Blockchain Programming

Dr. Adnan IMERI
Technical Lead, INFRACHAIN and Luxembourg Blockchain Lab (LBL)

Disclaimer: This training intends explanation of technological advancements, and it does not intend any financial advice toward cryptocurrency or any digital asset!

Practical Information

- Course Objective
  - To give a broad intro to BC prog. and guide the audience **through** first **hands-on** Smart Contract and Decentralized Application
- Course Organization
  - 17:30 - 19:30 PM, via Teams (Online only)
  - 9 Sessions
- Learning Outcomes
  - Introduction to Blockchain
    - Concepts
    - Programming
    - Tools
    - DApp and Web3
- Resources:
  - Team Post
  - Literature suggested during the course
- Contact
  - Send me an email: **adnan.imeri@infrachain.com**

Practical Information

- Getting Familiar with Working Environment
- Teams
- Virtual Machines
- Self-Working Environment

Dev Stack for blockchain programming

1

Install Node.js:

https://nodejs.org/en/download/

2

Install Visual Studio Code:

https://code.visualstudio.com/download

3

Verify node version :

node –version

4

Create a new project from Visual Studio Code

- npm init: To create package.json file
- npm install: To install full node nodules

Table of Content (Session 1)

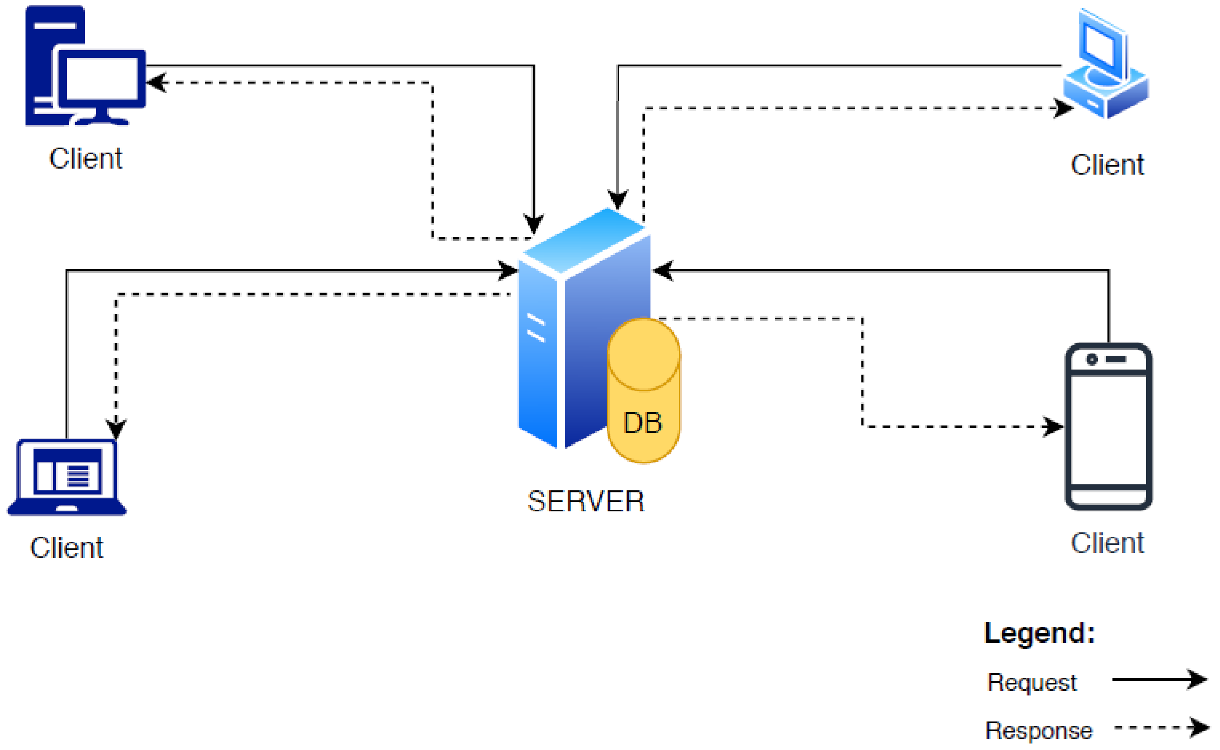
Technology Perspective

- Centralized
- Decentralized
- Decentralized-Distributed

Centralized: Client-Server Architectures

- Client-server (CS) architecture comprises two software processing sides: **server-side** and **client-side**.
  - The **client-side** allows users to query formalization, send a query on the **server-side**, and receive a response (query result) from the **server-side**.
  - The **server-side** stores and manages user data, processing application data, and user queries.

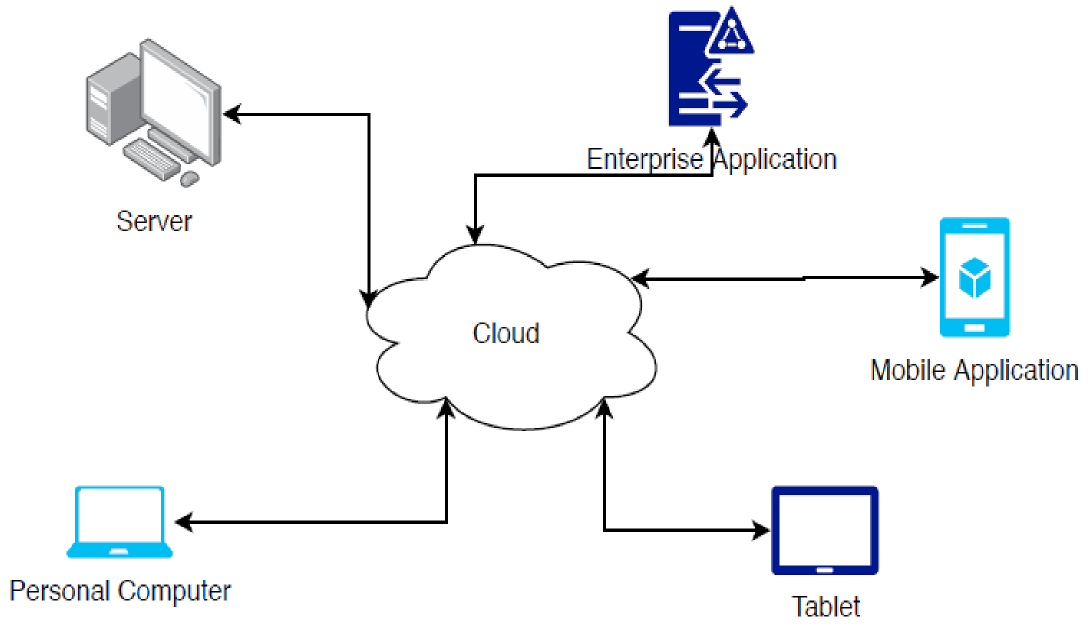
- Advantages:
  - Scalable
  - Mature technology
  - Accessible (easy programmable)
- Disadvantages:
  - Single point of failure
  - Trust and Transparency issues.
    - data loss, data altering, and data integrity.
  - Expensive to setup and maintain



Distributed

- Cloud Computing (CP)
  - Enabled moving from client-server into distributed architectures (remote operation system, software, hardware)
  - Software as a Service (SaaS)
  - Infrastructure as a Service (IaaS)
  - Application as a Service (AaaS)
  - …
- Internet of Things (IoT)
  - IoT describes a set of devices that are able to collect, exchange,

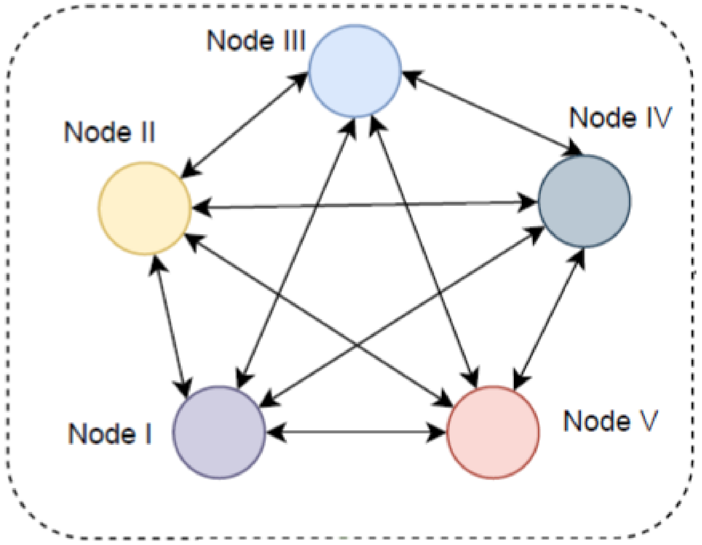
IoT



Cloud Computing

Decentralized-Distributed

- No centralized authority to maintain information
- Peer-to-peer communication protocol
- Network of nodes (computers/servers/low power devices)
- Same copy of **data** in all nodes
- Consensus algorithm



In Nutshell

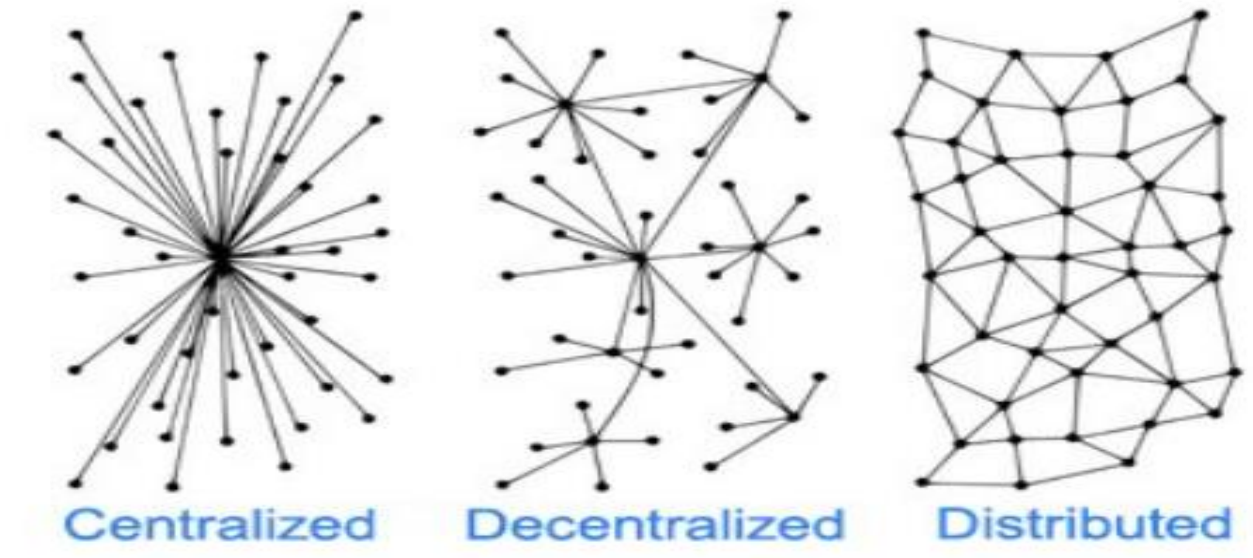
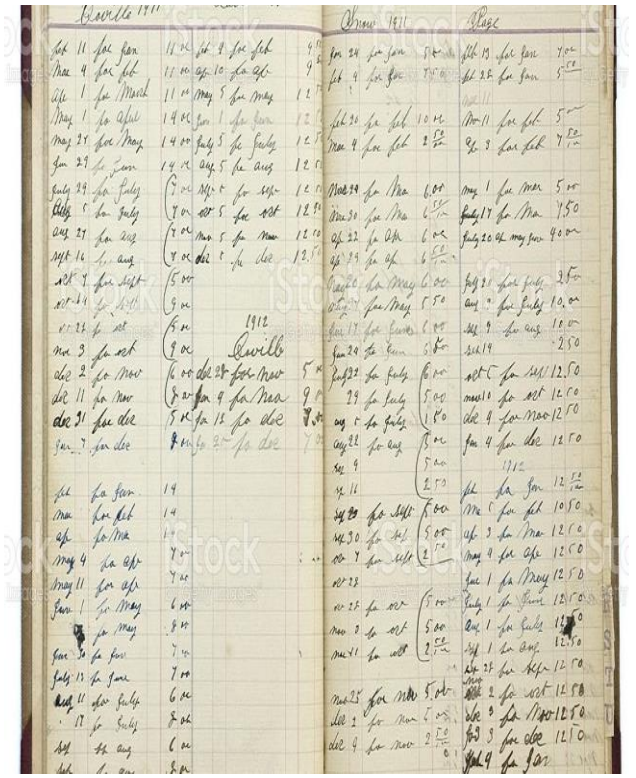


Image source: Link

Distributed Ledger Technologies



Ledgers

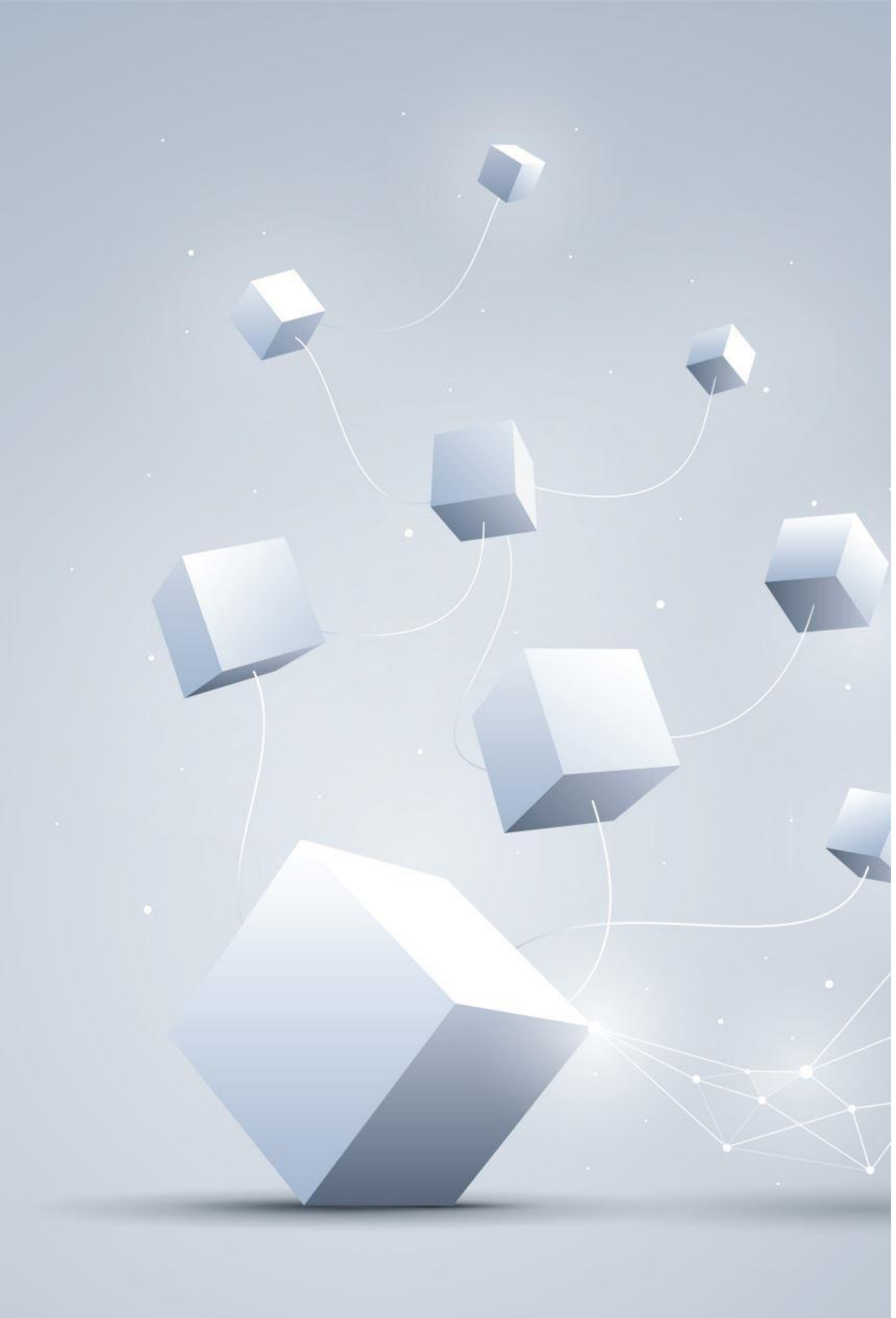
"Ledgers have existed since ancient times and have served as record-keeping of transactions" [1]

- a pen and paper ledger
- a large database maintained by a central authority e.g., Banks

"Distributed ledger technology (DLT) presents a distributed and decentralized database, shared among multiples parties, known as network participants"

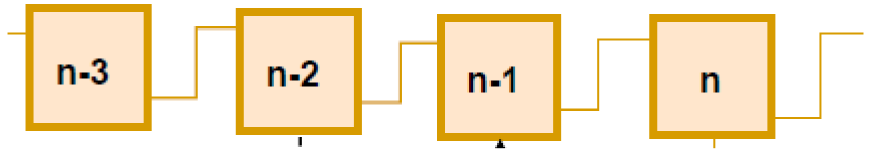
Blockchain

BC technology is an instance of the distributed ledger



Blockchain

- Blockchain (BC) is a distributed decentralized database that allows storing **immutable** cryptographically **signed** transaction data.
- Transaction data are gathered into **blocks** and **chained** together with the previous block, thus forming a **blockchain.**



Example:

a) Bitcoin: https://www.blockchain.com/btc/blocks?page=1

b) Ethereum: https://etherscan.io/blocks

Blockchain Technology Components

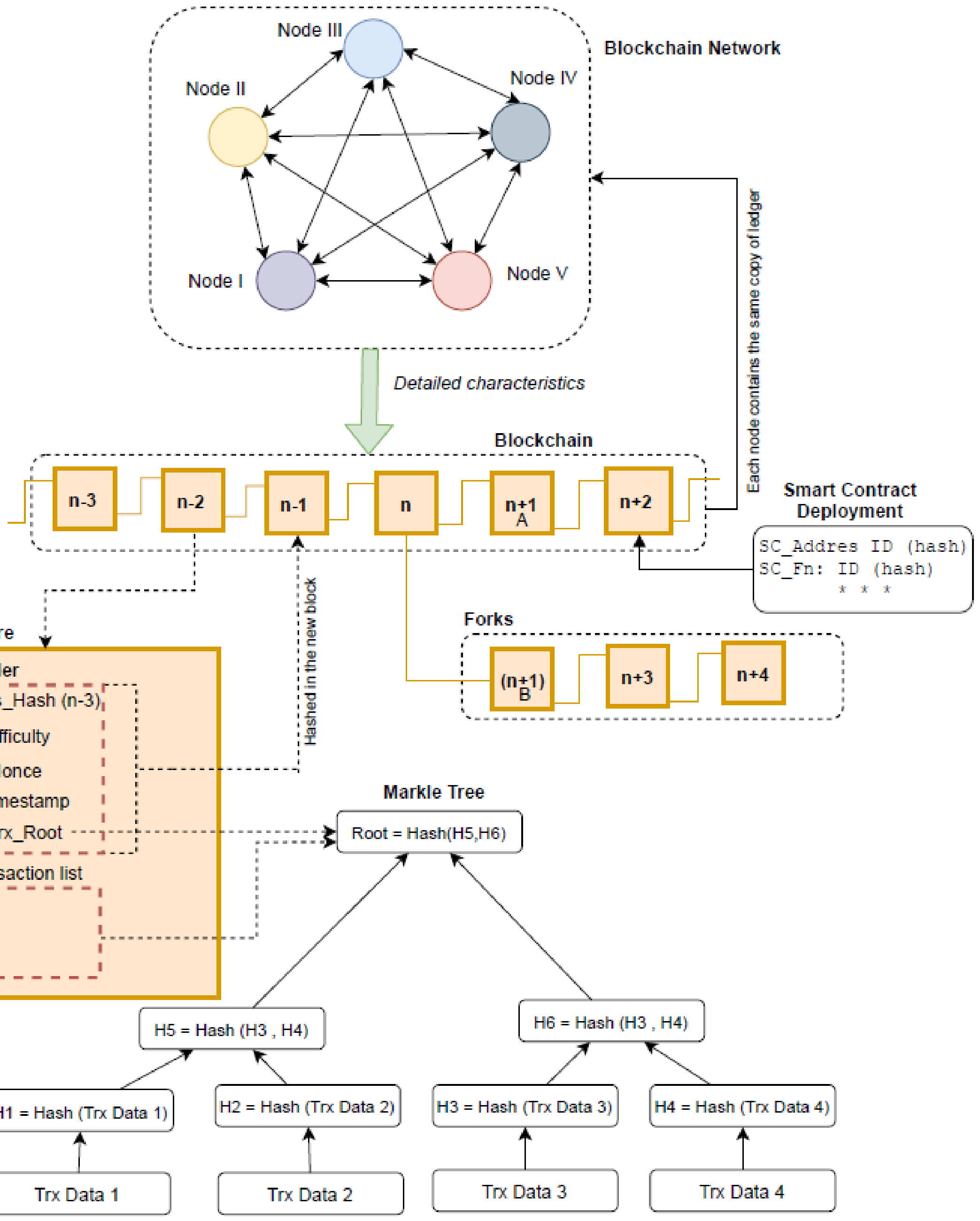


Image source [1]

Blockchain Technology Components

- Blockchain Network
  - Composed of several nodes that are distributed geographically
  - Peer-to-Peer communication
  - Rely on consensus algorithms
  - No central node/cluster/cloud
- Nodes
  - Computer/Servers that store and maintain the distributed ledger

  - Network Visualization: https://bitnodes.io/nodes/network-map/
  - https://dailyblockchain.github.io/
  - https://mempool.space/
- Source: https://news.bitcoin.com/18-visualizations-bitcoin-network/

Blockchain Technology Components

- Hashing
  - Cryptographic hash is a one-way function that yields a fixed-length string for the given input.

  has256(blockchain) → ef7797e13d3a75526946a3bcf00daec9fc9c9c4d51ddc7cc5df888f74dd434d1

- Working with hash: /dev/Hashing.js

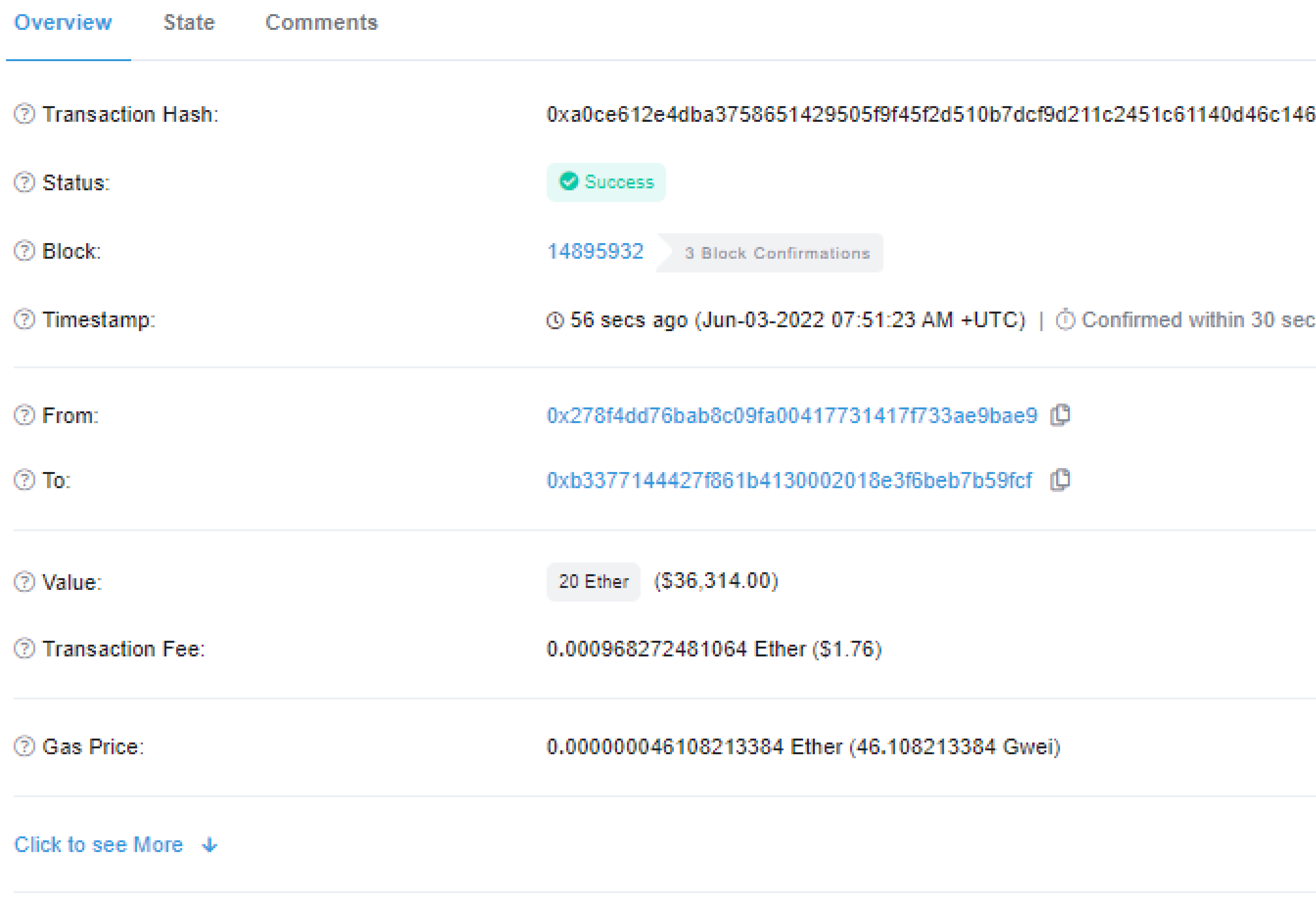
- Addresses in the blockchain:
  - Is derived from their public key + auxiliary data by using the hash function over it!
  - Types of Addresses:
    - User address (Wallet)
    - Smart Contract address

- Source: https://www.geeksforgeeks.org/passwords-and-cryptographic-hash-function/
- Hash Generator: https://passwordsgenerator.net/sha256-hash-generator/

Blockchain Technology Components

- Transaction
  - Any interaction/action in blockchain is done through transaction
  - [Explore transactions]

Timestamp: The time when the block/the transaction is executed (mined)



Overview State Comments

Transaction Hash:	0xa0ce612e4dba3758651429505f9f45f2d510b7dcf9d211c2451c61140d46c146
Status:	Success
Block:	14895932 3 Block Confirmations
Timestamp:	56 secs ago (Jun-03-2022 07:51:23 AM +UTC) | Confirmed within 30 sec
From:	0x278f4dd76bab8c09fa00417731417f733ae9bae9
To:	0xb3377144427f861b4130002018e3f6beb7b59fcf
Value:	20 Ether ($36,314.00)
Transaction Fee:	0.000968272481064 Ether ($1.76)
Gas Price:	0.000000046108213384 Ether (46.108213384 Gwei)

Click to see More

Digital Signature

- Public Key Infrastructure
  - Asymmetric cryptography is an encryption schema that provides two different mathematical related keys: public key and private key
  - Public key ≠ private key
  - Public key is publicly shared
    - Used to encrypt the message
    - Publicly shared
  - Private key (digital signature)
    - Digital Signature to prove that the "identity of the person"
    - Used to decrypt the message
    - Private (not to share)

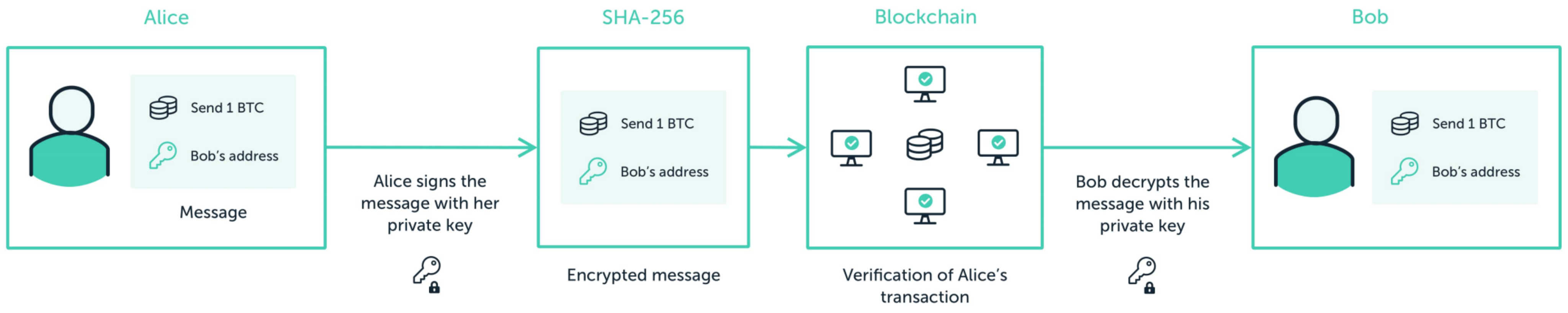

Alice
Send 1 BTC
Bob's address
Message
Alice signs the message with her private key
SHA-256
Send 1 BTC
Bob's address
Encrypted message
Blockchain
Verification of Alice's transaction
Bob decrypts the message with his private key
Bob
Send 1 BTC
Bob's address

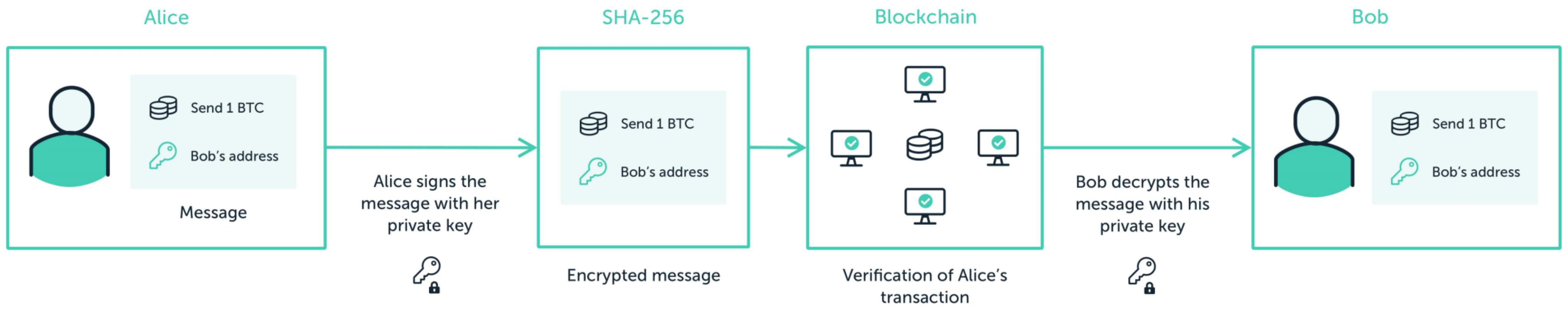
Image source [3]

Digital Signature

- In the blockchain, **Digital Signature** is fundamental and used to authenticate transactions. At any time when the user submits transactions, they must prove to every node in the system that they are authorized to spend those funds [5].

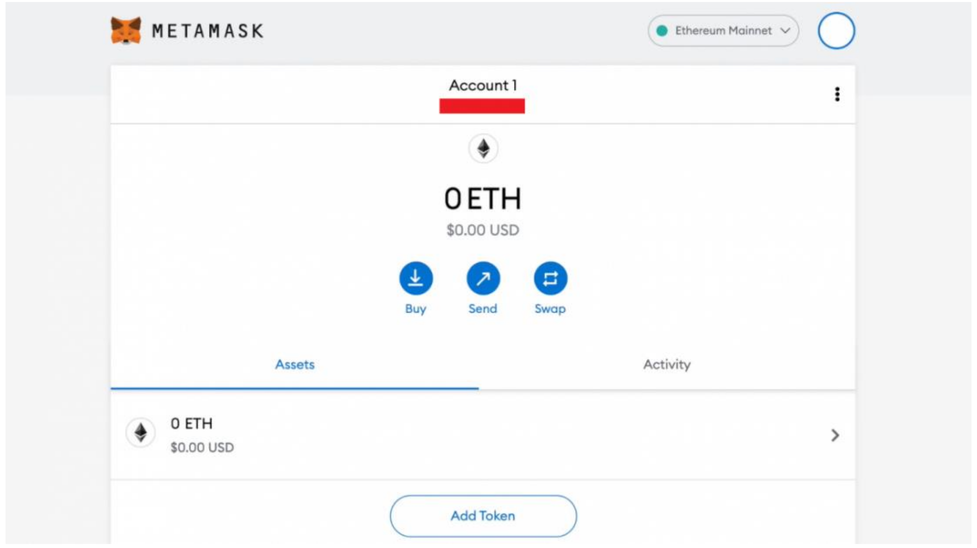
  Public-private key generation
  - /dev/PKIGen.js

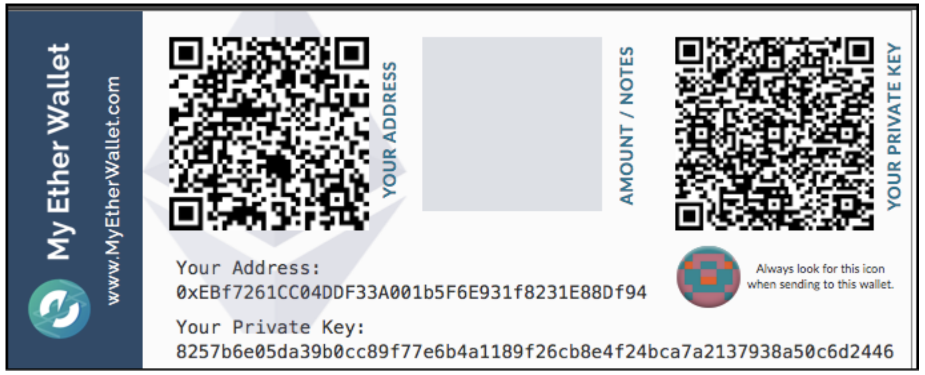
- Example: Encrypt and decrypt text using Public-key cryptography
  - /dsev/EncryptDecrypt.js


Alice
Send 1 BTC
Bob's address
Message
Alice signs the message with her private key
SHA-256
Send 1 BTC
Bob's address
Encrypted message
Blockchain
Verification of Alice's transaction
Bob decrypts the message with his private key
Bob
Send 1 BTC
Bob's address

Wallets

- Public and Private Key long “string” and hard to keep in mind
- Wallets:
  - Software packages that securely store private, public, and other related addresses
- Wallet types
  - Hot Wallet
  - Cold Wallet
  - Paper-based wallet




My Ether Wallet
www.MyEtherWallet.com
YOUR ADDRESS
AMOUNT / NOTES
YOUR PRIVATE KEY
Always look for this icon when sending to this wallet.

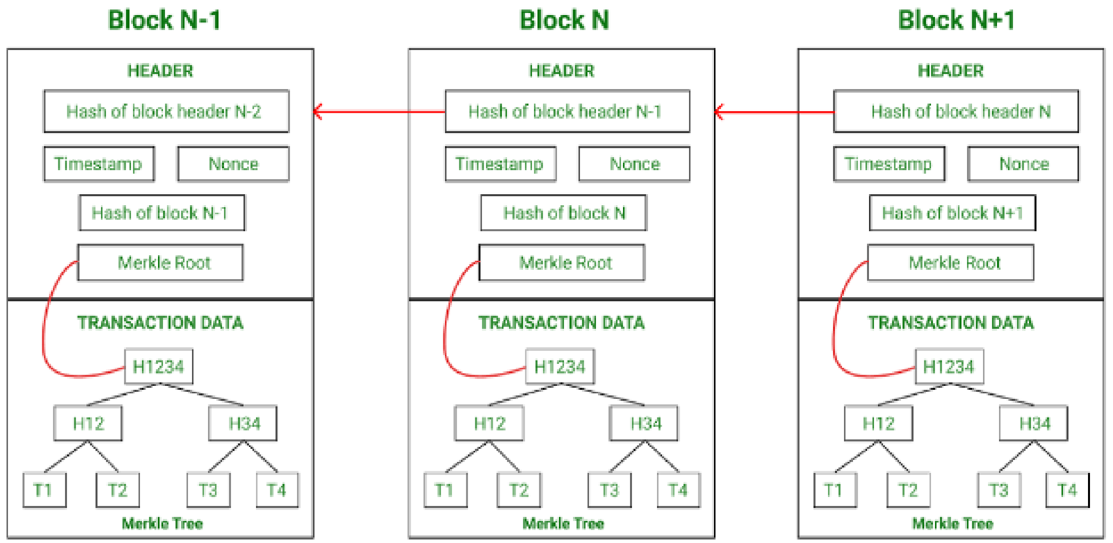
Your Address:
0xEBf7261CC04DDF33A001b5F6E931f8231E88Df94

Your Private Key:
8257b6e05da39b0cc89f77e6b4a1189f26cb8e4f24bca7a2137938a50c6d2446

Blockchain Technology Components

- Block
  - Header
    - Hash of the previous block header
    - Hash of the current block
    - Timestamp
    - Cryptographic nonce
    - Merkle root

  - Body
    - Transactions



Example:

a) Bitcoin: https://www.blockchain.com/btc/blocks?page=1

b) Ethereum: https://etherscan.io/blocks

Source: https://www.geeksforgeeks.org/blockchain-merkle-trees/

Blockchain Technology Components

- Markel Tree
  - Crucial role in verifying the integrity of transactions stored on the blockchain
  - Modifying information in one node is denied once comparing hashes in Merkle Tree

- How Merkle Tree works?

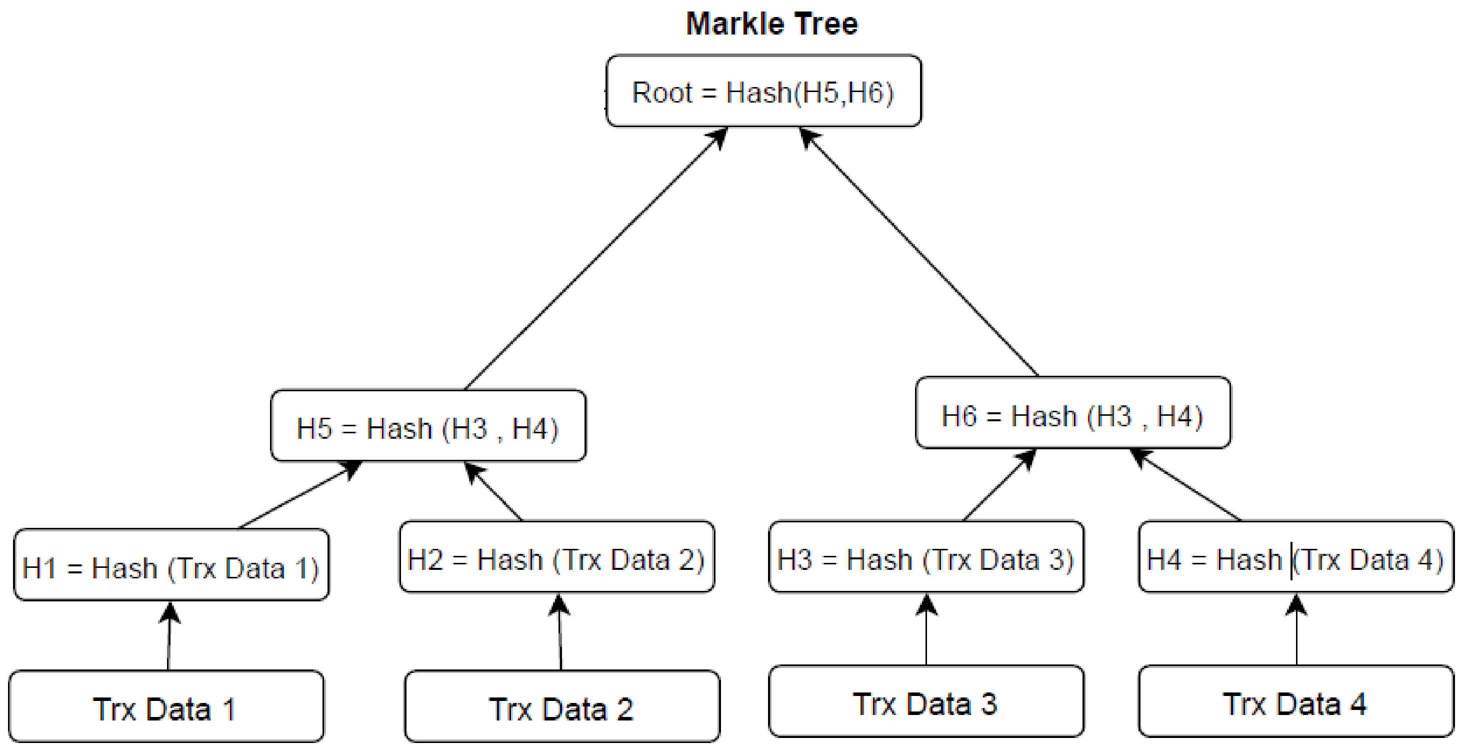
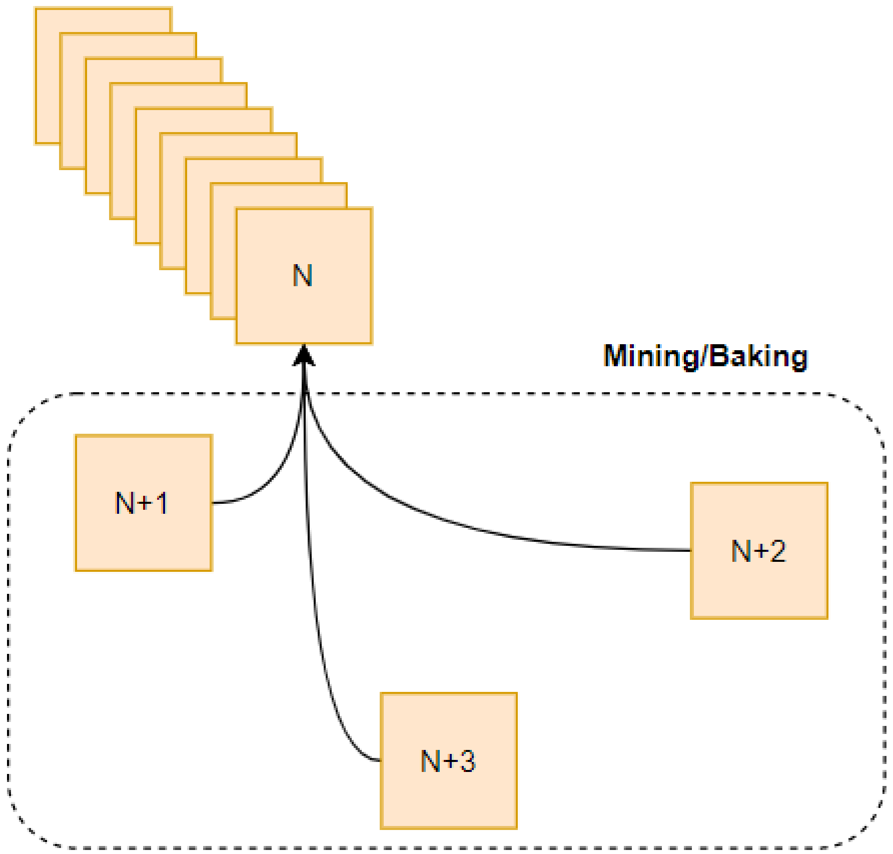


Image source [1]

Blockchain Technology Components

- **Mining/Baking**: The process of adding a new block to the chain is called mining (or baking).
- Miners are powerful computers that compete to solve a cryptographic puzzle, to mine a block
  - Miners (winner) gets a reward for adding a new block
  - More mines → Stable network



Blockchain Technology Components

- Consensus:
  - Many nodes/computers agree on the state (pieces of information) of data
  - Maintain the state of the distributed-decentralized ledger
  - In case some nodes try to modify data in their local ledger, consensus compares this data with N other nodes and denies changes → <u>Data Integrity!</u>
- Types of Consensus:
  - Proof of Works
  - Proof of Stake
  - Proof of Authority
  - Etc.

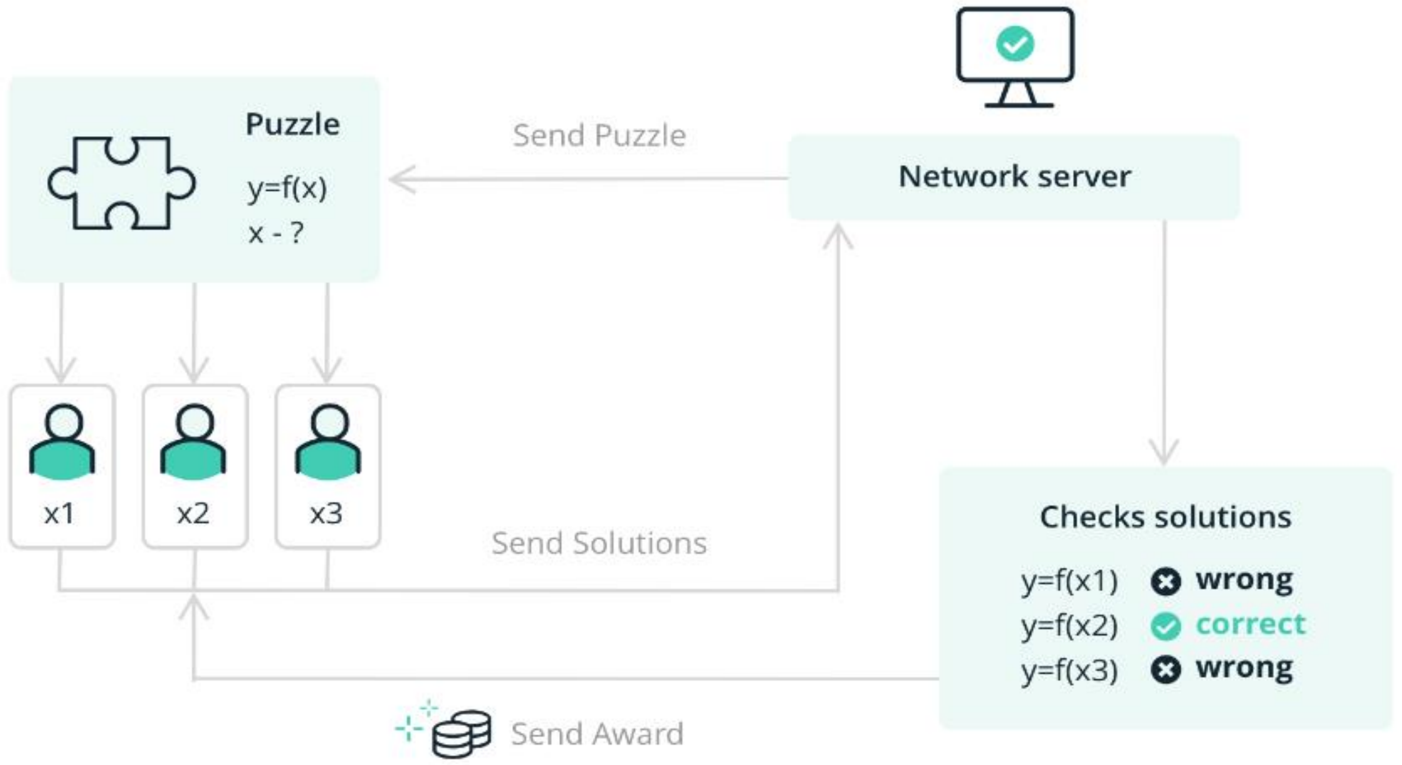


Image source [3]

Smart Contract

- Smart Contract is a computer code that is deployed on the blockchain and is triggered once some conditions are fulfilled
- Programmability of Smart Contract
  - Structure
  - Name/Address
    - Methods (functions)
  - Programming Languages
    - **Solidity**
    - JavaScript/Typescript
    - Java
    - Kotlin
    - Rust
    - Etc.

Benefits of using Blockchain Technology

- Trust and Transparency
- Data Security
- Data Immutability
- Auditability/Timestamped
- Smart Contracts (SC)

Blockchain Technology Components

- Install
  - Wallet Meta Mask
  - Get some eth (Sopholia)
    - https://cloud.google.com/application/web3/faucet/ethereum/sepolia
    - https://faucet.egorfine.com/
  - Explore Remix:
    - https://remix.ethereum.org/
    - Deploy simple SC

```
pragma solidity ^0.5.0;

 contract FistSmartContract {
   constructor() public{ }
       function getResult() public view returns(uint) {
                           uint x = 1;
                           uint y = 2;
                           uint result = x + y;
                           return result;
                       }
}
```

References

- [1] Imeri, Adnan. *Using the Blockchain Technology for Trust Improvement of Processes in Logistics and Transportation*. Diss. University of Luxembourg, Esch-sur-Alzette, Luxembourg, 2021.
- [2] Public-key Cryptography: https://www.ledger.com/academy/blockchain/what-are-public-keys-and-private-keys
- [3] Proof of Work: https://www.ledger.com/academy/blockchain/what-is-proof-of-work/
- [4] https://www.sohamkamani.com/nodejs/rsa-encryption/
- [5] https://www.coinbase.com/cloud/discover/dev-foundations/digital-signatures