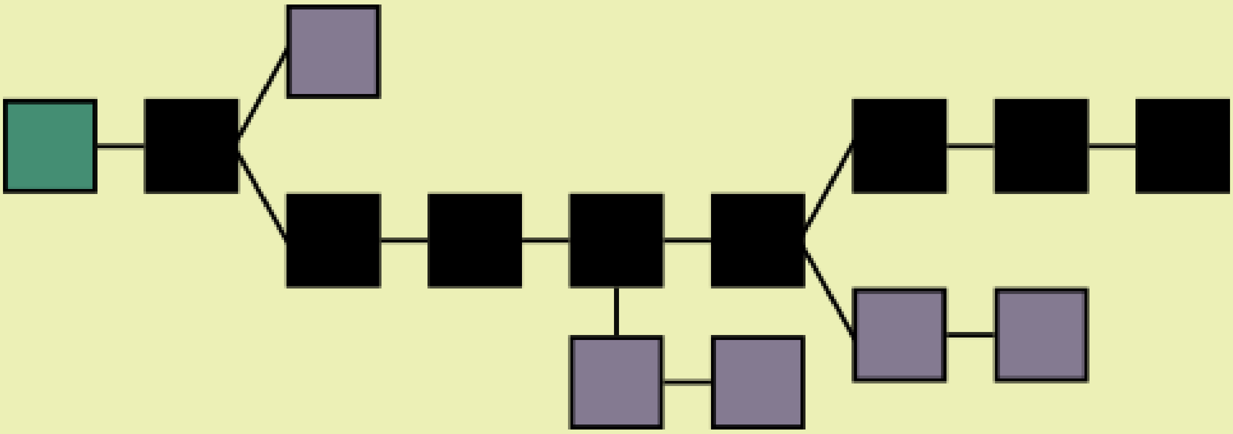
Layer 2 solutions are built on top of Layer 1 blockchains to address scalability issues. They enable faster and more efficient transactions by handling most operations offchain.

Polygon is a prominent Layer 2 solution for Ethereum. Polygon improves Ethereum's scalability by processing transactions off-chain and then bundling them back onto Ethereum's Layer 1. This allows for higher transaction throughput and lower fees.

Blockchain Forks

When a blockchain splits into two or more blockchain, it is known as fork. There are types of forks that happens in a blockchain:

1. Soft fork
2. Hard Fork



A **soft fork** is a backward-compatible upgrade to the blockchain protocol i.e. network continues to function smoothly even if some nodes are still using the old method.

The Bitcoin Soft Fork Case Segregated

The Segregated Witness (SegWit) upgrade in Bitcoin is a soft fork. Segregated Witness (SegWit) separates the signature data from the transaction data. This makes the transactions smaller and more efficient, allowing more transactions to fit into each block.

This change helps the Bitcoin network process more transactions faster. Over time, more and more people adopt the new system, but the network continues to function smoothly even if some nodes are still using the old method.

A **hard fork** is a non-backward-compatible upgrade that requires all nodes to upgrade to the new protocol. Nodes that don't upgrade will no longer be part of the same network.

The Ethereum Hard Fork Case

Background: The DAO Hack:

- **The DAO**: In 2016, a decentralized autonomous organization (DAO) called "The DAO" was created on the Ethereum blockchain. It was essentially a venture capital fund.
- **Hack**: A hacker exploited a vulnerability in The DAO's code and siphoned off about $50 million worth of Ether (Ethereum's cryptocurrency) from the fund.

The Fork Decision:

To address the hack, the Ethereum community had two choices:

- **Do Nothing**: Leave the blockchain as it is, accepting the hack and the loss of funds.
- **Hard Fork**: Implement a change in the blockchain to recover the stolen funds.

The Hard Fork:

The community decided to go with the hard fork and created two separate blockchains and cryptocurrencies:

- **Ethereum (ETH)**: This is the new chain where the effects of the hack were reversed. The stolen funds were returned to their original owners.
- **Ethereum Classic (ETC)**: This is the original chain that continued without any changes, keeping the hacker's transactions intact.

Blockchain Bridges

Imagine you live in the India and decide to go on a vacation to France. In the India, we use Indian Rupee, while in France, they use Euros (EUR). To spend money in France, you have two options:

1. You can go to a bank or a currency exchange service to convert your USD to EUR before you travel.
2. You can use an international credit card that automatically converts USD to EUR when you make purchases in France.

“This process is straightforward in the world of banks, where we can convert one currency with other, but it’s much more complex in the world of blockchains.”

Let's take two different blockchains: **Ethereum** and **Terra**. They operate independently, with their own rules and consensus mechanisms. The native token of Ethereum Mainnet is **ETH**, while the native token of Terra is **LUNA**.

Now, suppose you have a lot of LUNA on Terra, but you want to use it on the Ethereum Mainnet. How do you do that?

Then to solve this problem, **blockchain bridge** comes into play. A blockchain bridge allows tokens from one blockchain to be used on another blockchain

You use a blockchain bridge, like the Terra Bridge, to convert your LUNA into Wrapped Luna (WLUNA). Wrapped Luna is an ERC-20 token, which means it is compatible with the Ethereum Mainnet.

Blockchain Tokens

There are various tokens built on blockchain. Some of them are:

- Cryptocurrencies
- Stablecoin
- Utility Token
- Security Token
- Non-Fungible Token

Cryptocurrency Tokens are used to buy and sell products/services and can be easily converted to cash.

- Bitcoin (BTC)
- Ethereum (ETH)
- Litecoin (LTC)
- Etc

Utility Tokens works provide access to a product or service within a specific blockchain ecosystem.

- Basic Attention Token (BAT): Used within the Brave browser ecosystem to reward users for viewing ads and content creators for producing content.
- FileCoin (FIL): reward users for providing decentralized storage.

Security Tokens represents ownership or a stake in real world assets such as equity, or real estate and are compliance under laws.

- Polymath (POLY): A platform for issuing and managing security tokens, representing ownership of assets like real estate or stocks.

Stablecoins are generally pegged to stable assets like gold, fiat currencies to reduce volatility.

- Tether (USDT): A stablecoin pegged to the US Dollar, used to provide liquidity and stability in the cryptocurrency market.
- USD Coin (USDC): Another US Dollar-pegged stablecoin, commonly used for trading and as a store of value.

Governance Tokens grants holders the right to vote on protocol changes and decisions within blockchain project.

- Uniswap (UNI): Allows holders to vote on changes and upgrades to the Uniswap decentralized exchange protocol.
- Maker (MKR): Used in the governance of the MakerDAO protocol, allowing holders to participate in decisions.

Non-Fungible Tokens (NFTs) are unique digital assets representing ownership of specific items, collectibles, content.

- CryptoKitties: Digital collectibles representing unique virtual cats, each with distinct characteristics and ownership recorded on the blockchain.
- Decentraland (MANA): Represents ownership of virtual real estate within the Decentraland metaverse.

Blockchain Use Cases

Finance

- **Cross-Border Payments:** Faster and cheaper international transactions (e.g., Ripple).
- **Decentralized Finance (DeFi):** Financial services like lending, borrowing, and trading without intermediaries (e.g., Compound, Uniswap).

Healthcare

- **Medical Records:** Secure and interoperable patient data management (e.g., MedRec).
- **Drug Traceability:** Preventing counterfeit drugs by tracking the supply chain (e.g., Chronicled).

Government

- **Voting Systems:** Secure and transparent electronic voting (e.g., Voatz).
- **Identity Management:** Digital identity verification and management (e.g., uPort).

Supply Chain

- **Product Provenance:** Tracking the origin and journey of goods to ensure authenticity (e.g., IBM Food Trust).
- **Inventory Management:** Real-time monitoring and automation of supply chain processes (e.g., VeChain).

Entertainment

- **Digital Rights Management:** Ensuring fair distribution and monetization of digital content (e.g., Ujo Music).
- **Fan Engagement:** Creating unique experiences and rewards for fans through NFTs (e.g., NBA Top Shot).

Blockchain Fundraising Models

There are various token distribution Models, Some of them are:

Initial Coin Offering (ICO) companies raise capital by selling a new cryptocurrency token to early backers in exchange for established cryptocurrencies like Bitcoin or Ethereum.
Example: Ethereum's 2014 ICO, which raised funds to develop the Ethereum platform.

Reverse ICO an established company raises funds by issuing a new cryptocurrency token to diversify its funding sources or transition to blockchain technology.
Example: Telegram's TON (Telegram Open Network) reverse ICO aimed at expanding its services.

Airdrop free distribution of cryptocurrency tokens to a large number of wallet addresses to promote awareness and adoption.
Example: Stellar's airdrop of XLM tokens to Bitcoin holders to encourage use of the Stellar network.

Initial Exchange Offering (IEO), tokens are sold directly on a cryptocurrency exchange, which acts as an intermediary and ensures a certain level of credibility and security.
Example: Binance Launchpad's IEO for BitTorrent Token (BTT), which raised funds on the Binance exchange.

Initial DEX Offering (IDO), they are same like ICO and IEO, only difference the funding is based on decentralized exchange, without relying on central exchange where there is custodial wallet.
Example: Uniswap's IDO for SushiSwap (SUSHI) tokens, which enabled decentralized trading and liquidity provision from the start

Security Token Offering (STO), tokens representing ownership in an underlying asset, such as equity or real estate, are sold to investors and are subject to regulatory compliance.
Example: tZERO's STO, offering security tokens representing shares in the tZERO platform.