Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

PROJECT-1
COMPUTER NETWORKS
(CSU33D03)

SUBMITTED TO:
Dr. Ciaran Mc Goldrick

SUBMITTED BY:
PRACHI SINGHROHA
(Student ID: 21355131)

Contents

Preferred focus and Use Case

Focus of this project is to understand and implement a basic blockchain model. The project is based on decentralised peer-to-peer network and can be used for transactions without the interference of any third party. This will give the entire control over the data to the user keeping user secure from any data breach. The main aim of the project is to create network which prioritises security of personal information of the user by giving each user a unique and fictitious identity. The system can be used for the following things:

1. Users can send coins to each other by performing transactions
2. Users can get new coins after mining process

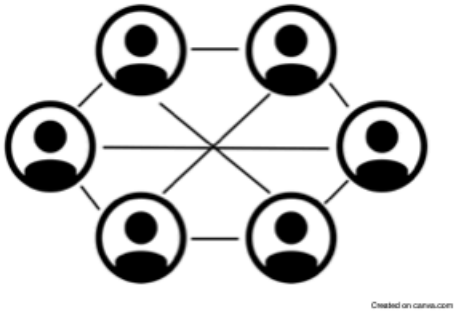


Fig1. Peer-to-Peer Network

(It is not necessary that all users interact with each other. They can choose who to do transactions with while keeping their data safe.)

Protocol Overview

The network is based on TCP/IP (Transfer Control Protocol/ Internet Protocol) connections. We'll create a blockchain which needs hash from the previous block which gives the blockchain immutability. To start this there should at least be one block that has to be created manually which will be called the GENESIS BLOCK. The transactions can be done by sending a request to the url which then will then be stored in a list.

Proof of Work, one of the consensus protocols, is also used to build the network which is based on computing the hash values and validating the transactions until a specified number of trailing zeros are found in the hash value. The reason to use this here is to discover a number which solves a problem while the user is mining.

Communication Model

Say we have added nodes in our network. They can send request to nodesRegister, explained later in the report, to know who all are part of the network. For transactions between the users, one user need to send POST request to the url"/transactions", which will then extract the data from this request and store it in the list called transactions. For the user to mine it simply has to use the command url"/mine" which will automatically do the following things:

- get proof of work from last blockchain used and use function to find new proof number
- Add new transaction, reward by blockchain to miner, to the list
- Create new block with all current transactions and append it to the blockchain list
- Clear all transitions list and return blockchain list to the miner

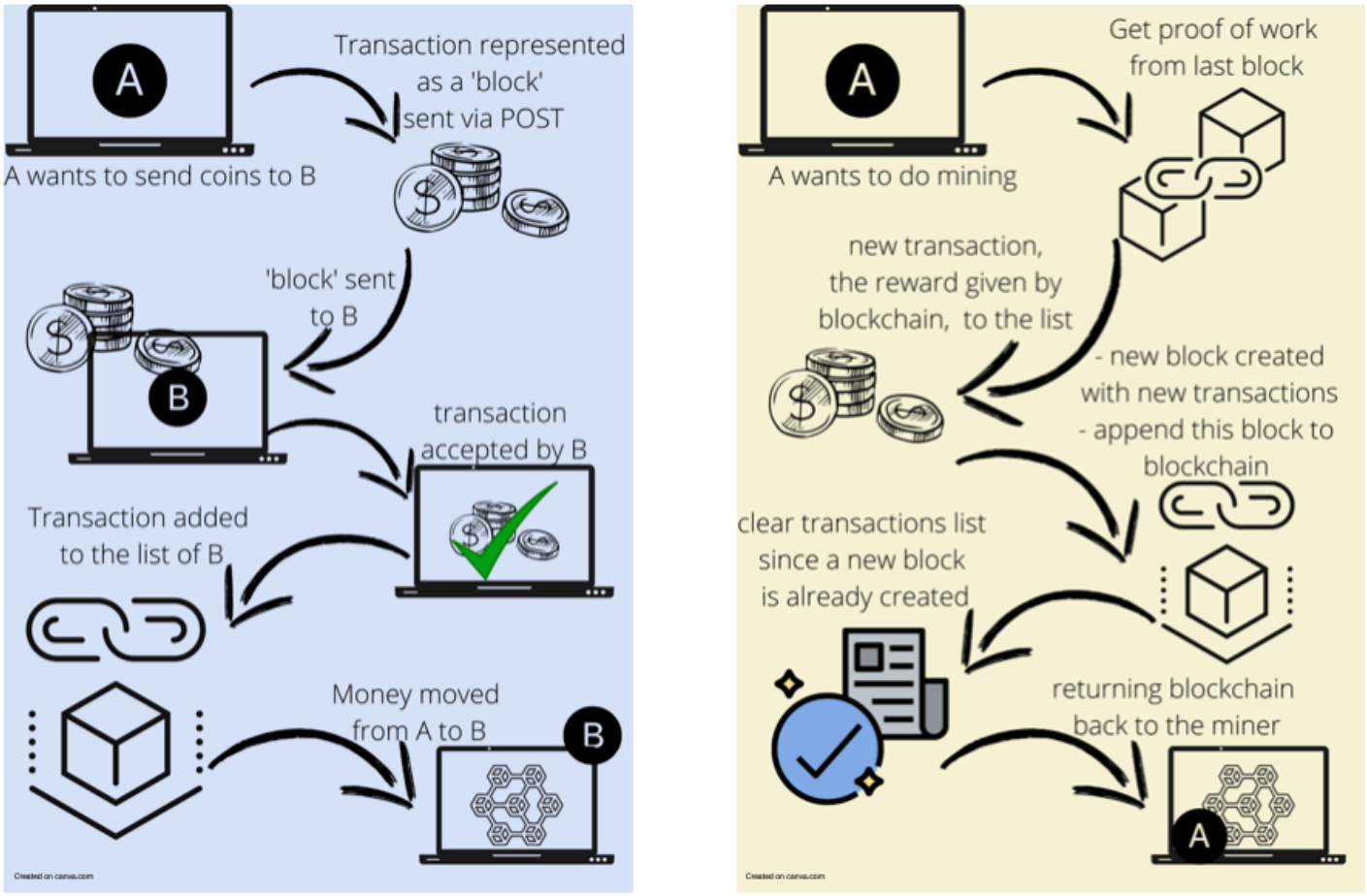


Fig2. Transactions and mining communication models

Module Descriptions

Handshake mechanism: To communicate between the nodes while performing transactions and mining which makes the transmit of data between the nodes reliable.

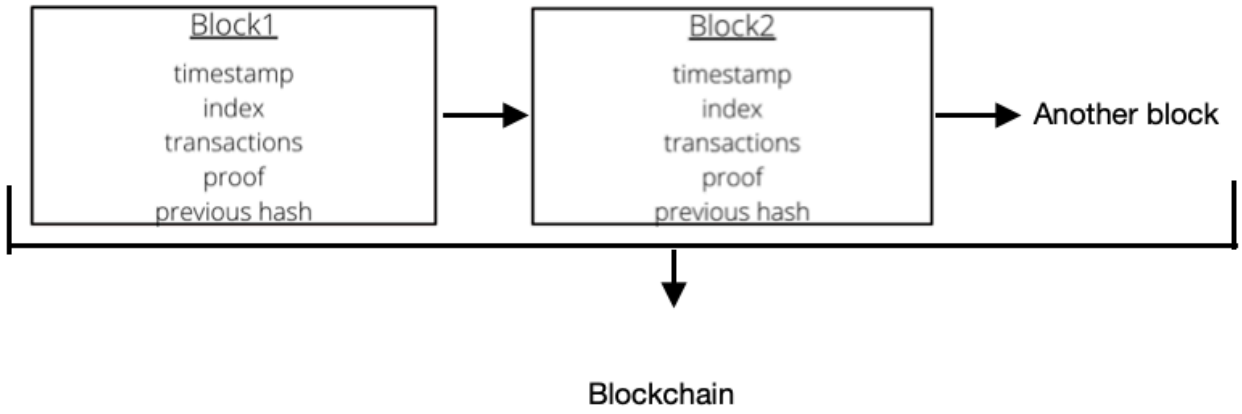
Proof of work is how blocks are mined n the blockchain. It is used discover a number which solves a problem while the user is mining. Its core ides is to ensure that the number is difficult to find but easy to verify by anyone on the network.

Hash encoded blocks: each block created contains within itself, hash of the previous block giving blockchain immutability i.e. if an attacker attacker corrupted a previous block in the chain then all the following blocks will have incorrect hashes. It makes each block unique and make the system more secure

Summary of Algorithms

1. Build a blockchain: create a blockchain *class* which creates an empty list, new blocks and a function to add transactions to the block. It also uses sha256 function from Python library to hash all of (index, timestamp, data, previous_hash) into one single hash which will be the hash of this block. This class is responsible for managing the chain. Proof of work algorithm will also be implemented under this class which will look as follows

```
// implementing proof of work
//validProof used below will validate he proof with respect to the hash function
        while self.validProof(lastProof, proof) is False:
                proof += 1
        return proof
```



2. Blockchain as API: use of python flask framework which makes it easy to map endpoints to Python functions and allows us talk to our blockchain over the web using HTTP requests. Following methods will be created:

- new transactions: adding new transactions to a black. To do this we'll modify the already formed transactions method in the blockchain class.
- mine: to mine a new block. This method will perform all the things mentioned in communication model on the previous page.
- chain: to return full blockchain

3. Consensus: this algorithm will make our system decentralised.

To implement this algorithm we'll need the following endpoints:

- nodesRegister: to accept a list of new nodes in the form of URLs
- nodesResolve: for the implementation of consensus algorithm which resolves any conflicts. To resolve the longest valid chain will be given more authority.

// returns True if the chain was replaced and replace the old one with new one by comparing the length of current chains with other chains of the system using if…else loop

Software Development Practices

- Requirements:
    - Python3 (with pip) installed with the following libraries:
        - flask
        - requests
    - HTTP client like postman
- Implementation: Explained previously in the report
- Testing:
    - API testing: ensuring that the interaction between the blocks in the blockchain ecosystem is as expected
    - Security testing: ensure that the application is vulnerable to attacks and Systems can protect the data and is capable of handling malicious attacks, etc.
    - Integration testing: ensuring that all the components of the application are integrated properly and performing the actions appropriately

References

- https://www.section.io/engineering-education/blockchain-consensus-protocols/
- https://hackernoon.com/learn-blockchains-by-building-one-117428612f46
- https://www.cryptopolitan.com/peer-to-peer-in-blockchain-how-it-works/
- https://blog.logrocket.com/complete-guide-blockchain-testing/
- https://www.guru99.com/blockchain-testing.html