

Secure Blockchain-based Categorical Information Transaction
and Depository Framework with Identification Enabled
Privacy and Decentralized System

by

Shahinul Hoque Ony

16101200

Suraiya Rahman

16101001

Takrim Rahman Albi

16101106

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
December 2019

© 2019. BRAC University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted or submitted for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Shahinul Hoque Ony
16101200

Suraiya Rahman
16101001

Takrim Rahman Albi
16101106

Approval

The thesis titled “Secure Blockchain-based Categorical Information Transaction and Depository Framework with Identification Enabled Privacy and Decentralized System” submitted by

1. Shahinul Hoque Ony (16101200)
2. Suraiya Rahman (16101001)
3. Takrim Rahman Albi (16101106)

Of Fall, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on December 26, 2019.

Examining Committee:

Supervisor:

Mahbub Alam Majumdar, PhD
Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

Program Coordinator:

Md. Golam Rabiul Alam, PhD
Associate Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:

Mahbub Alam Majumdar, PhD
Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

Ethics Statement

Hereby, We, *Takrim Rahman Albi* , *Shahinul Hoque Ony* and *Suraiya Rahman* consciously assure that for the thesis “Secure Blockchain-based Categorical Information Transaction and Depository Framework with Identification Enabled Privacy and Decentralized System” the following is fulfilled:

1. This paper is the authors’ own original work, which has not been previously published elsewhere.
2. The paper is not currently being considered for publication elsewhere.
3. The paper reflects the authors’ own research and analysis in a truthful and complete manner.
4. The paper properly credits the meaningful contributions of co-authors and co-researchers.
5. The results are appropriately placed in the context of prior and existing research.
6. All sources used are properly disclosed (correct citation). Literally copying of text must be indicated as such by using quotation marks and giving proper reference.
7. All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

The violation of the Ethical Statement rules may result in severe consequences.

We agree with the above statements and declare that this submission follows the policies of BRAC UNIVERSITY as outlined in the Guide for Authors and in the Ethical Statement.

Corresponding Authors’ Full Name & Signature:

Shahinul Hoque Ony
16101200

Suraiya Rahman
16101001

Takrim Rahman Albi
16101106

Abstract

The enormous scale activities of verifying information and data as an advanced resource is being increasingly more looked for after on each progressive year. A large portion of the well famous crypto-stages now-a-days are conceived dependent on the reason of the idea of blockchain. These stages fill in as an elective medium of transaction utilizing cryptography to verify the exchanges on an appropriated record. When utilizing customary, blockchain-based advanced every single other stage more often than not the regarded crypto-stages debilitates diverse all out data exchanges from being executed. Since frequently, the taking care of exchanges of various kind of data may bring about finding the security defects and over the top computational assets which may get disappointment in long haul. Accordingly, utilizing numerous kind of data exchange utilizing advanced crypto-stages stays in clouded side which have never been uncovered for making life progressively productive. Truth be told, empowering straight out data model may give the blockchain another floor in the realm of crypto-value-based infotainment. To floor the light on such a framework, our postulation model was exuded from the thought of less costly mining, which may be unraveled by means of planning of decentralized server miners as well as to grow such a crypto-stage condition which is multi categorical exchange and huge scale depository framework for safely hold and execute data. Consequently, our thought of categorical data transaction that we got from the hypothesis of making the UTXO all the more computationally proficient with the end goal that it can control various kind of data just as helping the floor of multi-enlightening storehouse and distributive processing where similarly a low controlled server can re-appropriate complex calculations while approving exchanges using the idea of blockchain for information recording based on nation identification.

Keywords: Blockchain; Distributed system; bitcoin; Internet of things; multi-categorical; Information; Depository; Storage

Dedication

Firstly, this thesis is dedicated to our parents, who sacrificed all of their lives towards our meaningful taught and made us able to do such work.

It is also dedicated to our beloved supervisor, who taught us that even the largest task can be accomplished if it is done one step at a time and with the effort of group work.

Last but not the least, we dedicate this work towards BRAC UNIVERSITY, as without this platform we could have never got such supervision and help.

Acknowledgement

First and foremost, praises and thanks to the Almighty Allah, for His gift of knowledge and good health that he bestowed upon us throughout our research. Our greatest and most sincere gratitude goes towards our research supervisor, Prof. Mahabubul Alam Majumdar, PhD for giving us the opportunity and immensely providing us with his invaluable guidance throughout the research. In addition to being an outstanding teacher his vision, integrity, resilience and sincerity have deeply inspired all of us. He has always challenged us to think outside the box and push ourselves to our absolute limit. Without his counselling, our thesis would not have turned out as we had hoped it would be. It was an outright privilege and honor to work under his guidance.

Our special thanks goes to Miss Syeda Ramisa Fariha for her genuine support and the keen interest shown to complete this thesis successfully.

We are thankful to BRAC UNIVERSITY for always providing us with the necessary facilities, equipment and resources during our research period. We would also like to extend our deepest thanks to all the faculty members of CSE department. Without their precious lessons they taught us throughout our entire undergraduate period, we would not have been here today.

Last but certainly not the least, we are beyond grateful and indebted to our beloved parents for their love, compassion and all the sacrifices they went through to give us a fighting opportunity in life and preparing us for the future. We are exceedingly thankful to all our peers and fellow classmates for their love, support and getting us through the all tears and hardship.

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iii
Abstract	iv
Dedication	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	x
List of Tables	xi
Nomenclature	xii
1 Introduction	1
1.1 Background	1
1.2 Research Problem	2
1.3 Research Objective	3
1.3.1 Public Blockchain	4
1.3.2 Private Blockchain	4
1.4 Scopes and Limitations	5
2 Literature Review	6
2.1 Blockchain security	6
2.2 Current Blockchain Systems	7
2.3 Distributed System	9
2.4 Smart Contract	9
2.5 Hyperledger	10
2.6 State Transition System	10
2.7 Quantum Computers and Blockchain	11
2.8 Usage of Blockchain	11
2.9 Blockchain storage issue	11

3	Methodology	13
3.1	Blocks	13
3.1.1	The Geneses Block	13
3.1.2	Data Blocks	14
3.2	The Consensus	20
3.2.1	Proof-Of-Work	20
3.2.2	Byzantine Fault Tolerance Algorithm	22
3.3	Shared Ledger	22
3.4	Decentralized System	24
3.5	Cryptography	26
3.5.1	Public Key	26
3.5.2	Private Key	26
4	Implementation	28
4.1	Flask web Framework	29
4.2	Blockchain functions	30
4.2.1	Create Block	30
4.2.2	Return previous block	31
4.2.3	Proof of Work	32
4.2.4	Check chain validity	33
4.2.5	Hash function	34
4.2.6	Add Transaction	34
4.2.7	Get Node	35
4.2.8	Add Node	35
4.2.9	Replace chain	35
4.3	Interacting with API	37
4.4	Workflow of Blockchain	38
4.4.1	Designing Blockchain	39
4.4.2	Chain initialization	39
4.4.3	Connect node	39
4.4.4	Add transactions	39
4.4.5	Mine block	39
4.4.6	Check validity	40
4.4.7	Check longest chain	40
4.4.8	Clear data	40
4.4.9	Data format	40
4.4.10	Usecase Diagram	41
4.4.11	GSM based Device Registration	41
4.4.12	Smart Device Registration	41
4.4.13	Vehicle Registration	42
4.4.14	Land Registration	42
4.4.15	Transaction Data	42
5	Result and Discussion	43
5.1	Data-set	43
5.2	Performance	46
5.3	Distributed System test	48
5.4	Block mining	49
5.5	New Block generation	49

5.6	Longest chain replication	51
5.7	Transactions	53
5.8	Discussion	54
6	Conclusion and Future Work	55
6.1	Conclusion	55
6.2	Future Work	56
	Bibliography	62

List of Figures

3.1	An Architecture of Genesis Block	14
3.2	An Architecture of All Data Block	14
3.3	Connection of Each block Through a Chain	15
3.4	SHA256 Hash Algorithm	16
3.5	Embedding Each Information with its associated Flag	17
3.6	Encrypting Financial Information To a Hash	18
3.7	Encrypting Device Information To a Hash	18
3.8	Extracting Categorical Data Based on Flags	19
3.9	How Nonce Controls the Mining Problems	19
3.10	The Whole Blockchain at a glance	20
3.11	How Proof Of Work Secures The System	21
3.12	Restricting Dual Transaction	22
3.13	Control Flow by the Smart Contract	23
3.14	Decentralization of the System	24
3.15	Control Flow by the Smart Contract	25
3.16	Secured Blockchains Transaction Protocol Using Public and Private key	27
4.1	Flask basic implementation.	29
4.2	Blockchain Flowchart	38
4.3	Blockchain usecase diagram	41
5.1	Time vs Block generation.	44
5.2	Different states of the System	47
5.3	Network address given to chain.	48
5.4	Parsed network addresses from other chains.	48
5.5	Mining the Block	49
5.6	Operation of a chain in the Blockchain network.	50
5.7	All Other Data Blocks	51
5.8	Chain information in Node 127.0.0.1:5002	52
5.9	Chain information in Node 127.0.0.1:5001	53

List of Tables

4.1	API information table	30
5.1	Blockchain network chain address table.	43
5.2	Transaction in a single chain of Blockchain network.	45
5.3	Proof of work complexity based on leading zeros of Target Hash.	45
5.4	Proof of work complexity based on Order of equation.	46
5.5	State change table	47

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

API application programming interface

BIoT Building Internet of Things

CPU Central Processing Unit

DApp Decentralized Application

DDoS Distributed Denial of Service

DoS Denial of Service

GSM Global System for Mobile

IBM International Business Machines

IMEI International Mobile Equipment Identity

IoT Internet of Things

JSON JavaScript Object Notation

M2M Machine to Machine

MSB Most Significant Bit

NID National Identification

P2P Peer to Peer

PoW Proof of Work

SHA256 Secured hashing Algorithm 256

SID Security Identifier

SSN Social Security Number

TPS Transaction per second

UTXO Unspent transaction output

Chapter 1

Introduction

1.1 Background

The start of crypto-transaction can be said to be one of the most developing and certain innovations of 21st century. Blockchain is essentially a web based computerized resource or a mode of exchange which utilizes cryptographic capacities to guarantee exchanges between parties just as enormous verified stockpiling [1]. In the mid-1990s, the "Cypherpunks" who laid the principal blocks to the establishment of the formation of cryptography imagined that the legislature and related administering associations have a lot of reconnaissance and authority over individuals' lives and their data [2]. As unquestionably not being the admirer of incorporated organization, these cypherpunks needed to utilize cryptography to enable the individuals to have more power over their cash and their own exchange data. It would have been a lot later in 2009, when the first historically speaking decentralized computerized cryptographic money named bitcoin was presented just because by Satoshi Nakamoto which was motivated by the early works of the cypherpunks [3] and in this manner another type of decentralized digitalized type of exchange was introduced. Crafted by Satoshi Nakamoto made an upset in the field of cryptographic exchange named blockchain [4]. Satoshi Nakamoto is said to be the innovator of blockchain who established the framework of computerized non-human position based exchange framework.

So as we would all be able to expect one of the most appealing part of blockchain is that it isn't constrained by any sort of overseeing authority at all [5]. It doesn't have a focal server or a point of control. Subsequently, it very well may be hypothetically considered as safe to any sort of focal obstruction and ward. This decentralization factor enables the whole system to be circulated among a huge number of servers and computers. Another of the other significant abstraction of blockchain is it's P2P (distributed) exchange framework [6]. The exchange consistently happens between just included gatherings. There are no confided in outsider, for example, a bank or budgetary organization to certify the exchange. Besides, the entirety of the computerized crypto-stages are known for their pseudonymosity. Pseudonymosity essentially implies that clients don't need to share their own data so as to claim a computerized resource, albeit the entirety of the exchange made by the client are made open to every one of the individuals in the system [7]. Pretty much every crypto-exchange in the market today runs on the blockchain arrange. The

blockchain is an ethical computerized record of exchanges that can be modified to record monetary exchanges as well as for all intents and purposes everything of worth. A blockchain is a period stepped chain of blocks [8]. Every block hold a progression of information as indicated by its utilization and data about exchanges. Aside from the engraved information a block likewise contains its own one of a kind identifier called the hash [9] and furthermore conveys the hash of its past square. So if a noxious client attempts to mess with a block in the blockchain, in the process he will change the identifier hash of the influenced blocks and subsequently, the square ahead can recognize the nearness of pernicious users [10]. Notwithstanding that, as blockchain is an appropriated record which implies every one of the reports about exchanges are repeated a few times with each update and are sent to each digital users in the system. Every client associated with the system has a duplicate of the whole chain [11]. So in the event that somebody needs to mess with any block the entire system won't arrive at accord and as a yield it will uncover that a programmer is attempting to get in.

One of the most critical key players of the blockchain are the miners. Each time an exchange is made between clients, those exchanges are counted and checked by the crypto-excavators. Their duty is to confirm if each data is all together. This procedure includes illuminating some entangled problems including cryptographic hash capacities related with every exchange square [7]. The hash esteem is fundamentally a fixed length numeric worth. The excavator utilizes different calculation, if there should be an occurrence of bitcoin it is SHA256 (Secured hashing Algorithm 256) to focus in on a hash esteem which is not exactly the objective [12]. After validation, the excavator at that point adds the block to the blockchain which contains a proof-of-work POW and updates the record. Thus the chain gets updated by adding the block and run simultaneously.

1.2 Research Problem

After the successful foundation of Bitcoin in 2009, the conduits were opened introducing another influx of new cryptographic exchange. Before long individuals of various degrees began to understand the significance of utilizing blockchain based exchange framework and extended more up to date methods for investigating the framework. Accordingly, individuals began enabling the stage in tremendous fields. One of the most reproducing restrictions of the present blockchain innovation is that it neglects to deal with numerous classification of information security concern. In addition, while the trading of numerous kind of data accessible in the blockchain their safe design should be created and upgraded to satisfy the framework prerequisites. According to the requirements, growing such a thorough structure to deal with the vigor of numerous data turns into the entrancing exemption as far as making life productive, versatile and defensive simultaneously. Furthermore, for this situation protection turns into an extraordinary risk as the more individuals will utilize blockchain the more information is being executed and the more it makes danger to security as aggressors may attempt to do fake exercises because of the impediments found in the framework. Defeating with such a productive framework where cost will be less and decentralization becomes helpful is an incredible test as far as both redistribution and rebuilding servers. In this manner, a verifies just as versatile and

security situated framework demonstrating turns into the most extreme test is such conditions.

On top of that, another pickle that has been approaching over the idea of blockchain based crypto-transaction is the term scalability [13]. Scalability alludes to the quantity of exchanges a system can withstand per unit time. For the most part, scalability is measured by TPS (Transaction per second). A system's Transactions Per Second appraising is the quantity of exchanges it can run every second and convey one-second reaction time to 95 % of the solicitations [14]. Starting at 2019, Bitcoin has the capacity of preparing 7 TPS [15]. At this very moment, alongside the expanding number of clients the bitcoin arrange is by and large increasingly clogged and with the development of every client, the TPS is decaying. Thus, exchanges set aside a long measure of effort to process and exchange charges are accumulating also [16]. In the event that digital money is ever to turn into a suitable substitute to contemporary installment frameworks, an installment entryway with a higher value-based throughput is an absolute necessity.

For the sole reason for empowering multi-class data exchange framework as far as blockchain, we thought of utilizing python as our essential language for executing the framework and the information that will be put away will be on JSON format that is key, value pair. Nearby python we will use flask [17] as our web application framework. It will be fundamentally used to actualize the API calls for sending and receiving requests from the system as per needs. Moreover, flask empowers, strong web platform generation alongside cross platform. For our web server we will be mainly using php and JavaScript. We will utilize postman to see whether the genuine requests are being properly executed that are by and large appropriately GET and POST or not. Lastly, we will build up the entire framework from scratch, as we need to plan the entire design of the model and need to convey it as the entire framework. Therefore, keeping the principles and guidelines of the blockchain structure we will attempt to refurbish the impediment that are depicted previously.

1.3 Research Objective

1. To create a decentralized advanced crypto-stage which will legitimize multi-class data exchange .
2. Constructing a stage which won't just be actualized on a basically excavator less condition however at the same time will likewise have incomplete mining capacities at client's prudence.
3. The arrangement and selection of the excavators will be resolved dependent on Scheduling calculations and capacities of every server.
4. The proposed crypto-stage will have extensive raised TPS in contrast with accessible cryptographic forms of money.

As a matter of first importance, for guaranteeing the essential reason of our target the proposed crypto-platform model will work on a blockchain empowered framework.

Blockchain is one of the most profoundly progressive and discussed decentralized disseminated system that has surfaced in recent years. Blockchain joins the use of an inventive as well as innovation on terms of decentralized record performing crypto-exchange. Blockchain is essentially a chain of blocks, in other words Linked List Data Structure, which can be traversed through one way [18]. We can see many existing blockchain based innovation a large portion of which are basically crypto-currencies. The most well-known platform which established the framework of blockchain was bitcoin. Different digital forms of currency which depend on blockchain comprises of exchange data bunched into a consecutive chain of blocks. In any case, blockchain has been additionally produced for multipurpose reasons, for example, ease in data management as well as transaction in different forms, Supply chain management, verifying data over systems, etc [19]. Furthermore, where there is the need of secure administration just as exchange framework which may include an outsider may join their framework utilizing blockchain for included security and protection just as for simplicity of safe of information [20]. Other than blockchain has additionally raised their arms through various kind of systems in agreement to their needs and use. Those can be delegated as :

1.3.1 Public Blockchain

Blockchains that are being utilized for open profited purposes, for example, for making it open for individuals of all stages are known as open blockchain [21]. For instance, bitcoin or Ethereum can be considered as the greatest case of open blockchain [22]. They are open for individuals all things considered and all stages and is non authority based which is no individual has a power over it rather individuals with a larger part number of readiness can make certain principles and guidelines of the entire framework [23]. The development of Public blockchain are never constrained by the clients on the grounds that as much as the make exchange the more the framework extends and develops exponentially.

1.3.2 Private Blockchain

Some companies or business institutions can make a framework as per their needs for the ease of their management and legislature using the blockchain which are known as private blockchain. Most of the private blockchains are confined over the boundary of the institution that are using such a blockchain [24]. These systems are never made public for general mass and people outside the institution are ignorant about what is happening and the sole purpose of the system. Only the incorporation that are using the system are likely to know the pros and cones and the bindings of the system. In the past few years the usage of private network-based system was not much popular amongst people as the massive usage of the system was not realized by people at scale. But as the technology started to emerge, people started to become more interested in the private network base blockchain and now-a-days private blockchains are mostly used.

1.4 Scopes and Limitations

To appoint the extent of the proposed framework, the preeminent undertaking of our postulation model is to build a multi-classification data exchange system. The development of such a crypto-stage with an exceptionally ease requires a planned mining condition. The explanation for making the proposed crypto-stage planned was to debase exchange charge. Looking for such a minimal effort model, our design was upgraded which will be depicted later on as we continue with the improvement. The arrangement that our proposed model urgently required came as a recently organized blockchain exclusively named as connected rundown yet the design of the blocks [25]. On the other hand referred to as blocks will change as we are going to include another information structure within the past structure prevalently utilized [26]. As we will assemble the framework on JSON format, we will fuse another field supplanting the past field. This enabled the exploration to break free from the current blockchain based crypto-stages. Also, we will attempt to make the framework less exorbitant making booking for CPU processors.

To show the impediments on the qualities of structure or philosophy that affected or impacted the translation of our postulation look into, the prime applicant is simply the plan engineering. As we will assemble a fundamental model we won't have the option to cover an enormous measure of clear-cut data accessible for exchange. As we increment the quantity of classifications in the framework the enhancement of exchange should be increasingly engaged. Also, as we will make the framework without any preparation, there may should be the procedure of programming designing which is past the degrees of our proposal look into. What's more, making the framework with an absolute new design makes it less dependable to individuals at various scales as we will develop an open blockchain [27]. Our framework is just over the utilization of open systems. Thus, this framework isn't adaptable for private systems. Proceeding with the way that, our proposed model might result in future worthiness as we will utilize the Nation Identification / Social Identification / Social Security Number [28]. Moreover, our proposed model will work for frameworks kept have a semi-brought together framework. As we will utilize essentially brilliant gadgets and budgetary data, these things should be included the principal endeavor by a brought together power yet again after it will begin filling in as a decentralized framework. Last however not the least, The web structure needs to have an additional security worry as we will be predominantly utilizing open systems there may be instance of different assaults done on the servers, subsequently these servers needs legitimate security and the board [29]. Accordingly, when it went to the usage of the proposed calculation as opposed to executing the calculation on recently settled system, our group needed to probe a summed up blockchain organize which was actualized with the assistance of python. Therefore, it must be summed up which we will attempt to actualize in our future work. Be that as it may, for this proposal we concentrated essentially on the multi absolute exchange alongside less cost mining.

Chapter 2

Literature Review

Satoshi Nakamoto, who is considered as the creator of Blockchain first gave the concept of Blockchain in a white paper published in 2008, the world started to realize how Blockchain could revolutionize our society [4]. In this era, Blockchain is considered to be one of the most revolutionary technology. It has advantages that could be use in more than one part of our society. It could benefit the financial sectors, data storage systems, registry systems, online e-commerce sites, banks and many more institutions in our society.

2.1 Blockchain security

According to Don and Alex Tapscott ‘The Blockchain is an incorruptible digital ledger of economic transactions that can be programmed to record not just financial transactions but virtually everything’ [30]. The main purpose of using Blockchain is to store validation data so that in future, the data can be used for validation of information. Satoshi Nakamoto introduced the idea of Blockchain, a distributed system through his model of electronic cash called Bitcoin [31]. The writer of the paper introduced the concept of digital currency which would be free from any kind of regulations by a central authority or government. Thus creating cryptocurrency for the first time. According to S. Underwood ‘Data stored on Blockchain is secure, trusted, auditable and immutable’ [32]. Firstly, immutable means unchanging over time or unable to be changed. Blockchain stored data is immutable means that data stored on blocks in the Blockchain cannot be changed after once it has been mined. If an entity tries to change the data, the following blocks including the modified block will become invalid and thus the chain will replace the data from the network by replicating from other chains. Thus data stored on Blockchain can be considered as immutable. Blockchain data is also auditable which means it can be verified. Data on Blockchain are publicly available on the network and can be traced back to the public identity of the sender or the recipient making Blockchain auditable. Blockchain also can provide information like if transactions really occurred, if the balance exist or what is the owner of data. Blockchain data is also trusted as the

data cannot be changed due to the reason of immutability, once a transaction has occurred it cannot be changed by anyone else if not the whole network is taken down. However, as Blockchain typically are made of millions of chains and the chains continuously validate and replicate the data, it is nearly impossible to chain information stored in Blockchain. All this features of Blockchain contribute to the fact that Blockchain is a secure way of keeping records and data. Another feature of Blockchain is that it is fully transparent. The data is publicly available and the transactions can be linked to individuals. For all this reasons Blockchain is considered among one of the most secure systems for storing information. This is why thousands of people currently are using Blockchain systems like bitcoin instead of government issued currencies.

2.2 Current Blockchain Systems

As Blockchain increases in popularity, newer and newer Blockchain based systems can be seen emerging. One of the most popular emerging systems based on Blockchain is IOTA. IOTA is a Blockchain based system designed to work like cryptocurrency for Internet of Things based systems [33]. The mainline Blockchain systems like bitcoin require powerful systems to accommodate large and fast computational needs. However, IoT devices are mostly basic computational devices with small processing power. They would not be able to compete with the powerful devices in the stream lined Blockchain systems. Therefore, the need for an IoT based Blockchain could be seen. IOTA tries to solve the issue by providing a cryptocurrency systems for internet of things. In IOTA the weight of the transaction is based on the amount of work done by a node in the network. The transactions are sorted by weights where higher weights get more priority and lower weights get less priority. Moreover, they assumed that no devices can generate a large number of transactions with acceptable weights in a short period of time due to avoiding spamming like attacks which would flood the system. Moreover, the paper discussed how it could resist a quantum computer based attack. A quantum computer is capable of faster computational power compared to normal binary computers as it can calculate large amount of numbers faster and thus could predict the correct nonce faster to always be able to solve the proof of work faster than anyone else in the network and thus could be used to introduce corrupted nodes to hack the Blockchain system.

The paper [34] discusses about a system where IoT devices use Blockchain to store generated data and where users have to pay a service charge in order to receive data from the system. In this way, valuable data generated from IoT devices can be utilized for research and other purposes. The paper illustrates a model which can store large amount of data which is accessible to all.

In this paper [35] the researchers demonstrated an attack on a BIoT network and analyzed the result. The researches discussed different components of a BIoT network and described the working procedures of the network. The paper also described about Ethereum transactions.

An important issue for Blockchain is to maintain privacy and data security. David Lopez Bilal Farooq worked on the issue of privacy for Blockchain [36]. They segregated the data stored on Blockchains into multiple segments and worked on the idea of continuous data generation from different nodes by several companies' production transport information, mobile devices registered by cellphone towers. They designed a smart contract that would give freedom to companies and institutions in the level of authority. In their test case they used Hyperledger Indy which is a public permissioned Blockchain for decentralized digital identities and the exchange of information in secure peer-to-peer connections. They provided a model which could solve the privacy and security issues presented by Blockchain system. Although current quantum computers are not capable of such computational power, analysts are predicting that within the next decade companies will be able to develop quantum computers with such computational power. This would provide a risk for not only Blockchain systems but also other security systems that use hashing and digital signatures using such functions.

In the current time period we can see the emergence of Blockchain based applications that utilize the security power of Blockchain. P. Zheng Z. Zheng and W. Chen discussed about Blockchain based application that use smart contract also known as decentralized application or DApp [37]. They have discussed how Blockchain based applications have shifted the idea of a central trusted authority to a distributed authority which provides a more trustworthy figure for a larger audience. The authors provides a visual prospective of cost evaluation in terms of data storing within a Blockchain. They also discussed about the issues of the model such as throughput of public Blockchains and information filtering and management. We are living in an era where our personal lives are filled with different smart devices like smartphones, home automation devices, smart watches, and home assistants. However, due to the fact that a large number of companies produce a very diverse set of devices, it is very hard to provide a standardized identification system that can be followed by everyone. J. Koo and Y. Kim's research on device identification system for IoT platform gives us an ideas in how to handling this kind of issues [38]. They discussed how IoT devices can use multiple standard of identification like M2M standard, GS1 standard, IBM standard. The researches proposed system which could translate various standards of Identification into a specific one such that smart devices which followed different standards could exchange information and operate in newer standards. According to author G. Eder, Blockchain based land registry systems are the future [39]. According to the writer, property rights is one of the fundamental rights in international development. Different countries have tried to use Blockchain to store land registry information. Among them Georgia and Sweden have started Blockchain based land registry in 2016 which has successfully contributed in reducing land ownership related disputes and corruption. The ownership can be easily traced to the original owner, therefore, is has increased transparency in terms of an individual's property estimation and taxation system too. The writer has discussed in perspective of Ghana how a Blockchain based land registry system could benefit a country like the Republic of Ghana in which 80 percent of the land ownership are undocumented [40].A Blockchain based land registry would be a step in the right way in terms of development for Ghana.

2.3 Distributed System

Blockchain is a distributed system which refers to that Blockchain system is not stored in a specific server or central computer. Rather is it stored in every device operating under the Blockchain network. Even if multiple devices get disconnected from the Blockchain network, the network is still able to maintain itself and conserve the data. This concept of a system which is not located in any particular servers is known as distributed systems. The benefit of distributed systems is that, this systems do not have a single point of failure. In a general server client configuration. If the server goes down, the whole system is interrupted. To solve this issue sometimes server client systems are configured in such a way that multiple servers need to stand by to provide services if one of the servers go down. However, this systems could prove to be extremely expensive and sometimes could also prove to be redundant as some servers would always remain unutilized. On the other hand, having a distributed system would result in a system which stay on without any down time. The system would be scalable. Could be scaled based on the needs of the system. There could be a debate of how a system of this kind can be managed or modified or monitored if needed. But that is a different topic of discussion.

2.4 Smart Contract

In our day to day life we follow certain rules and regulations. Even our digital systems need to follow certain rules and need certain verifications. Such kind of regulations are need of distributed systems too. Smart contract can be considered as a form of regulatory policies that distributed systems follow. Smart contracts very important in order to verify transactions and regulate how a systems should operate. In terms of Blockchain, smart contracts define the rules and penalties and agreements of the system. Smart contracts not only define certain aspects of the system, but also it automatically enforces those obligations while storing data or conducting transactions in the system. Simple distributed systems can operate without a smart contract. However, for the uninterrupted and fluent operation of large systems smart contracts are essential. Although smart contracts can be debated to be overheads for systems which need to be reduced. Till now there is no other concrete system which could full replace and fulfill all the usage of smart contracts. Much research is being conducted about how to reduce the complexity and execution expenses of smart contracts. Till then this current forms of smart contracts are necessary for proper operation.

2.5 Hyperledger

A very crucial part for Blockchain technology is Hyperledger. According to Hyperledger.org, Hyperledger is an open source collaborative effort created to advance cross industry Blockchain technologies which is a global collaboration hosted by Linux foundation. Hyperledger is not only used in Blockchain, but it is also used for internet of things, supply chain management, manufacturing and different other technologies. Due to the need of databases to store information, databases have become relational by storing all information in rows in tables. Through the advancement of internet, now the world is connected, therefore the need for a distributed database system pushed for further research. A solution would be shared database. However, shared database have many issues like trust factor, centralization of data, security, and risk of data loss. This is why currently Blockchain is considered as the most prominent alternative for shared databases and the solution for distributed data storing system. As more and more companies realized the need for Blockchain research, they started to contribute to the technology and in 2015 different firms decides to pool their resources and create an open-source Blockchain technology which would be easily useable by anyone. This is how Hyperledger began in 2015 under the ownership and guidance of Linux Foundation[41].

2.6 State Transition System

Another important part of Blockchain is its State Transition System. According to M. Nygaard and E. Schmidt, transition system can be described as a dynamic processing system which transits based on timestamp [42]. State transition system represents how a system transits from one state to another by the passing of time. If we could consider all the variables related to a computer as a matrix where the rows would reflect the variables and the columns would reflect the timestamps, the matrix could fully represent in what state a system would be in a specific time and could successfully provide an accurate time line of what is happening inside a system including the inputs and outputs of the system at any particular time. In theoretical computer science study, state system is used to analyze and understand the behavior or characteristics of a system. In Blockchain, state system is used to verify transactions and make the network immutable. After each change of state of the Blockchain system, the information of the previous state is fully recorded in the network and it immutable. As the Blockchain system progresses in continuous states, the previous states are being stored as part of the network, the publicly available information can be accessed by anyone to verify, study or analyze any state of the Blockchain system. The state transition system is very important for Blockchain as it helps to validate transactions. For example, in bitcoin, due to the state transition system, it is possible to figure out who owns what amount of coins and where did does coins came from and if coins were used for transactions where they have gone, and also answers how and when the transitions took place and what is the current state. This is what provides transparency to Blockchain systems.

2.7 Quantum Computers and Blockchain

Another aspect of Blockchain is that it is secure due to the ‘one-way’ mathematical functions. Conventional computers need many years to solve these problems. However, it is estimated that future quantum computers could very quickly solve these mathematical functions and break Blockchain’s secure system. According to the article written on Nature’s website [43], quantum computers could find prime factors of very large prime numbers very quickly compared to a binary computer which could take years to calculate. This could mean that hashes could be forged and thus, hackers could forge different identities and takeover their properties, balance and such stored data. This would make Blockchain systems obsolete as they will not be able to provide security anymore.

2.8 Usage of Blockchain

According to author S. Underwood, Blockchain has many features which could benefit financial sectors [32]. Blockchain started as a financial system with the introduction of Bitcoin which is a cryptocurrency. However, it is not just limited to cryptocurrency. In financial services, storing large amount of data in synchronous manner for many location has become a burden. Blockchain is a beam of light for financial sectors as according to the author, Blockchain is basically a giant time stamp. Everything on Blockchain is timestamped. Therefore, it is easier to keep large amount of transactions and data synchronized. However, there are problems in terms of privacy when financial institutions use Blockchain as data stored in Blockchain are publicly available and therefore the question arises how much information needs to be exposed for proper operation.

2.9 Blockchain storage issue

A key factor of Blockchain is how much storage space it needs and what is the amount of transactions it can accommodate. According to authors of Blockchain Challenges and Opportunities: a survey [44], the most popular application of Blockchain, Bitcoin faces multiple challenges, like scalability, block size, amount of transaction per time unit, amount of blocks generated per time unit. According to the authors if a greater storage space is used by blocks, it will result in centralization of the Blockchain as users do not like to store a Blockchain which would consume a large amount of storage space and would also contribute to slower propagation in the network. Therefore, chains would be moved to servers which would make Blockchain a centralized system. Authors have also focused on the issue of miners using selfish mining strategies to profit from Blockchains [45]. The writers also discussed about the issue of choosing the wrong proof of work (PoW) could waste more elec-

tric energy for users and rendering mining revenue to become a loss for the miners. Thus reducing the number of miners and making the Blockchain vulnerable to brute force attacks. Therefore, making Blockchain storage capabilities as low as possible a very high priority to make a Blockchain network a success. R. Twesige in his understanding wrote about the features of Blockchain technology. According to the author, Blockchain is fast in terms of communication, it is cheap in terms of payment and exchange of value, it is easily accessible and publicly open, it is an open source system which can be programmed for improvement if needed, the data stored in the Blockchain network is easily accessible to anyone making it transparent, and finally a distributed system which is free from any kind of central authority [41]. Blockchain is a technology that can contribute to multiple factors of our life like finance, funding, trade, voting system, monetary systems.

As Blockchain gets more and more users and the networks gets flooded with transactions, the block chain system faces the issue of space snowballing effect problem due to ongoing transactions conglomeration. This means that as the number of transactions increases, the amount of data needed to be stored in the system rapidly increases and thus it increases the size of the system in massive amounts after each day. C. Ehmke F. Wessling and C. Friedrich discussed a Blockchain model which would be light weight and faster compared to the standard form of Blockchain used in current times [46]. They discussed how currently established newer forms of Blockchain based system like IOTA [47] and Ethereum [48] try to handle the issue. They discussed about the issues faced by decentralized system known as Fork. Their ideology was to create a block chain where older blocks were not needed to verify the newer blocks, thus reducing the overall proportion of data. The model was shown as a tree which created new branches with each transaction to validate the transactions and add to the main chain. They proposed a system where the transactions would be done outside of the system to a sub system. When the system shifts the transaction to the sub system it would verify needed information like current state of sender and receiver and the Blockchain. The subsystem would conduct the computation needed for the transaction and would verify the transaction. Then the sub system would return the validated transaction to the main system by providing validation information and other materials which could be easily verified by the main system. This would reduce the computational power needed by the network as some part of the work would be outsourced. The authors discussed about what other approach could be taken to create the model.

Chapter 3

Methodology

The primary purpose of our model is to enable the system for conducting such a way to ensure multiple types of data transaction through large scale privacy and to facilitate the users to use the system through trustworthiness. To erect such a system we need to yield our own architecture of the blockchain upholding the four building blocks of the Blockchain [49]. Thus we focused on the five elementary components of the blockchain:

3.1 Blocks

Every blockchain are constructed using blocks where blocks work as a data holder [50]. Thus the connection of blocks through chains are known as blockchain. These blocks are the main virtual materials of constructing the blockchain. The whole structure of a blockchain are as such of a linked list where each block is connected with its previous hashes. In our system the blocks are in JSON format and the data types are as per needed which we have described below.

3.1.1 The Geneses Block

The very first block in our system is the genesis block. For the theory of blockchain we know every blockchain starts from the genesis block which refers to the first block of the blockchain [51]. This can be referred to as a dummy head of the blockchain from which the initial chain starts. Continuing the fact that, all other nodes or blocks are the child or chain of this block. In our architecture this block doesn't contain any data rather only contains the hash, a timestamp and a nonce which makes this the genesis block [52]. When our blockchain is initiated the genesis block is automatically created [53]. Rather than genesis block all other block refers to its previous blocks hash and finally connects to the genesis block which is the very first block of the blockchain [54]. Thus the general architecture of the Genesis Block is as follows:

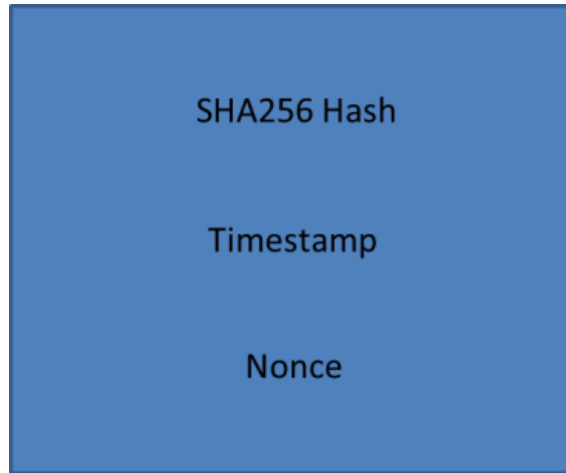


Figure 3.1: An Architecture of Genesis Block

3.1.2 Data Blocks

Other than the genesis block all the blocks in our blockchain are the main data depository units. All the transactions that occurs are stored in the data blocks and are added in the blockchain. Our blockchain are built utilizing blocks which behaves as an information holder. Subsequently the association of blocks through chains are known as blockchain. These blocks are the principle virtual materials of building the blockchain. The entire structure of our blockchain are in that capacity of a connected rundown where each block is associated with is past hashes. The primary component of storing the information in the blockchain is called the blocks or nodes [55]. All the transactional or non-transactional data are kept in these blocks and gets added to the blockchain [56]. In our framework, the blocks are added linearly, except for side chain or multiple threaded chain such that IOTA [57], through a chain to the previous hash like a linked list. In our architecture, building the blocks are the main challenge as different types of data will be stored in the data. The architecture of each of our blocks are as follows:

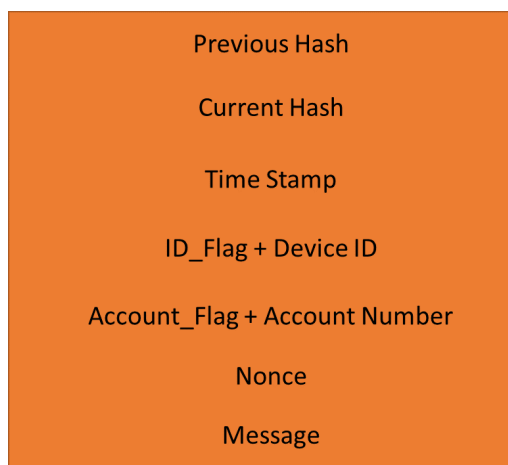


Figure 3.2: An Architecture of All Data Block

To know the actual architecture of the implemented system we need to know how the blocks are created first. Thus knowing about each block is much more important than anything else on blockchain. The components of each block are as follows:

Previous Hash

All the blocks in our blockchain are connected via the previous hash (SHA256) [58] which incorporates the integrity of the one-way retrieval of the blockchain. In this system the hash used was 64-bit SHA256 hash and is immutable that means a hash can never be overwritten [58]. In our system all the blocks refer to the previous block and thus connects to the genesis block. But one can never go to the genesis block by traversing through the blocks because the blocks are dynamically generated and changes continuously. Thus, Blockchain retains its integrity and is immutable.

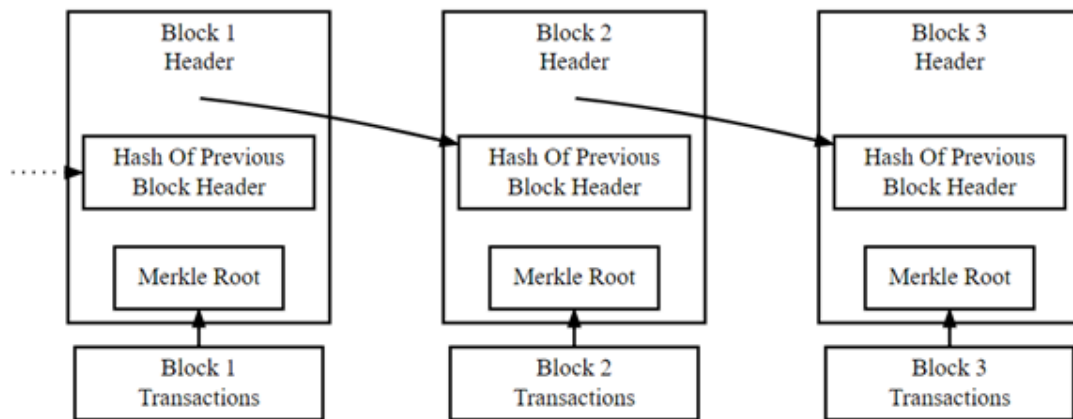


Figure 3.3: Connection of Each block Through a Chain

Current Hash

All the blocks in our blockchain has a hash which is used as the unique identity of the block [59]. These hashes are immutable and are the main building block of the blockchain as they represent as well as connects our whole blockchain. Each block refers to its previous blocks hash. Now the question arises how these hashes are generated in such a way that they are non-deterministic as well as unique. These hashes are generated based on the timestamp, nonce and the flags. As all of these values are unique so we generate only one hash from the 264 numbers with a probability of 1 out of 3.7×10^{11} that the hashes should match [60] .

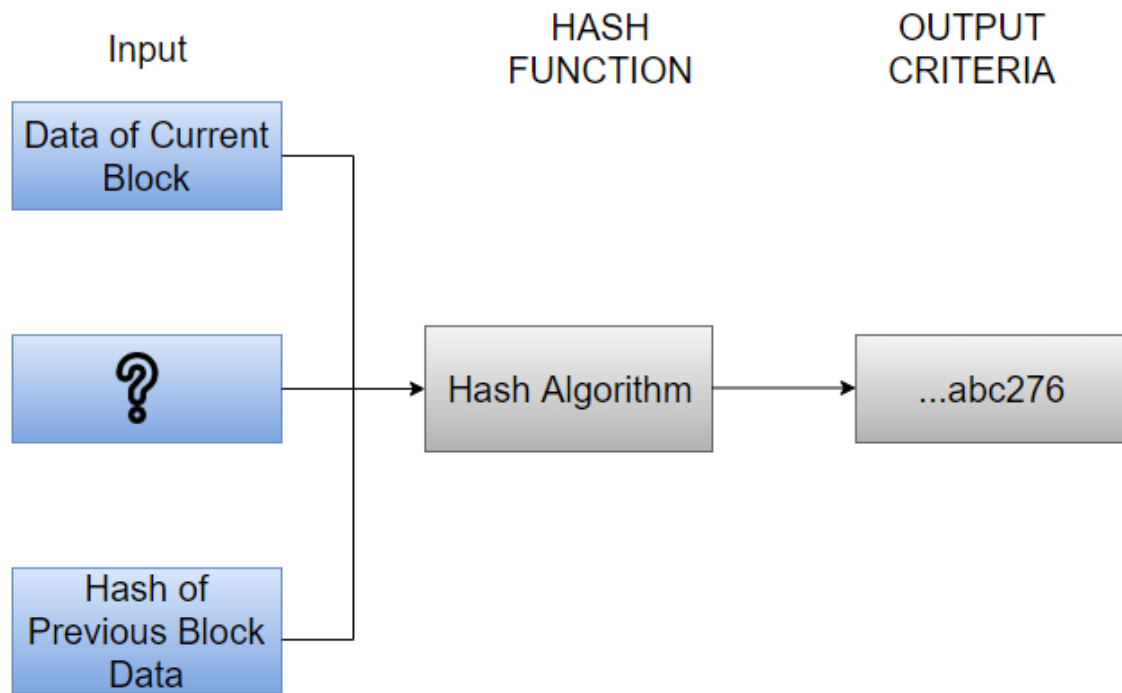


Figure 3.4: SHA256 Hash Algorithm

Timestamp

When the block is mined it saves the timestamp and shows when it was mined [61]. Thus every block in our system is unique and cannot be mutated further and gets added with the given timestamp on blockchain. This timestamp is also used for generating a unique hash that makes the blockchain more transparent [61]

ID_Flag and Account_Flag

As we are dealing various form of information to store and transact, every category needs to be separated uniquely so that we can control it within the blockchain. For this model we will be using mainly two types of information which are

1. Smart devices
2. Financial Information

First of all, choosing such two types of information is that they represent completely two different domain in. In addition, the purpose is to simplify or generalize the model in future. In this model we will be using a 1 directional array of length 1 which contains only the above information. Continuing with the fact that, these two types of dossier must have a symbolic notation so that we can signify them and can make them more understandable for the model implementation and future analysis. Thus we popularize two new types of variable, which are, ID_Flag and

Account_Flag. We will retain ID_Flag to be 1 if the transaction is a device and 0 for the Account flag which indicates financial information such as Bank Account information or registry information and so on. Thus if we can extend the value of flags to n, we can work with n category of information.



Figure 3.5: Embedding Each Information with its associated Flag

But as the category increases the usability increases exponentially and the performance a concern issue [62]. Because we have to handle n categories of data. Along with this, as categories increase we assume people will tend to add more information for further ease and security. Besides, adding more information will need more storage as well as more computation time to mine and compile.

Device ID and Account Number

As all the devices has a unique number which we represent as our Device ID and all the financial information has a unique transaction number which is the Account ID in our case, has to be represented in such a way that we can perform multiple arithmetic and modular operations over them and make them volatile for our model. Thus this information must be kept to each blocks being hashed so that we can uniquely identify each device and types of each device.

The financial information are encrypted as follows:

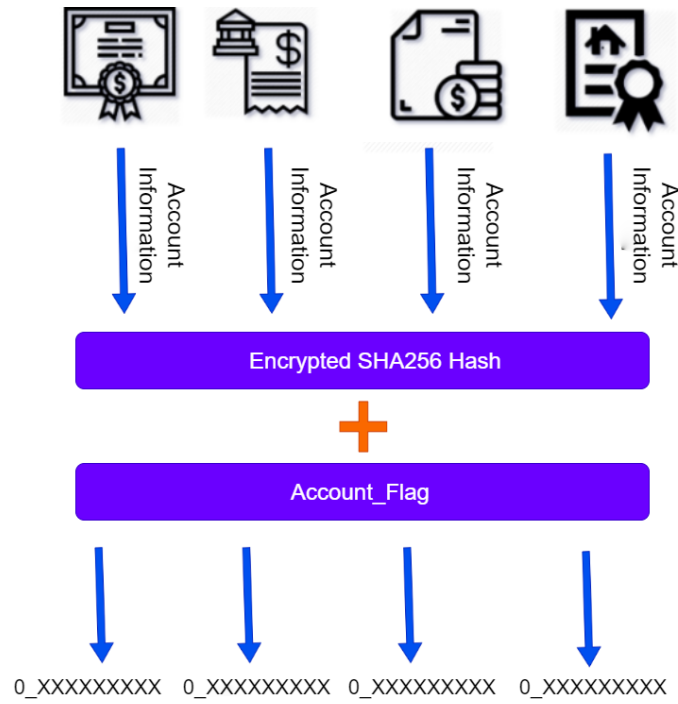


Figure 3.6: Encrypting Financial Information To a Hash

Along with financial information, the Device information are also needed to be embedded to a certain hash and thus added to flag. In our system the device hashes are also created thus:

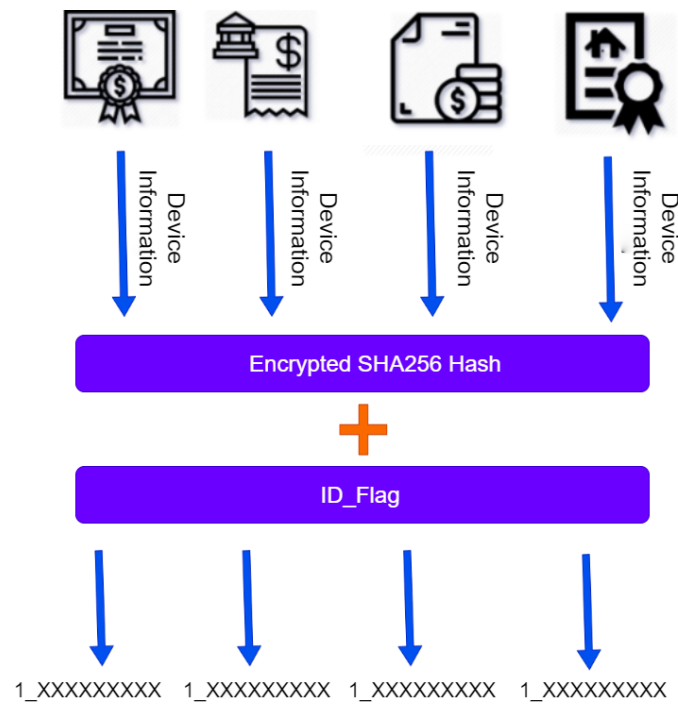


Figure 3.7: Encrypting Device Information To a Hash

In each of the block of the blockchain there might be one or several amount of transactions that could occur. Thus, from the above tagged Account Flag and ID Flag we can easily extract the needed transaction information and can group them for simplicity.

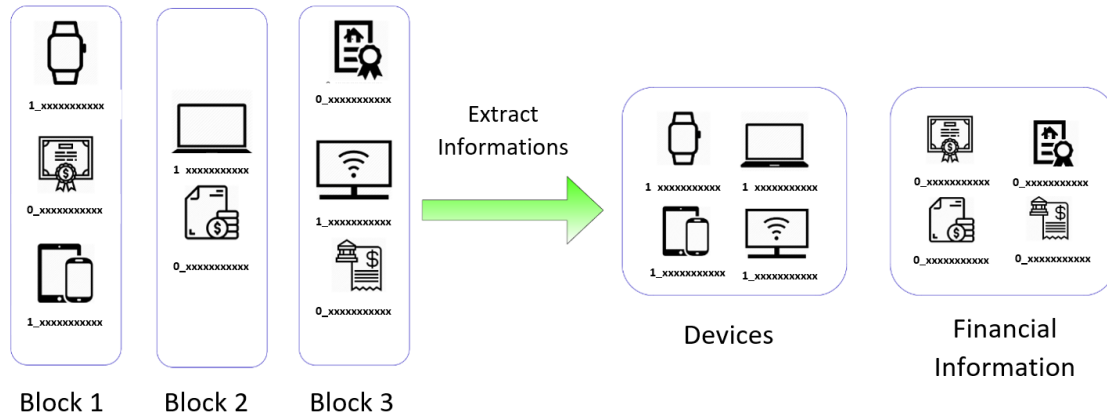


Figure 3.8: Extracting Categorical Data Based on Flags

Nonce

Nonce is a random number popularly used in cryptography to generate and control hashes. In our system Nonce are usually random numbers but they can be pseudo-random numbers as well. In our system, Nonce along with timestamp and other information can be used for more transparency in cryptographic communication. We used nonce largely on proof-of-work in blockchain to find the desired hash from the mempool[14]. Thus to find the desired hash upto a certain limit we hashes more faster and can make our model more dynamic.

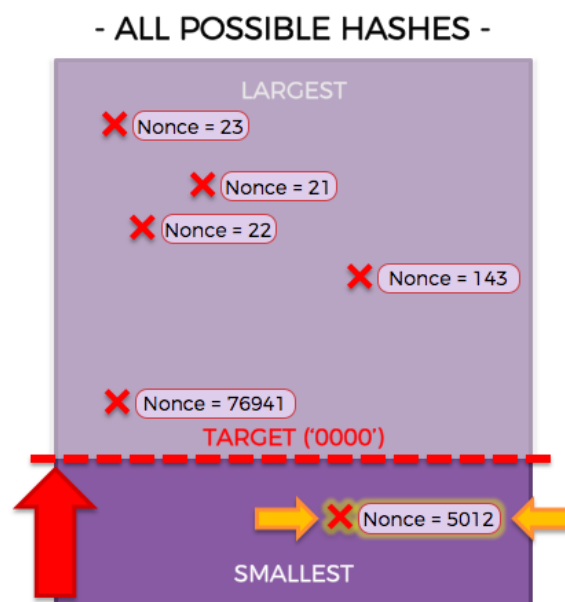


Figure 3.9: How Nonce Controls the Mining Problems

Message

In our system, the Message after each transaction is made shows the current state of the block such as whether the block was mined successfully or had issues or not. This makes the blockchain system more user friendly as well as helps the developers to find issues and flaws.

So finally all the blocks are ordered as follows :

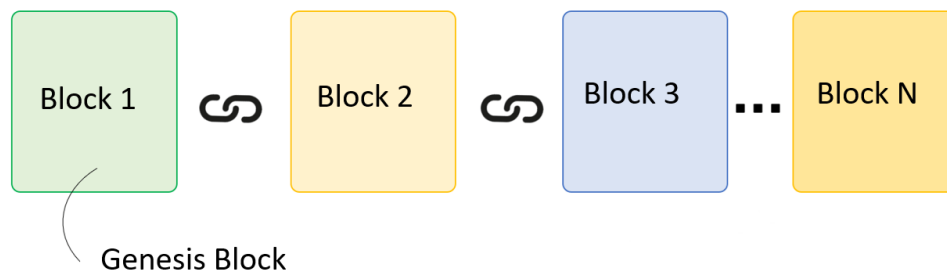


Figure 3.10: The Whole Blockchain at a glance

3.2 The Consensus

The consensus is mainly an agreement that can be used as a fault-tolerant procedure for a digitalized system [63]. One of the main problem of traditional transaction was the involvement of a third party that harness the time and money [64]. Blockchain overcame with this limitation but needed some kind of algorithm that can supervise the activity or transaction of information. This brought the idea of consensus. We developed a consensus protocol for our system and implemented it for the transparency and secured communication. In our system the main part of consensus is the Proof-of-Work algorithm that enables a digitalized as well as decentralized system to know whether the transaction is valid or not.

3.2.1 Proof-Of-Work

Proof-Of-Work is a protocol the determines mainly whether a transaction is valid or not. Moreover, it deals with cyber-attacks and keeps the translucence of the system by validating the whole system/model [65]. In our decentralized system there might be unwanted conditions like denial-of-service attack (DoS), or Distributed DoS attack which is (DDoS), which can exhaust the resources of the system with huge amount of unverified requests. Along with this in most of the cases in our system

it removes the problem of double spending [66]. The term double spending actually means whether an individual is trying for a fraudulent transaction is not. That is, for instance, a person has 10 unit of money (e.g. \$, £ and so on). In a very short period of time, if that person wants to make a dual transaction by fooling the system, he won't be able to do so. Because our proof of work won't allow him to do such a transaction which is not valid as the proof of work will look at all the blocks and will make sure he has made a valid transaction that is he has a proper amount to transact [67]. Thus PoW can keep the clarity of the system by validating the flow of the activities. The Process can be shown as follows :

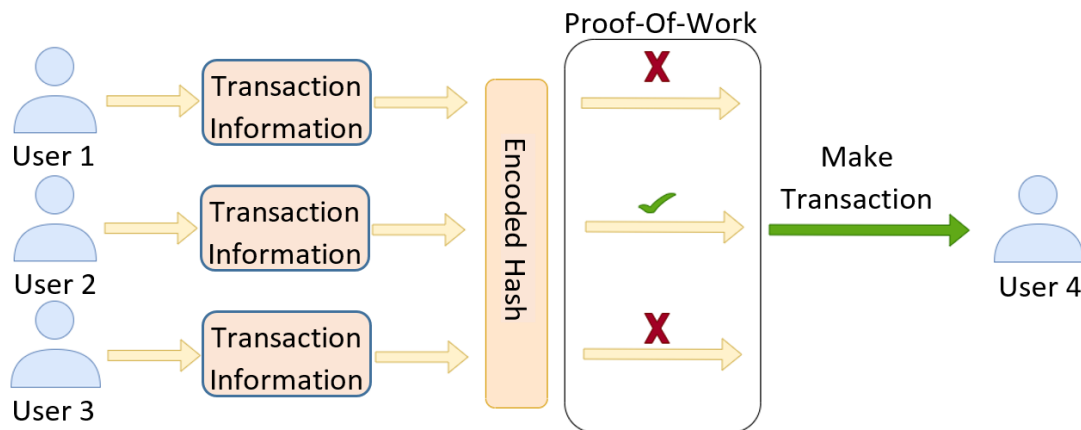


Figure 3.11: How Proof Of Work Secures The System

In fact, in the framework it controls all the transactions without obligations of any third party. Like, for a traditional system if a person wants to do a transaction of anything in almost every cases he has to go through a third party to hand over the property, money or information. In this case transaction refers to handing over something to anyone for good. But with the concept with the blockchain is revolutionary in this case. We can transact directly with larger security without the intervention of any third party [68]. But to make the system secured and free of lag, we have to ensure every transaction is valid and done by a valid user. The proof of work comes handy here. Proof-Of-Work validates the transaction by looking through all the blocks available in the blockchain and makes the calculation whether this is legitimate transaction or not. Cynthia Dwork and Moni Naor, first originated the idea of Proof-Of-Work back in 1993, but the term “proof of work” was coined by Markus Jakobsson and Ari Juels in a document published in 1999 [69]. Now the question arises how Proof-Of –Work works. In our Blockchain, the Proof-Of-Work works by all the blocks validating the transaction and whether the transaction actually exists or not.

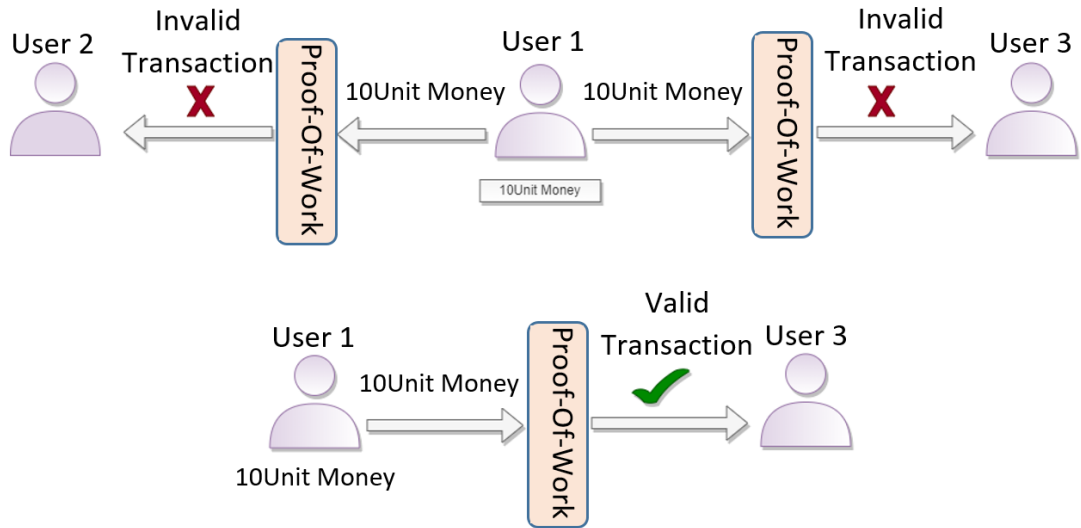


Figure 3.12: Restricting Dual Transaction

3.2.2 Byzantine Fault Tolerance Algorithm

It was essential to illuminate how the blocks inside a disseminated system concede to a choice when a portion of the nodes come up short or act deceptively. Byzantine General issue was clarified just with a gathering of commanders assaulting a city and need to have appropriate correspondence and understanding among them. In the event that solitary a piece of commanders assault the city, the assault would come up short [70]. As our blockchain is a dispersed system and every one of the hubs have a similar power without any focal position, Byzantine Fault Tolerant Algorithm guarantees consistency is kept up in record of various hubs.

3.3 Shared Ledger

One of the most decisive hallmark of blockchain is the distributed public ledger. That means a document which is available to all and anyone can view this but is immutable that is no one can edit [71]. In addition, shared ledger provides a covenant that every user as well as the miners must follow to retain the blockchain. Basically, the ledger is a document of rules that governs the blockchain where each blockchain must follow these set of regulations to sustain. In our system all other activities of blockchain must be controlled through the ledger. The ledger sets up under which conditions the transactions will happen or when each step will start and so on [72]. These creates more transparency to our blockchain as the system is following a flow of works which cannot be disrupted. One of the most important part of shared public ledger is Smart Contract.

Smart Contract

Talking about contracts is such that, it is an agreement that two parties follow to retain an exchange. In such cases there must be an involvement of a third party that has to verify the signature of both the parties [73]. On the other hand, a smart contract is a digital document that is digitally signed and accepted by both the parties based on some agreements while doing a transaction without the intervention of a third party. In other words, a Smart Contract is a public ledger where the regulations and agreements are written in an immutable digital form and the blockchain has to accept or follow the contract while doing a transaction. In our case there might be different servers where each server will provide the sophistication of using the blockchain. Under each server a client can host and join the blockchain. While adding a transaction, a client must maintain the rules of a smart contract. With the rapid growth of data as computations become expensive and time-consuming, we propose to host the smart contract on a server where multiple service providers can connect to the smart contract with the clients and the latency decreases.

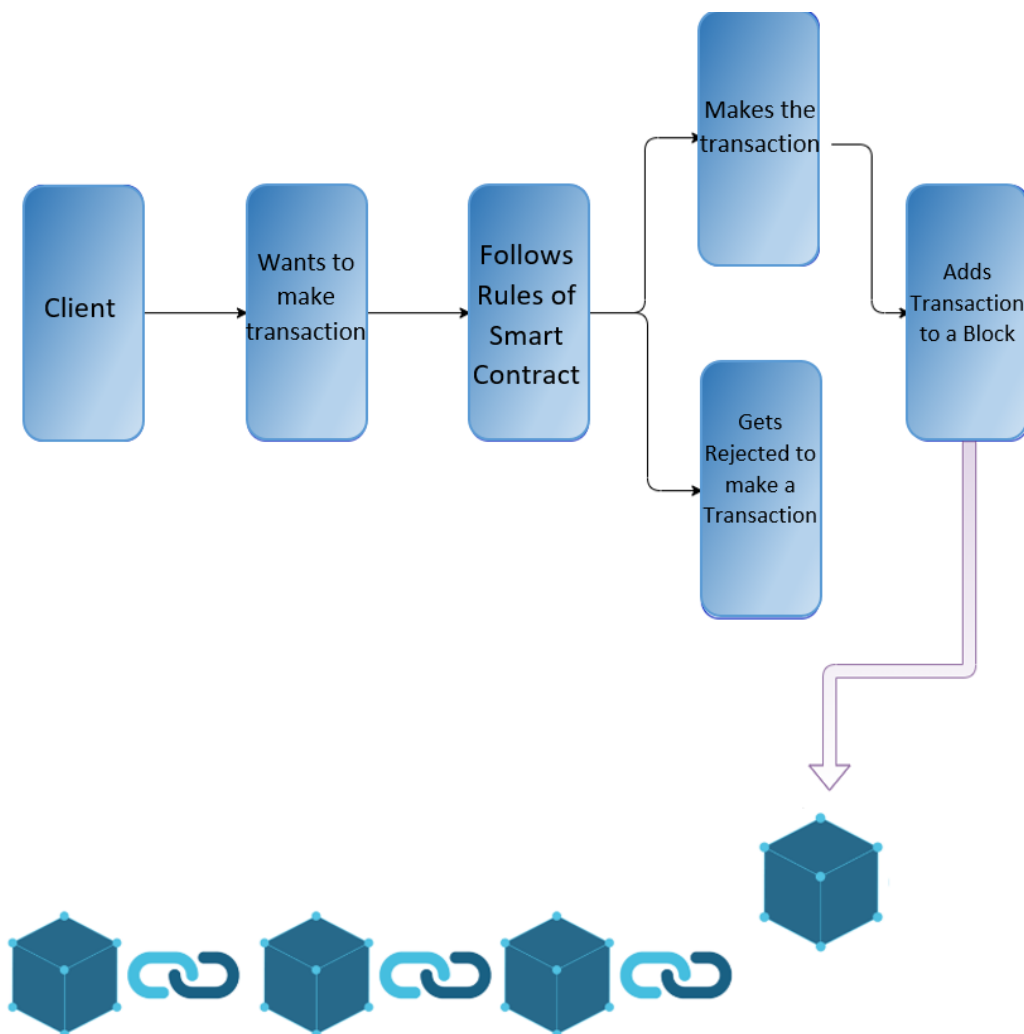


Figure 3.13: Control Flow by the Smart Contract

3.4 Decentralized System

The term decentralization means that not only one individual can control the whole system but many [74]. The main purpose of making a system decentralized is that no single authority can influence the whole system but the majority of the users have to have the same opinion. In addition, no single individual can interfere on transactions in the system, and any transaction request requires the consensus of most participants [2]. In our blockchain system, all the transactions that happen has to be recorded by all the users so that they can participate in the proof-of-work. If a transaction were to happen majority of the users must confirm that the transaction is a valid one. If an individual or small group of people tries to manipulate the system for their ease or unethical conducts they won't be able to do so, as these needs to have the legitimate confirmation of majority group of people. It is one of the most distinguishing and powerful fact of our blockchain that, rather than traditional database system it stores the data in each of the blocks and each blocks are stored on a central chain which is immutable [75]. There might be a central authority who might have the right to control the locations partially but cannot modify the network or the chain without the consensus of vast majority. As the system becomes decentralized, the transparency of the system increases. Thus, with the help of Proof Of Work the role of miners comes handy.

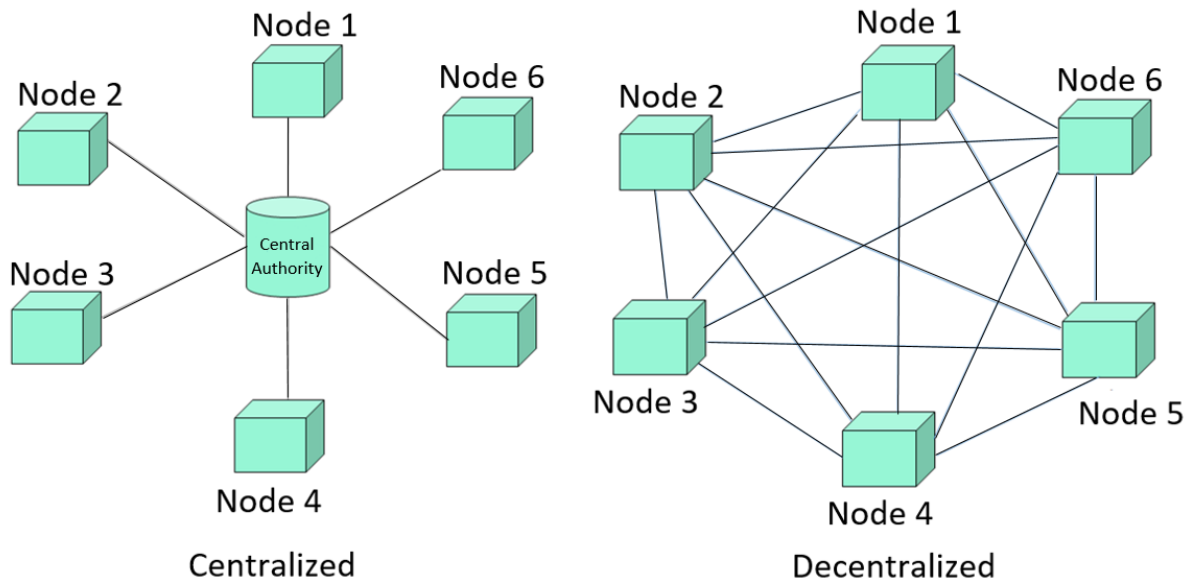


Figure 3.14: Decentralization of the System

Mining

In general sense the term mining refers to extracting meaningful information from a huge amount of resources. On the other hand, mining, in the sense of block chain, is the process of adding a block of transactions to the large distributed chain given the consensus and regulations being followed [76]. The individual or authority that does mining is known as miners. Miners generally, compete with each other and given an authority to mine a block which also is decentralized and unpredictable. Thus the process becomes more transparent and can never be predicted as no one knows who will mine. In our case, it's a mathematical or computational problem for which all the miners have to compete for solving. In more general sense, the problem is a random hash based mathematical problem which can be solved only by random guessing [77]. The mining machines have to randomly guess the problem and have to solve this problem. In this case the nonce plays an important role. The nonce controls the difficulty of the problem by adding zeros at the head of the hash. Such as, if there are more zeros at the front the problem tends to be more difficult. At the very end the miner who solves the problem gets the chance to mine the block with all the transactions. Before mining he has to follow the consensus and the smart contract. Thus a block is being mined and gets added to the blockchain.

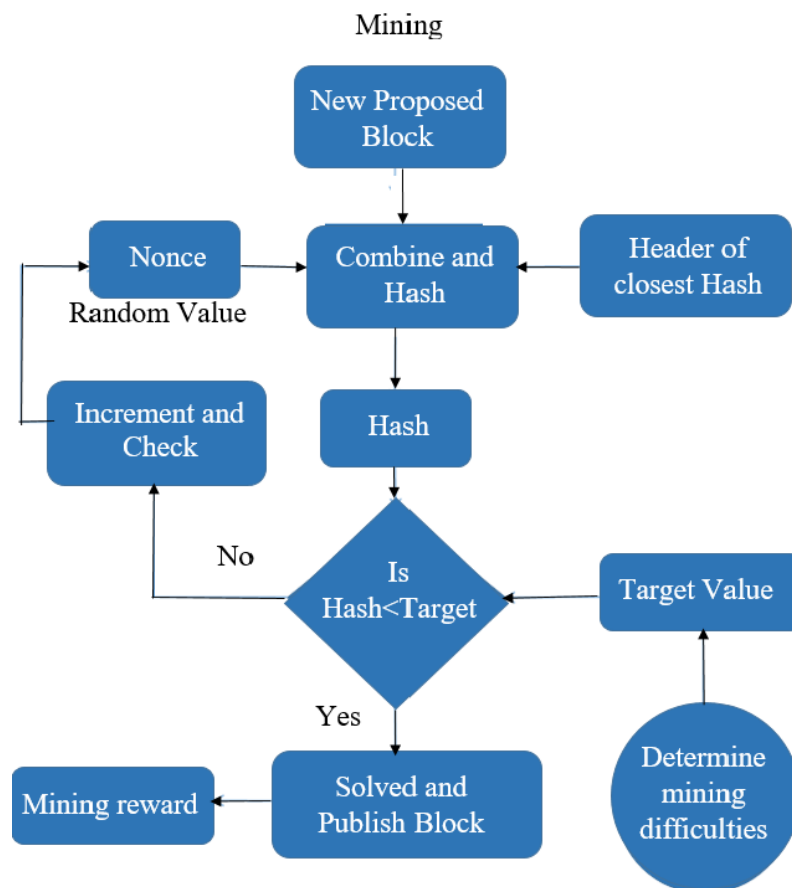


Figure 3.15: Control Flow by the Smart Contract

3.5 Cryptography

The term cryptography specifies that something that is encoded with such a key that cannot be turned back to previous form without the actual key. The main building block of blockchain security is cryptography or hash. In our blockchain framework the whole system is such that no one knows the actual information of other without the key or the willingness of that individual. For instance, if a person wants to transact any kind of information within the blockchain with someone else, he has to send the information to a key that is known as a public key [78]. The sender will also send this information via his public key so that no one can willingly attack his information. We designed the system such that this public key is known to everyone for transaction of information or for tracking that individual. On the other hand, in our system another key that only the user knows and can use for his own purpose is the private key. The users in the last stage uses this private key for accessing his own information and for modifying it. This is very personal and the leakage of it can create serious threat to that person as by comparing the public and private key any individual might trace the actual key of his hash and can decode his hash and can create serious threat to the other person. There might be other stages of security in the blockchain other than this public and private key which we might go for in future implementation. This will be used for added security in the blockchain. But adding more intermediate layers might create computational complexity and harness the efficiency of the system.

3.5.1 Public Key

In our system, public key is used for publicly showing the information of an individual that is everyone in the blockchain knows who that person is and his other information via this public key. Besides, in the blockchain the public key is used to transact information amongst individuals. Everyone receives information and data via this public key and that gets added to the person's account via UTXO [79]. In our system the public key is hashed using the Gmail account, the person's National Identification Number (e.g. NID, SSN, SID), his birth date and a security number provided by us. We used SHA256 hash which is unpredictable so far. SO, the hash is never predictable.

3.5.2 Private Key

Other than public key in our framework, the key that is used to access all the information of a person is known as Private key. The private key is used for accessing all the information of that person and to do modifications and transaction in his account. Any individual in our blockchain system has to access his account via his private key and can transact and do modification in his account only via this private

key [80]. The leakage of any individual's private key can create a serious threat to that person's information because comparing both the public and private key any individual can find the actual key that was used to hash the keys and can take access of that person's account. As like the public key we used Gmail account, the person's National Identification Number, his birth date, a security number and besides all these an password provided by that person to create this private key [81].

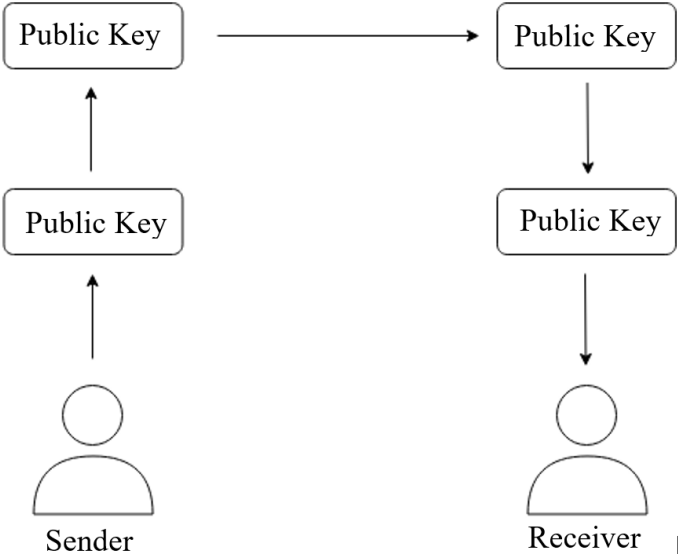


Figure 3.16: Secured Blockchains Transaction Protocol Using Public and Private key

Chapter 4

Implementation

The key idea of Blockchain is to create a system which can add information to blocks and linking them in such a way so that data users can keep track of data integrity.

Talking about our system after initiation, the first block that is created is called the genesis block. The genesis block can be a dummy block or contain information depending on the way it has been configured. In our system we created a dummy genesis block as it doesn't contain any information rather than its hash, Nonce and Timestamp. As information starts to get added to the next block, after certain condition, the block will be mined by a trusted miner who will add the block to the Blockchain. The mined block will contain the hash of the previous block. The next block which will be mined will contain the hash of the currently mined block. In this way each block contains the hash of the previous block and thus creates a chain which can be traced back to the genesis block.

As information piles up and more chains are created forming a network of chains, after a certain interval in our system, each chain will compare itself to check if it is the longest chain or not. If the chain is the longest chain it will do nothing, otherwise it will replace itself with the longest chain. In this way each chain tries to achieve the state of the longest chain.

In the framework that we are implementing each block to be mined, the miners have to generate a hash value of the current block by increasing the Nonce value in such a way that the hash value is less than the target hash value. The winning miner will get the opportunity and the award of mining the block and adding it to the Blockchain. A relation can be seen between the numbers of miners and the time interval in solving the problem to mine a block. Before adding each transaction to the block, the transaction has to be validated. Otherwise it makes the system useless. Therefore, a system to validate the transaction has been designed and implemented. Moreover, while a shorter chain tries to copy the longest chain, it also has to validate if the chain is valid or corrupted. Otherwise, the shorter chains could start to duplicate corrupted data and turning the system useless. Therefore,

before replication, in our system there is a process of validation included which has been done in this model.

For the purpose of storing a smart device or information in the Blockchain, the first transaction linked with the device or information will not contain any sender identification and will only contain a receiver identification. Thus, information can be linked to its first appearance on the Blockchain network which might provide important analytic for different purposes like research.

In our Blockchain network both tangible and non- tangible elements can be transferred like smart devices which is a tangible element and funds which is not a tangible element. The system also can provide a list of information linked to a specific identity.

Due to the nature of Blockchain technology, it is possible to trace devices linked to a certain identity or id if the person has only one identity thorough which the entity conducts all the transactions. Therefore, it could be harmful for individuals to have such kind of privacy issues. Therefore, the system changes the user's identity with each transaction such that it is no more possible to trace information to an individual identity.

The Blockchain model consists of some particular functions and regulations for transaction and verification. The described Blockchain model has been implemented using Python programming language.

4.1 Flask web Framework

Flask is a web framework implemented for Python which has similarities and differences with python's Django web framework. Flask can be used to create a Python server for hosting web. Thus we used flask to host a blockchain in our local machine and it's initialization look as follows:

```
1 from flask import Flask
2 app = Flask(__name__)
3
4 @app.route('/ValidateTransaction')
5 def check():
6     CheckTransaction();
7
8     return flag
9
10 if __name__ == '__main__':
11     app.run()
12
```

Figure 4.1: Flask basic implementation.

Applications, creating POST and GET APIs, Flask library is directly utilized in the Python code to imitate a server which can host web-pages or respond using JSON response in our system. We also used Flask to run servers on open ports. More than one chain can be initiated in a single server with different port numbers for each chain. In the implemented model Flask Library has been used to create a local server for the Blockchain where users can transact and view information related to Blockchain. The APIs created for the Blockchain have been shown in the following table with description of functionality and type of API.

Name of API	Description	Type
Mine_block	The API initiates the mining process of the Blockchain. It returns the mined block information	GET
Get_Chain	The API returns the blocks of the Blockchain	GET
Is_valid	Checks the Blockchain and returns a response with the information of the validity of the chain	GET
Add_transaction	Validates a transaction and adds the queue for adding to the Block which will be mined.	POST
Connect_node	This is sent to the Blockchain network when a new chain has been initiated.	POST
Replace_chain	Check if the current chain is the longest chain or not and if not replaces the chain with the longest chain.	GET

Table 4.1: API information table

4.2 Blockchain functions

For the proper operation of the Blockchain model, multiple functions are required. Our model has the functions that are essential for the Blockchain system. The model contains the following Functions.

4.2.1 Create Block

After mining each block, devices need to start to collect transactions to insert in the new block. New block has to be generated. The function validates the proof of work and previous hash value and adds the block to a chain in the Blockchain network. The created block is also timestamped.

Algorithm:

1. Create Block dictionary with index, timestamp, proof, previous hash, and data key value pair.

2. Empty data for storing next state data.
3. Append Block to chain.
4. Return Block information.

After the proof has been calculated, this function takes the proof the hash of the previous block and timestamp to generate a new block. The pseudo code for the new block is given bellow

```
FUNCTION create_block(self,proof,previous_hash):
block j- 'index' : len( chain) +1 ,
'timestamp' : str(datetime.datetime.now()) ,
'proof' : proof,
'previous_hash' : previous_hash,
'transactions' : transactions
transactions j- []
chain.append(block)
RETURN block
```

4.2.2 Return previous block

In the Blockchain system it is necessary to get the information of the previous block. The functionality of the function is to return the previously mined block from the chain.

Algorithm:

1. Gets index of current state.
2. Returns Block with previous index value.

Firstly the system gets the value of the current state. The current state contains information about the previous block. The system just returns the previous block address. pseudo code for the function is given below:

```
FUNCTION get_previous_block(self):
RETURN chain[-1]
```

4.2.3 Proof of Work

This is one of the most important functions of the Blockchain. This function calculates the Nonce value for which the hash of the block is lower than the target hash and returns the value so that the block can be mined.

Algorithm:

1. Set value of new proof to one.
2. Set proof validity to false.
3. Check proof validity, if true return new proof value, else go to next step.
4. Calculate hash value for new proof.
5. Compare hash value with target hash value.
6. If hash value is less than target hash value set proof validity to true.
7. Increase new proof value by one.
8. Go to step 3.

To generate the hash with the value lower than the target hash, the system increases the value of the nonce from 1. For a given nonce and a specific timestamp a system will generate a hash which will be lower than the target hash. The pseudo code for the proof of work is given below:

```
FUNCTION proof_of_work(self,previous_proof):  
new_proof j- 1  
check_proof j- False  
WHILE check_proof is False:  
hash_operation j- hashlib.sha256(str(new_proof**2 - previous_proof**2).encode()).hexdigest()  
IF hash_operation[:4] = '0000':  
check_proof j- True  
ELSE:  
new_proof += 1  
ENDIF  
ENDWHILE  
RETURN new_proof
```

4.2.4 Check chain validity

This functions checks if the chain is valid or not by checking each block's validity.

Algorithm:

1. Set previous block to first block of the chain.
2. Set block index to one.
3. If block index is less than chain length go to next block, else return true.
4. Set block to block value of index from chain.
5. If previous hash value of block is not equal to hash value of previous block return false.
6. Set previous proof to proof value of previous block.
7. Set proof to proof value of block
8. Calculate hash value using previous proof and proof value.
9. If hash value is less than target hash value return false, else go to next step.
10. Set block value to previous block.
11. Increase block index by one.
12. Go to step 3.

The system starts from the first last block of the chain and recursively checks each block of the chain to compute and verify the whole chain until it reaches the genesis block. Pseudo code for the function is given below:

```
FUNCTION proof_of_work(self,previous_proof):  
new_proof j- 1  
check_proof j- False  
while check_proof is False:  
hash_operation j- hashlib.sha256(str(new_proof**2 - previous_proof**2).encode()).hexdigest()  
IF hash_operation[:4] = '0000':  
check_proof j- True  
ELSE:  
new_proof += 1  
ENDIF  
ENDWHILE  
RETURN new_proof
```

4.2.5 Hash function

The system users hash values extensively. Therefore, it needs a function that can return hash value for a specific object. The following pseudo code has been written below:

```
FUNCTION hash(self,block):  
encoded_block j- json.dumps(block,sort_keys=True).encode()  
RETURN hashlib.sha256(encoded_block).hexdigest()
```

4.2.6 Add Transaction

The purpose of this function is check the validity of a transaction and adding it to a queue which then will transfer the data to a block for being mined. After a transaction has been done and all other criterias are being fulfilled such then this function occurs to add a transaction to the blockhain. For example, when a transaction has occued and has passed the proof of work and retained all other integrity of consensus as well as the smart contract, then it passes to the miner and the miners compete for mining. If the mining happens this function is called for adding a transaction to a block and thus this block gets added to blockchain.

Algorithm:

1. Get sender, receiver and transaction information.
2. Create a Dictionary with the information.
3. Append the information the data Queue.
4. Increase previous block index value by one.

The system receives the transaction information and converts it into a JSON format which then is added to the transactions queue.

```
FUNCTION add_transaction(self,sender,receiver,data):  
transactions.append('sender' : sender,  
'receiver' : receiver,  
'Data' : data)  
previous_block j- get_previous_block()  
RETURN previous_block['index'] + 1
```

4.2.7 Get Node

Another important function of the network is the get Node function. The functionality of this node is to generate a JSON list of all known chains in the Blockchain network and transmit the data so that other chains can find out the chain addresses of the Blockchain network. Both Get Node and Add Node are mandatory functions for full operation of the Blockchain system. Without this functions the each chain in the Blockchain would have to be manually configured to connect to each of the other chains. Even then new chains could get left out of the Blockchain system.

4.2.8 Add Node

The purpose of this function is to add a new node to its list of known list. A new node means a new chain which has been initiated. Without knowing other nodes in the Blockchain network this chain or node cannot validate the blocks and chain and also cannot check if it the longest chain in the network or not.

Algorithm:

1. Get address.
2. Add parsed value to nodes

The function receiving a node value adds is to nodes queue. The node queue contains the list of all the known nodes to this node or chain. The pseudo code is given below:

```
FUNCTION add_node(self,address):  
parsed_url j- urlparse(address)  
nodes.add(parsed_url.netloc)  
ENDFUNCTION
```

4.2.9 Replace chain

Replace chain is another important function of the Blockchain algorithm. The purpose of this function is to check if the current chain is the longest chain or not among the chains in the list of known nodes.

Algorithm:

1. Set network value to nodes.

2. Set longest chain value to null.
3. Set maximum length value to chain length.
4. Get chain length from node.
5. If chain length is greater than longest chain, set longest chain value to node's chain length, else go to step 8.
6. Copy nodes chain and replace with own chain.
7. Return.
8. Go to next node.
9. Go to step 4.

The function gets the list of nodes from the nodes queue and iterates through the nodes to compare the node's chain with the longest chain. If it finds a longer chain, it replaces the value with the longest chain. The pseudo code is given below:

```

FUNCTION replace_chain(self):
network j- nodes
longest_chain - None
max_length - len( chain)
for node in network:
response - requests.get(f'http://node/get_chain')
IF response.status_code = 200:
length - response.json()['length']
chain - response.json()['chain']
IF length > max_length AND is_chain_valid(chain):
max_length - length
longest_chain - chain
ENDIF
ENDIF
ENDFOR
IF longest_chain :
chain - longest_chain
RETURN True
ENDIF
RETURN False

```

4.3 Interacting with API

For interacting with the Python APIs Postman platform was used. Postman is a platform created for making API development easier. Postman can show received data in a formatted manner and also API keys can be easily defined in Postman. Postman provides many useful features for working with APIs.

While testing the whole system we used postman to verify whether the system is working perfectly by continuously sending proper GET and POST commands as per requirements. Such as while a transaction is being done the request for doing the transaction is a POST request. Because, if a user wants to do transaction he has to fulfill the criterion for doing that transaction thus he has to POST the request to the server with necessary information. POSTMAN does these in a very fruitful manner as it formats all the API calls with ease and allows us to do such kind of post requests.

On the other hand if we consider the works of the blockchain, we need to send different GET as well as POST requests, such as while the `get_chain` function is called, it is a GET method as it works such that the blockchain is requesting to get all the blocks for the current chain and the server in reply responds. `Mine_block` is also a GET request as the blockchain is not requesting with any data input to the server. Thus POSTMAN interacts with the blockchain as such. The whole procedure of the whole system is describe on later part.

4.4 Workflow of Blockchain

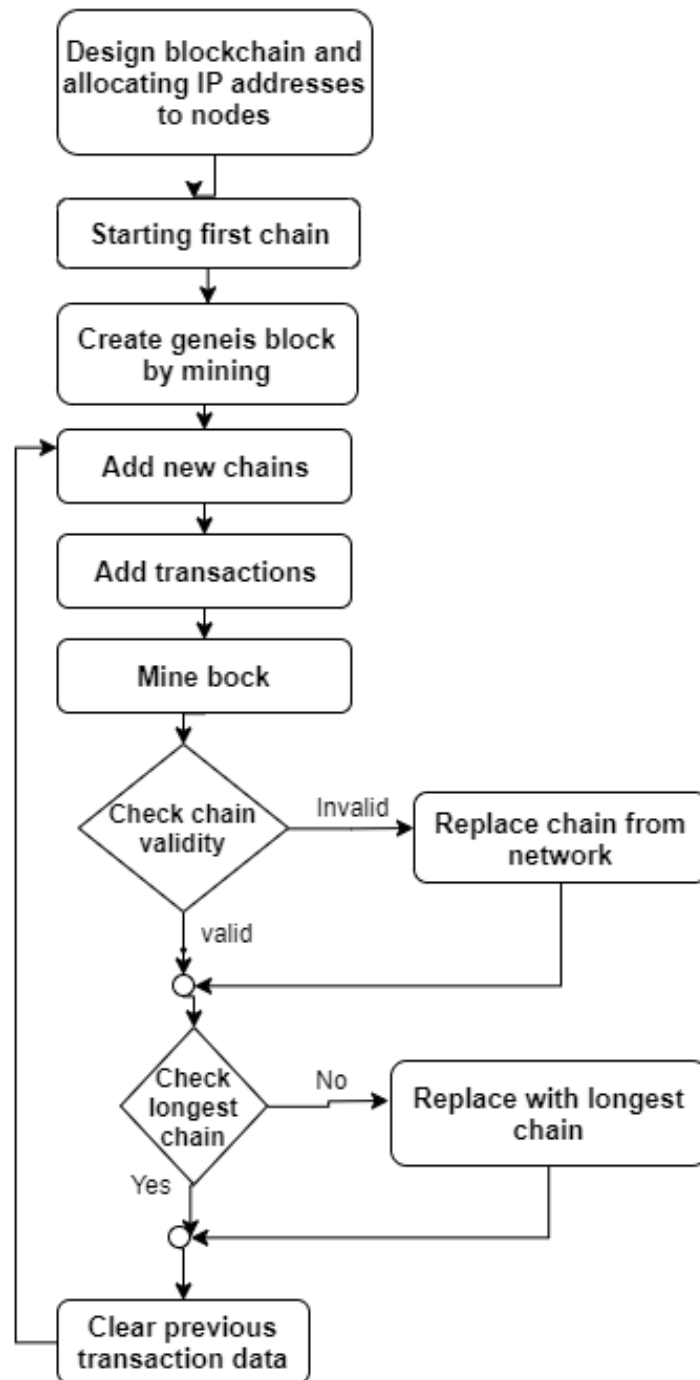


Figure 4.2: Blockchain Flowchart

4.4.1 Designing Blockchain

In this step we plan, analyze and design a Blockchain and allocate IP addresses to the nodes or chains. Three chains were implemented in three devices with address 192.168.0.100:5001, 192.168.0.152:5002, 192.168.0.112:5003 .

4.4.2 Chain initialization

To start the Blockchain we have to run the Blockchain algorithm and mine the first block of the Blockchain which can be a valueless block or a block with dummy value. This block is also known as Genesis block. All chain start block validating from this block. The System was initiated in the device with the address 192.168.0.100:5001 as the first chain.

4.4.3 Connect node

In this step Blockchain adds all the nodes in its node queue. This step is necessary to create a Blockchain network. Without this step, the chain cannot locate other chains in the network and cannot validate the chain and check if it is the longest chain.

4.4.4 Add transactions

All the validated transactions are stored in the data queue. In this step the Blockchain copies the data to the block which will be mined.

4.4.5 Mine block

To mine the block the chain needs to solve the proof of work. After solving the proof of work the function will return the nonce value for which the hash of the block is less than the target hash value. Using the hash value the block will be mined.

4.4.6 Check validity

The chain needs to check its validity for data integrity. If the validity is returned false, the chain has to replace its value using the validated data from the Blockchain network. In this case there might be multiple invalid chains in the network. The chain needs to replicate data from only the valid chains, in this case majority chain which have the same values is considered the valid chains. The chain needs to find the longest chain among those chains and replicate that data.

4.4.7 Check longest chain

The Blockchain needs to check if it is the longest chain. If it is not the longest chain it needs to locate the longest the chain and replace the current chain with the longest chain. This way the chain can still keep up with the other chains in the network.

4.4.8 Clear data

In this step the chain clears the entire data queue to accommodate new data for the next block. The data from the data queue have been already added to the previously mined block. Therefore, clearing the queue will not cause any loss of data.

4.4.9 Data format

The model uses an API based approach for communication. The communication happens through JSON data packets. Our model can store multiple formats of Data.

4.4.10 Usecase Diagram

Interaction between users as follows:

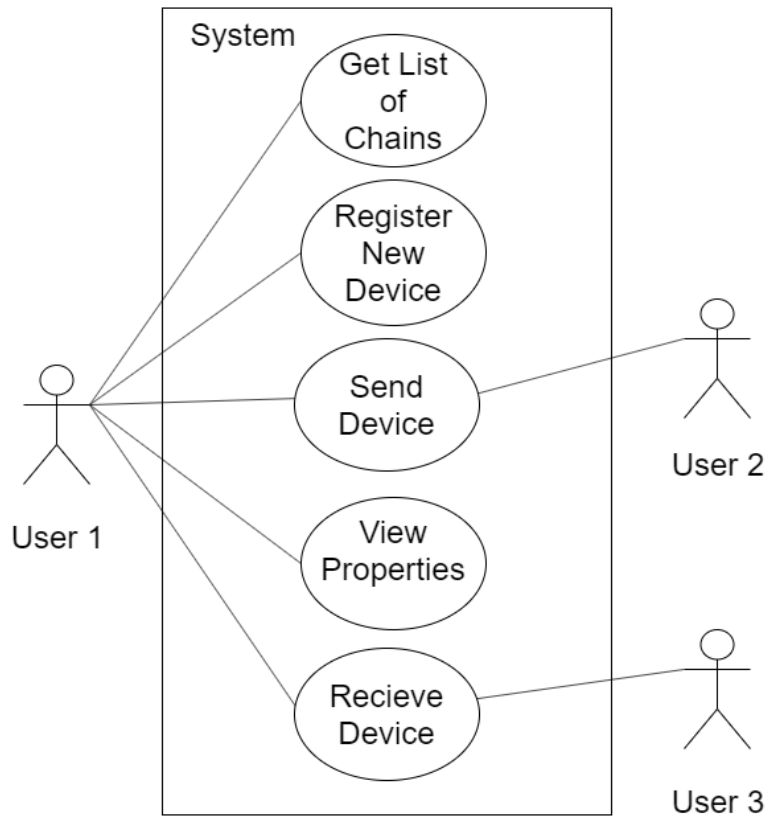


Figure 4.3: Blockchain usecase diagram

4.4.11 GSM based Device Registration

Devices that utilize GSM technologies can be identified using IMEI number. Our model users IMEI numbers to identify those devices. Each GSM based device is initialized using this packet. The JSON packet contains only receiver information. No sender information is included in this packet.

4.4.12 Smart Device Registration

All devices have a serial number embedded in the hardware. Our model user the Serial Numbers of those devices to link those devices in the system if there are not any other form of identification number in the device. The JSON packet contains only receiver information. No sender information is included in this packet.

4.4.13 Vehicle Registration

All vehicles have chassis numbers printed in the body so that they can be traced. This is an internationally recognized system of identification for vehicles. Our model uses vehicle chassis numbers to identify the vehicle and link it to the system. The JSON packet contains only receiver information. No sender information is included in this packet.

4.4.14 Land Registration

Land and real-estate based properties have registration numbers stored in government database. In this way the government tracks who owns what property. We utilized registration numbers to link properties to individuals in our system. The JSON packet contains only receiver information. No sender information is included in this packet.

4.4.15 Transaction Data

In the transaction JSON packet, it has 3 fields. Firstly, the sender. To verify the transaction, the system has to check if the sender is the original owner of the property. The system also has to check if the sender is currently in possession of the property. If both conditions are true, only then the sender is eligible to conduct the transaction.

Secondly the packet contains receiver information. The system also needs to verify if the receiver is a registered user or not. If the receiver is registered, only then the receiver is eligible for the transaction. Moreover, the system also needs to check if the sender and receiver are not the same person. If they are the same person, the system will not verify the transaction.

Finally, the JSON packet contains the registration id. This id is unique for each property. And also this information can be classified. The system needs to verify that firstly the property has been introduced in the system using any registration packet. If the property is not registered in the system, the transaction is not valid.

Chapter 5

Result and Discussion

The prototype of the Blockchain system accommodated three independent chains. Each has been initiated on a singular device using multiple socket addresses. Three PORT from the registered port range of 1024-49151 was used for choosing the desired ports. The three ports are 5001, 5002, 5003. The dedicated IP address for local server is the predefined address 172.0.0.1 IP address.

Chain Name	IP Address	Port Address
Chain 1	172.0.0.1	5001
Chain 2	172.0.0.1	5002
Chain 3	172.0.0.1	5003

Table 5.1: Blockchain network chain address table.

5.1 Data-set

For testing purposes, data for transaction has been randomly generated in python. Both sender and recipient address was generated randomly to represent true scenario where the sender and recipient username will be 128 bit HASH address. The data contains information about the device where the length of the address can vary depending on the category of data. The data information can store multiple formats of value. In our test we used multiple users automated configuration to add transactions to the Blockchain system. A total of six thousand transactions were generated for depositing in the Blockchain. Numbers of blocks in the starting Minute are represented in figure 5.1.

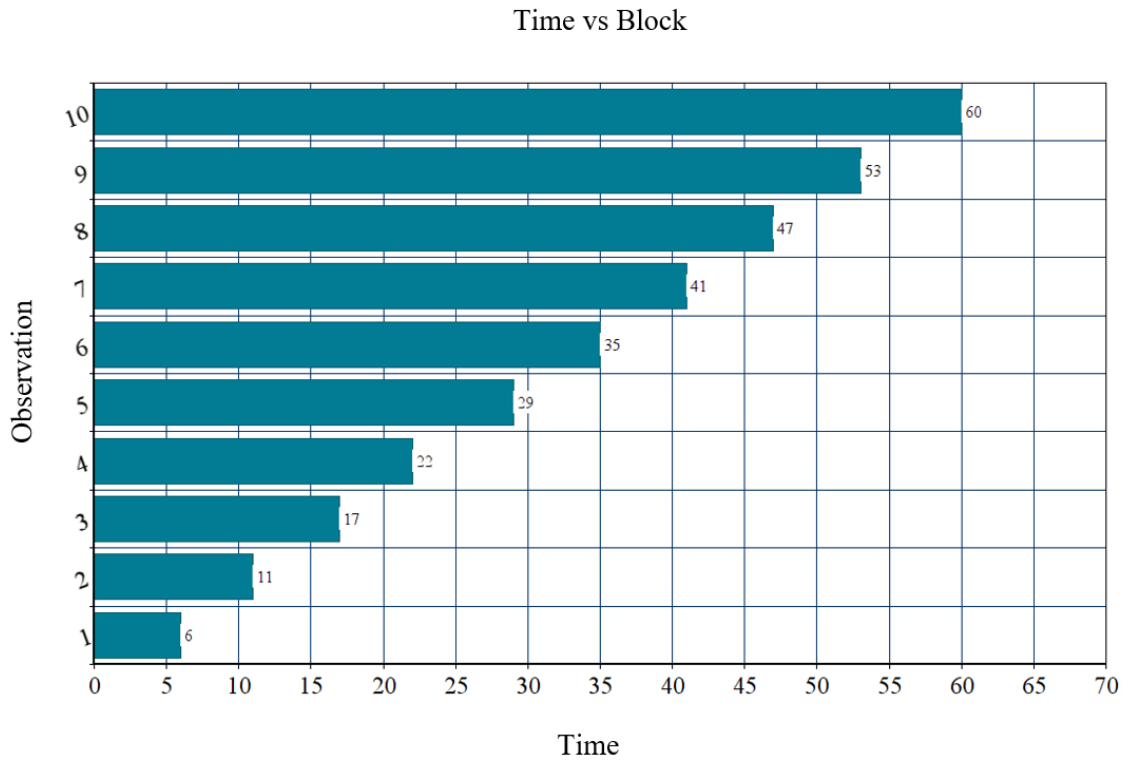


Figure 5.1: Time vs Block generation.

Each user system added a transaction based on a random interval of three to seven seconds for adding transaction. Each block mining is initiated at an interval of ten seconds after the previous block has been added to the blockchain. A tabular representation is shown for the first twenty blocks generated in a chain of the Blockchain network in table 5.2 containing the index of the block, value of Nonce or proof, timestamp of when the block is generated and the number of transaction in the block.

The transaction were added using four automated users whom added each transaction within a given interval so that the result could be duplicated in an appropriate environment. The mining interval was chosen based on testing based of optimal number of transactions in each block. The numbers of Blocks generated in a given interval has been represented in the figure 5.1. The six thousand generated transactions were added within approximately twenty minutes. Increasing the leading zeros for target Hash will increase the time required to deposit the same amount of transactions will increase dramatically as time required to mine each block increases exponentially based on the number of leading zeros in target Hash.

The optimal settings for our system was chosen as four leading zeros and the power of the Hash generating equation was chosen to be two, thus a curve.

Index	Proof	Timestamp	Number of Transactions
1	1	12:37:39.409978	0
2	533	12:37:56.748273	5
3	45293	12:38:13.207833	8
4	21391	12:38:29.520651	9
5	8018	12:38:45.662530	8
6	48191	12:39:02.195232	8
7	19865	12:39:18.505137	10
8	95063	12:39:35.292175	11
9	15457	12:39:51.544803	12
10	15479	12:40:07.758151	12
11	7889	12:40:23.936847	15
12	72474	12:40:40.578504	14
13	126616	12:40:57.611848	13
14	64161	12:41:14.394433	11
15	144125	12:41:31.556901	13
16	2492	12:41:47.694196	13
17	22592	12:42:03.987885	14
18	107780	12:42:20.845995	16
19	47346	12:42:37.354394	15
20	46891	12:42:53.895067	14

Table 5.2: Transaction in a single chain of Blockchain network.

The Blockchain system has multiple states. Each state can have different properties and different number of chains. A representation of different states of the Blockchain system is given in figure 5.2 where the Blockchain system contains three distinctive chains and each chain has been initiated at a different time intervals for analysis of the performance of the Blockchain system. In the system each chain was capable of identifying other operating chains in the network and communicate with the chains. Thus, was able to collect information about the operational chains present in the Blockchain and compare the information of the blocks for verification.

Number of leading Zeros	Time required (Seconds)
4	1.0849964618682861
5	5.25169825553894
6	20.6606292724609
7	1473.9516570568085

Table 5.3: Proof of work complexity based on leading zeros of Target Hash.

Another perspective of the computational time required can be the power of the values in the equation of generating the valid Hash for which the Hash values is lower than the target Hash value. In our analysis, the Blockchain system prototype

used an equation containing a power of two. The equation for Hash generating is given in table 5.3.

$$\text{Hash value} = \text{new_proof_Hash}^2 - \text{previous_proof_Hash}^2$$

Now for different orders of the equation we get different computational time. A representation of the time required for different power values is given in table 5.4. The time required to compute Hash value for different order of the equation have very low differences. Therefore, it can be concluded that computational time required to generate Hash value is not largely dependent on the order of the equation.

Power value	Time required (Seconds)
1	1.2212324142456055
2	1.145613670349121
3	1.0227384567260742
4	1.0944087505340576
5	1.2442378997802734
6	1.275045394897461
7	1.3591926097869873
8	1.3885324001312256
9	1.3314905166625977
10	1.3451454639434814

Table 5.4: Proof of work complexity based on Order of equation.

5.2 Performance

The proof of work algorithm used in our model was based on generating a target hash of a value range containing four leading zeros which gave a large value pool to generate a acceptable hash. For each increasing zero, the computational power and time required to generate a target hash increases rapidly. The following table shows the time needed to generate a valid Hash for which the Hash value is lower than the target Hash.

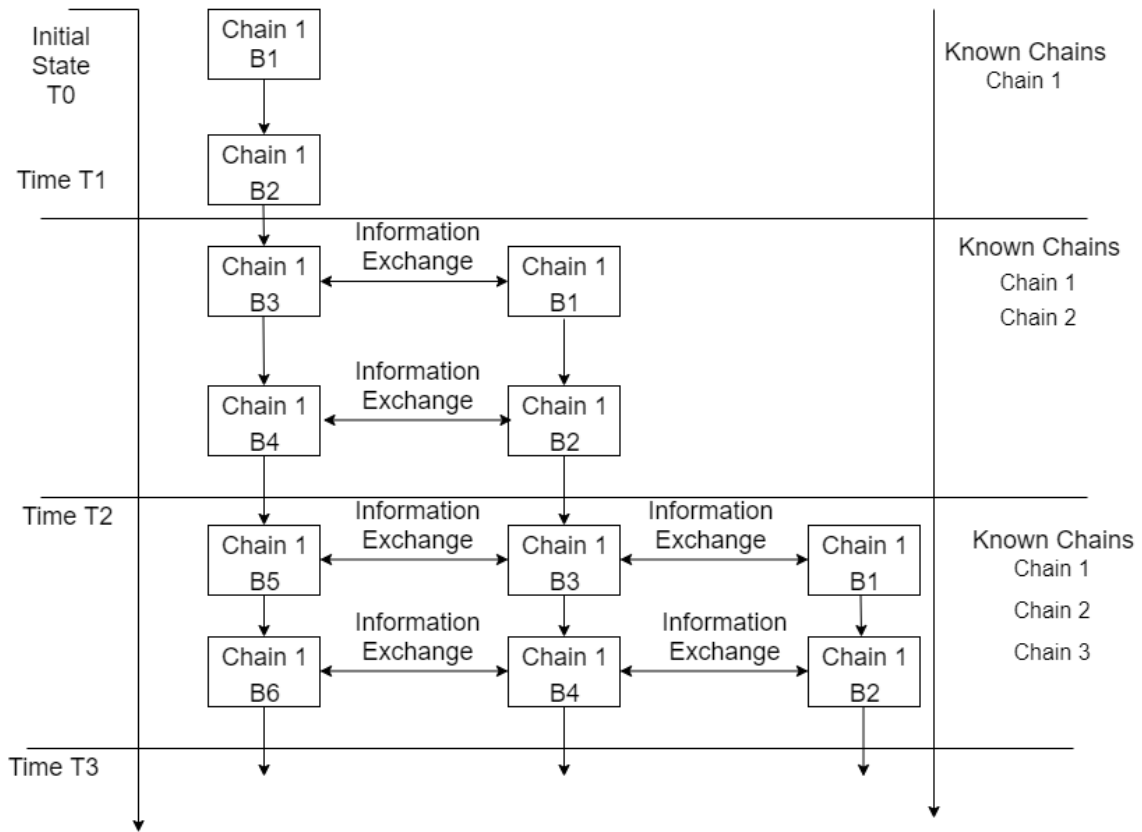


Figure 5.2: Different states of the System

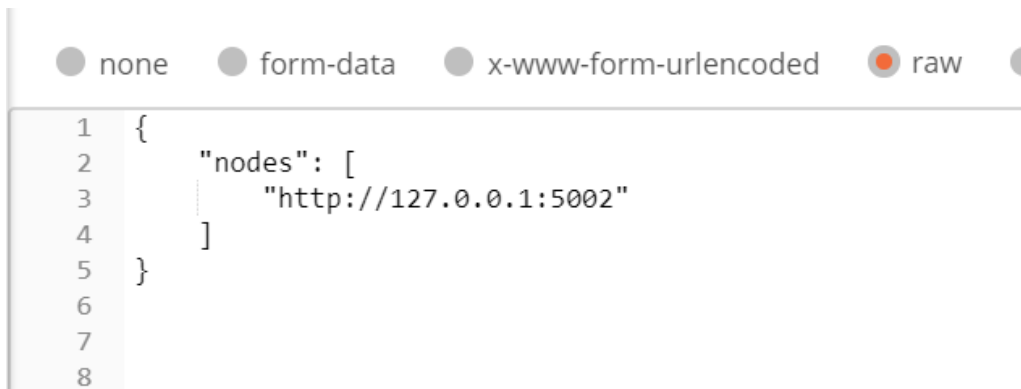
A model of different states of the Blockchain system needs better understanding for a clear visual of what is expected from each chain in the network and how the network should operate. The information of the model can create a visual representation of an optimal operation. Information of nodes at different states of the system are reflected on the table 5.5.

State	Time	Known chains in system
Initialization of system with chain 1	T0	Chain 1 (127.0.0.1:5001)
Initiation of chain 2	T1	Chain 1 (127.0.0.1:5001), Chain 2 (127.0.0.1:5002)
Initiation of chain 3	T2	Chain 1 (192.168.0.100:5001), Chain 2 (127.0.0.1:5002), Chain 3 (127.0.0.1:5003)

Table 5.5: State change table

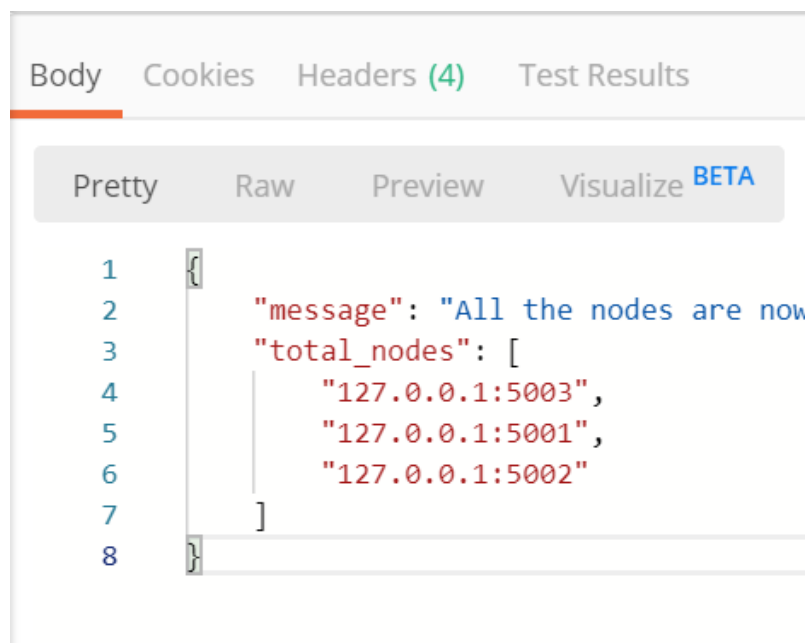
5.3 Distributed System test

Each chain in the Blockchain system was successful in connecting to the Blockchain system and receiving and sending node information among the chains in the Blockchain system. In our test, we created the chain 127.0.0.1:5001 and 127.0.0.1:5002. Chain 127.0.0.1:5002 contained the addresses of chain 127.0.0.1:5001, 127.0.0.1:5002, 127.0.0.1:5003. After this, we added the address of 127.0.0.1:5002 to chain 127.0.0.1:5001. Then when we retrieved the list of known addresses from 127.0.0.1:5001, the chain was successful in getting all the chains from the network. In this way having the address of one chain of the Blockchain network, each chain is capable of collection information of the whole Blockchain network.



```
1 {
2   "nodes": [
3     "http://127.0.0.1:5002"
4   ]
5 }
6
7
8
```

Figure 5.3: Network address given to chain.



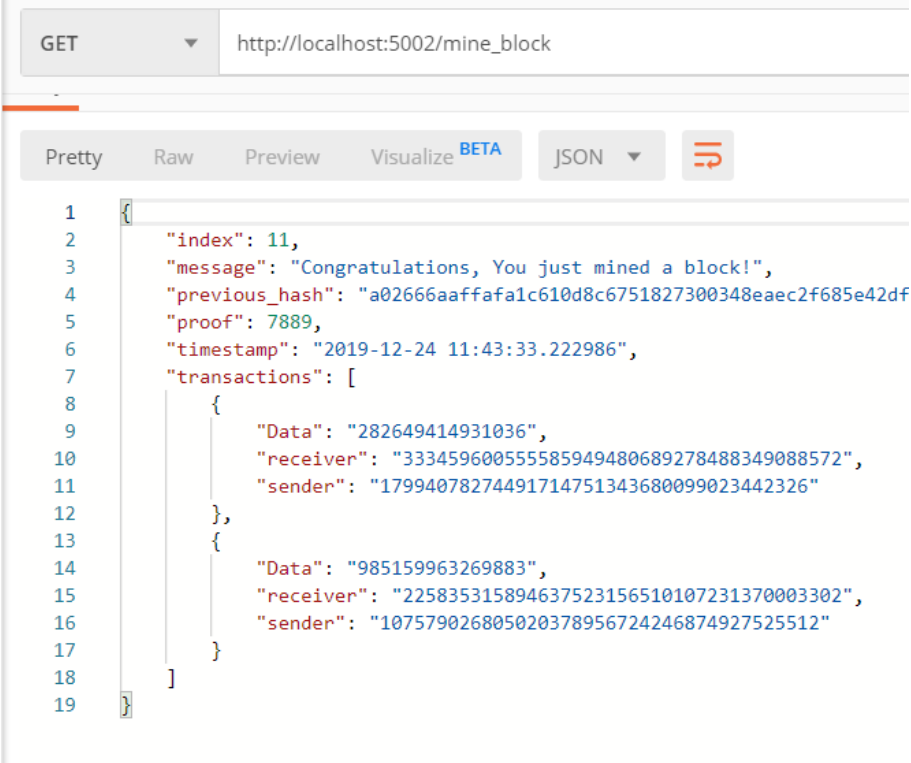
```
1 {
2   "message": "All the nodes are now",
3   "total_nodes": [
4     "127.0.0.1:5003",
5     "127.0.0.1:5001",
6     "127.0.0.1:5002"
7   ]
8 }
```

Figure 5.4: Parsed network addresses from other chains.

5.4 Block mining

For the proof of work we provided a fairly simple calculation for which the hash value target is any value with hash value where first four MSB bits are equal to '0000'. Our system was able to solve the proof of work in each attempt with run time less than 1 second.

However, in a real scenario the target hash value will be a very low value so that the mining device requires more computational time to solve the problem. Moreover, the arithmetic operation to solve the proof of work was an equation with n^3 time complexity. Therefore, the complexity is can be considered fairly simple. The data format is shown below-



```
GET http://localhost:5002/mine_block

Pretty Raw Preview Visualize BETA JSON

1 {
2   "index": 11,
3   "message": "Congratulations, You just mined a block!",
4   "previous_hash": "a02666aaffafa1c610d8c6751827300348eaec2f685e42df",
5   "proof": 7889,
6   "timestamp": "2019-12-24 11:43:33.222986",
7   "transactions": [
8     {
9       "Data": "282649414931036",
10      "receiver": "333459600555585949480689278488349088572",
11      "sender": "17994078274491714751343680099023442326"
12    },
13    {
14      "Data": "985159963269883",
15      "receiver": "225835315894637523156510107231370003302",
16      "sender": "107579026805020378956724246874927525512"
17    }
18  ]
19 }
```

Figure 5.5: Mining the Block

5.5 New Block generation

There can be two approaches to generate new blocks. One is to generate new block after a certain amount of transactions have been added to the queue. And the second approach is to wait for a certain amount of time to generate a new block. Our system was developed using the second approach. Each block mining initiated after ten seconds of the completion of the previous block. Before, the block generation is

initiated all the transactions in the transaction queue are added to the block. And the transaction queue is cleared for new transactions. The new blocks are appended to the chains.

```

127.0.0.1 - - [23/Dec/2019 12:22:30] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:34] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:39] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:42] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:22:43] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:43] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:22:47] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:51] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:55] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:22:56] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:22:57] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:22:59] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:03] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:04] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:07] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:09] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:10] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:11] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:11] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:15] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:16] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:19] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:20] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:23] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:23] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:24] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:27] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:27] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:31] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:33] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:35] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:37] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:38] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:39] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:40] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:43] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:43] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:47] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:47] "POST /add_transaction HTTP/1.1" 201 -
127.0.0.1 - - [23/Dec/2019 12:23:50] "GET /mine_block HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:51] "GET /replace_chain HTTP/1.1" 200 -
127.0.0.1 - - [23/Dec/2019 12:23:51] "POST /add_transaction HTTP/1.1" 201 -

```

Figure 5.6: Operation of a chain in the Blockchain network.

After a transaction is done the data blocks are created. Each individual blocks looks as follows:

```

92  {
93    "index": 6,
94    "previous_hash": "d4c7008ed7c60165299f6a9d11ae725033cdb2266076371230ecc6e19b87e0df",
95    "proof": 48191,
96    "timestamp": "2019-12-24 11:42:19.261362",
97    "transactions": [
98      {
99        "Data": "935125084539406",
100       "receiver": "333088699347638993827334312235648775083",
101       "sender": "239477971193824179040884085663268908420"
102     },
103     {
104       "Data": "298664808388333",
105       "receiver": "82104697248375856987101812394335086031",
106       "sender": "222469072268023397774830132435593716690"
107     },
108     {
109       "Data": "128098002150203",
110       "receiver": "254367483671343324498974965422458741993",
111       "sender": "107378613659791269380637872499738613621"
112     },
113     {
114       "Data": "128154506431984",
115       "receiver": "110774605045799917801156661858342975751",
116       "sender": "183026111882359225737283243517202902724"
117     },
118     {
119       "Data": "134053300560617",
120       "receiver": "295655730488887420748981145018852658788",
121       "sender": "213872749025368957556187952040518377365"
122     },
123     {
124       "Data": "291825362601136",

```

Figure 5.7: All Other Data Blocks

5.6 Longest chain replication

Our system started with chain 1. After chain 1 mined few blocks and chain 2 was initialized in the network, chain 2 ran the longest chain replication algorithm and copied the value from the longest chain which is chain 1 in our network. After few more blocks were added to the chain, chain 3 was initialized. Chain 3 was able to copy data from the longest chain too.

In each time interval each chain contained the same data due to synchronization. In figure 5.8 we see that Node 2 with address 127.0.0.1:5002 contains the same information in the chain as the Node 1 with address 127.0.0.1:5001 even after Node 2 was initiated after Node 1. Node 2 was successful in replicating data from the longest chain.

We wanted to create a system which could store information about ownership of smart devices, and properties like land and vehicles. Our system is able to register new devices. Vehicles and land information successfully. Our system can keep track of newly enter information. Moreover, this system is successfully able to verify exchange of properties.

```

GET http://localhost:5002/get_chain

Pretty Raw Preview Visualize BETA JSON

834 {
835   "Data": "268625912949977",
836   "receiver": "13390436097689995619026380547928841953",
837   "sender": "96086486231043524992606787350682960714"
838 },
839 {
840   "Data": "397783644583073",
841   "receiver": "140126017535741392331759509897036594294",
842   "sender": "47964876142375717367419519982503454314"
843 },
844 {
845   "Data": "297401636325348",
846   "receiver": "221721516470627383416753631351258564271",
847   "sender": "200607683356783556391141485982440545076"
848 },
849 {
850   "Data": "200572548722841",
851   "receiver": "9241993830997622111703987459889248272",
852   "sender": "271542135721630155459986236967307983894"
853 },
854 {
855   "Data": "240453852649785",
856   "receiver": "102962215710150216060419748742384498181",
857   "sender": "148645238729968048308572455434185655297"
858 }
859 ]
860 }
861 ],
862 "length": 18
863 ]

```

Figure 5.8: Chain information in Node 127.0.0.1:5002

```

GET http://localhost:5001/get_chain

Pretty Raw Preview Visualize BETA JSON
{
  834 {
  835   "Data": "268625912949977",
  836   "receiver": "13390436097689995619026380547928841953",
  837   "sender": "96086486231043524992606787350682960714"
  838 },
  839 {
  840   "Data": "397783644583073",
  841   "receiver": "140126017535741392331759509897036594294",
  842   "sender": "47964876142375717367419519982503454314"
  843 },
  844 {
  845   "Data": "297401636325348",
  846   "receiver": "221721516470627383416753631351258564271",
  847   "sender": "200607683356783556391141485982440545076"
  848 },
  849 {
  850   "Data": "200572548722841",
  851   "receiver": "9241993830997622111703987459889248272",
  852   "sender": "271542135721630155459986236967307983894"
  853 },
  854 {
  855   "Data": "240453852649785",
  856   "receiver": "102962215710150216060419748742384498181",
  857   "sender": "148645238729968048308572455434185655297"
  858 }
  859 ]
  860 }
  861 ],
  862 "length": 18
  863 }

```

Figure 5.9: Chain information in Node 127.0.0.1:5001

5.7 Transactions

Our system does not refer to transactions as sending or receiving. Instead, our system considers transactions as form of exchanging ownership of properties. Our system successfully completed transactions after verification. In this case each chain verified if the sender was truly the owner of the property and if currently the sender was in possession of the property. Secondly, each chain verified if the user was capable of receiving the device. For example transaction to self is denied in the system as it is redundant. Our system was successful in evaluating transactions and adding them to the transaction queue. Moreover, the system was able to verify transactions if they were justified or not.

5.8 Discussion

Although the computational power required by our model is lower compared to other Blockchain based systems, the amount of transactions conducted in our system will be higher due to the fact that it handles multiple categories of information. However, our model is capable of replacing multiple systems which can store only one category of information due to the large number of overhead in each system that needs to be conducted. In our system all of the requirements are executed only once compared to multiple times in single category multi class Blockchain. Moreover, our model introduces a new idea of a Blockchain system that can handle multiple categories of information and illustrates how the system operates.

Our model uses a fairly simple approach to tackle the issue of computational requirement by each device. Our current model is capable of being deployed on very low computational power capable devices like Raspberry pi and hand held computers. Even increasing computational power can be handled by shifting the chains to servers where each server will run a chain and the users will interact with the server. The server will interact with the chain and the Blockchain network.

Our proposed model has very few regulations. Due to it being a prototype, there is space for modifying the model according to the needs. The model is scalable due to changeable size of transactions amount needed to generate new blocks and no limitations in how many numbers of chain need to be present to run the network properly. Our model can handle from a few chains to a few thousand chains to a few millions of chains with fairly simple modifications. Moreover using different port numbers, multiple chains can be initialized from a single point.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Our essential objective was to create a system that could store multi categorical information of users and could be linked with national identification (NID). Previously, all blockchain models have been implementing and modeled for working in respect to one form of data or information. Our model was successful in proposing an idea where the blockchain technology can be used for storing various categories or classes of information and successfully administrate the transactions and simultaneously withstand a standard level of throughput, performance, quality of computational work and most importantly security. Our implemented system was successful in proving that our model is sustainable and fully implementable and better performs in contrast to the current Blockchain systems having low throughput, high idleness, and low protection in some scenarios where computational power is limited. The proposed model is also successful in contradicting the current turn of mind for Blockchain technology where systems are considered to be computationally costly and include high data transmission overhead and postponements. Blockchain systems of this manner are proven to be unsuited for regular usage computational devices like smartphones, tablets and notebook modeled devices whom rely on efficient computability and power usage as core for performance benchmark. The proposed model is capable of operation in devices with restricted computational capabilities or considered to have less elevated computability compared to generalized computational devices with regular computational power. The proposed model is capable of information safeguard in an environment where information security is being continuously bombarded with freshly concocted mechanism of attacks. A newer form of lack of security arises from the research and development of quantum computing, however, the technology to secure blockchain systems for such kind of complication is also continuously updating. Our research on establishing a system based on the proposed model also provided us with the opportunity to analyze the scalability of such a system in terms of real world usage where the system needs to withstand attacks and provide standardized quality of service in contrast to currently operating client-server based models. In today's designing a model that maintains standard, follows the concept of a system for providing information storage capabilities with-

out downtime is a challenging task. The analyzation of the implemented system and structure of the model is considerable in developing further full-scale systems capable of real-world operations.

6.2 Future Work

The implemented model is a basic form of Blockchain that contains only the essential functions for operation and analysis. To fully deploy a system which can handle the transactions, there should be a smart contract that considers all the possible cases. Moreover, our system handles multiple categories of information. However, some categories of information might not proof beneficial in our system directly and might need reformation or conversion.

Our system does not consider the possibility of cryptocurrency transaction. However, it might be possible to develop our system in such a way that it might be able to handle cryptocurrency and properties related information simultaneously.

Further research needs to be conducted in respect to handling new users and initialization of properties in the system. Moreover, with the passing of time, the system could get flooded with properties that have been abandoned. Storing those information could turn out to be problematic for the system. A system to remove information from the system might be needed for security purposes of criminal purposes. Therefore, the system should be modified to accommodate such functions.

The model might need updates to the smart contract in future for accommodation of new rules and regulations. Moreover, to handle different categories of data newly introduce in future, the system should have a function to update the smart contract. This functions needs to be implemented in the system. Overall, the system was successful in its operation. But research and analysis needs to be done to improve the system and find vulnerabilities in the system.

Bibliography

- [1] G. Zyskind, O. Nathan, and A. S. Pentland, “Decentralizing privacy: Using blockchain to protect personal data”, in *2015 IEEE Security and Privacy Workshops*, May 2015, pp. 180–184. DOI: 10.1109/SPW.2015.27.
- [2] M. Atzori, “Blockchain technology and decentralized governance: Is the state still necessary?”, *Available at SSRN 2709713*, 2015.
- [3] E. Hughes, “A cypherpunk’s manifesto”, *URL (accessed 3 August 2004): <http://www.activism.net/cypherpunk/manifesto.html>*, 1993.
- [4] S. Nakamoto *et al.*, “Bitcoin: A peer-to-peer electronic cash system”, 2008.
- [5] H. D. Zubaydi, Y.-W. Chong, G.-S. Ham, K.-M. Ko, and S.-C. Joo, “A decentralized consensus secure and authentication framework for blockchain-based healthcare application”, in *Advances in Computer Science and Ubiquitous Computing*, Springer, 2018, pp. 550–556.
- [6] A. Yakovlev, A. Chernikova, M. Livintsova, and T. Lebedeva, “Improving the quality management system of goods and services based on the blockchain concept implementation and quality assessment in the digital economy”, in *E3S Web of Conferences*, EDP Sciences, vol. 135, 2019, p. 03 082.
- [7] I.-C. Lin and T.-C. Liao, “A survey of blockchain security issues and challenges.”, *IJ Network Security*, vol. 19, no. 5, pp. 653–659, 2017.
- [8] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An overview of blockchain technology: Architecture, consensus, and future trends”, in *2017 IEEE International Congress on Big Data (BigData Congress)*, IEEE, 2017, pp. 557–564.
- [9] B. Preneel, “Analysis and design of cryptographic hash functions”, PhD thesis, Citeseer, 1993.
- [10] C. Cachin and M. Vukolić, “Blockchain consensus protocols in the wild”, *arXiv preprint arXiv:1707.01873*, 2017.
- [11] P. K. Sharma, M.-Y. Chen, and J. H. Park, “A software defined fog node based distributed blockchain cloud architecture for iot”, *IEEE Access*, vol. 6, pp. 115–124, 2017.
- [12] N. Z. Aitzhan and D. Svetinovic, “Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams”, *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 840–852, 2016.
- [13] G. Karame, “On the security and scalability of bitcoin’s blockchain”, in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, ACM, 2016, pp. 1861–1862.

- [14] L. Bach, B. Mihaljevic, and M. Zagar, “Comparative analysis of blockchain consensus algorithms”, in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, IEEE, 2018, pp. 1545–1550.
- [15] Y. Sompolinsky and A. Zohar, “Secure high-rate transaction processing in bitcoin”, in *International Conference on Financial Cryptography and Data Security*, Springer, 2015, pp. 507–527.
- [16] J. Becker, D. Breuker, T. Heide, J. Holler, H. P. Rauer, and R. Böhme, “Can we afford integrity by proof-of-work? scenarios inspired by the bitcoin currency”, in *The economics of information security and privacy*, Springer, 2013, pp. 135–156.
- [17] P. Vogel, T. Klooster, V. Andrikopoulos, and M. Lungu, “A low-effort analytics platform for visualizing evolving flask-based python web services”, in *2017 IEEE Working Conference on Software Visualization (VISSOFT)*, IEEE, 2017, pp. 109–113.
- [18] S. Gupta and M. Sadoghi, *Blockchain transaction processing*. 2019.
- [19] M. Swan, *Blockchain: Blueprint for a new economy.* ” O’Reilly Media, Inc.”, 2015.
- [20] J. Vos, C. Lemmen, and B. Beentjes, “Blockchain based land administration feasible, illusory or a panacea”, in *Netherlands Cadastre, Land Registry and Mapping Agency. Paper prepared for presentation at the 2017 World Bank Conference on Land and Poverty. The World Bank, Washington, DC*, 2017.
- [21] M. Crosby, P. Pattanayak, S. Verma, V. Kalyanaraman, *et al.*, “Blockchain technology: Beyond bitcoin”, *Applied Innovation*, vol. 2, no. 6-10, p. 71, 2016.
- [22] Y. Sovbetov, “Factors influencing cryptocurrency prices: Evidence from bitcoin, ethereum, dash, litcoin, and monero”, *Journal of Economics and Financial Analysis*, vol. 2, no. 2, pp. 1–27, 2018.
- [23] K. Wüst and A. Gervais, “Do you need a blockchain?”, in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, IEEE, 2018, pp. 45–54.
- [24] S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, “Performance analysis of private blockchain platforms in varying workloads”, in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, IEEE, 2017, pp. 1–6.
- [25] J. Eberhardt and S. Tai, “On or off the blockchain? insights on off-chaining computation and data”, in *European Conference on Service-Oriented and Cloud Computing*, Springer, 2017, pp. 3–15.
- [26] O. Alphand, M. Amoretti, T. Claeys, S. Dall’Asta, A. Duda, G. Ferrari, F. Rousseau, B. Tourancheau, L. Veltri, and F. Zanichelli, “Iotchain: A blockchain security architecture for the internet of things”, in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2018, pp. 1–6.
- [27] A. M. Antonopoulos, *Mastering Bitcoin: Programming the open blockchain.* ” O’Reilly Media, Inc.”, 2017.

- [28] J. B. Earp and D. Baumer, “Innovative web use to learn about consumer behavior and online privacy”, *Communications of the ACM*, vol. 46, no. 4, pp. 81–83, 2003.
- [29] D. E. Robling Denning, *Cryptography and data security*. Addison-Wesley Longman Publishing Co., Inc., 1982.
- [30] M. Campbell-Verduyn, *Bitcoin and beyond: Cryptocurrencies, blockchains, and global governance*. Routledge, 2019.
- [31] C. S. Wright, “Bitcoin: A peer-to-peer electronic cash system”, *SSRN Electronic Journal*, 2008. DOI: 10.2139/ssrn.3440802.
- [32] R. Beck, “Beyond bitcoin: The rise of blockchain world”, *Computer*, vol. 51, no. 2, pp. 54–58, 2018. DOI: 10.1109/mc.2018.1451660.
- [33] S. Popov, O. Saa, and P. Finardi, “Equilibria in the tangle”, *Computers Industrial Engineering*, vol. 136, pp. 160–172, 2019. DOI: 10.1016/j.cie.2019.07.025.
- [34] L. Sharifi, F. Freitag, and L. Veiga, “Combing smart grid with community clouds: Next generation integrated service platform”, *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014. DOI: 10.1109/smartgridcomm.2014.7007685.
- [35] Q. Hong, J. Kraus, M. Lymbery, and F. Philo, “Conservative discretizations and parameter-robust preconditioners for biot and multiple-network flux-based poroelasticity models”, *Numerical Linear Algebra with Applications*, e2242, 2018.
- [36] D. López and B. Farooq, “A blockchain framework for smart mobility”, in *2018 IEEE International Smart Cities Conference (ISC2)*, Sep. 2018, pp. 1–7. DOI: 10.1109/ISC2.2018.8656927.
- [37] P. Zheng, Z. Zheng, W. Chen, J. Bian, and J. E. Yang, “Ethershare: Share information in jointcloud environment using blockchain-based smart contracts”, in *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, Apr. 2019, pp. 233–2335. DOI: 10.1109/SOSE.2019.00040.
- [38] J. Koo and Y. Kim, “Interoperability of device identification in heterogeneous iot platforms”, in *2017 13th International Computer Engineering Conference (ICENCO)*, Dec. 2017, pp. 26–29. DOI: 10.1109/ICENCO.2017.8289757.
- [39] G. Eder, “Digital transformation : Blockchain and land titles”, 2019.
- [40] M. Nygaard and E. M. Schmidt, *Transistion systems*, Feb. 2014. [Online]. Available: <https://www.cs.au.dk/~gerth/dADS1-12/daimi-fn64.pdf>.
- [41] R. Twesige, “Bitcoin a simple explanation of bitcoin and block chain technology january 2015 richard lee twesige”, Jan. 2015. DOI: 10.13140/2.1.1385.2486.
- [42] A. K. Fedorov, E. O. Kiktenko, and A. I. Lvovsky, *Quantum computers put blockchain security at risk*, Nov. 2018. [Online]. Available: <https://www.nature.com/articles/d41586-018-07449-z>.
- [43] B. Saltaformaggio and R. Bhatia, *Screen after previous screens: Spatial-temporal recreation of android app displays from memory images*, https://www.usenix.org/system/files/conference/usenixsecurity16/sec16_paper_saltaformaggio.pdf, (Accessed on 12/09/2019).

- [44] H. Wang, Z. Zheng, S. Xie, H. N. Dai, and X. Chen, “Blockchain challenges and opportunities: A survey”, *International Journal of Web and Grid Services*, vol. 14, no. 4, p. 352, 2018. DOI: 10.1504/ijwgs.2018.10016848.
- [45] I. Eyal and E. G. Sirer, “Majority is not enough: Bitcoin mining is vulnerable”, *Financial Cryptography and Data Security Lecture Notes in Computer Science*, pp. 436–454, 2014. DOI: 10.1007/978-3-662-45472-5_28.
- [46] M. M. M. d. Nascimento, “Blockchain: Uma nova abordagem sobre votação eletrônica”, 2018.
- [47] S. Yu, K. Lv, Z. Shao, Y. Guo, J. Zou, and B. Zhang, “A high performance blockchain platform for intelligent devices”, in *2018 1st IEEE international conference on hot information-centric networking (HotICN)*, IEEE, 2018, pp. 260–261.
- [48] P. Urien, “Blockchain iot (biot): A new direction for solving internet of things security and trust issues”, in *2018 3rd Cloudification of the Internet of Things (CIoT)*, Jul. 2018, pp. 1–4. DOI: 10.1109/CIOT.2018.8627112.
- [49] M. Singh, A. Singh, and S. Kim, “Blockchain: A game changer for securing iot data”, in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, Feb. 2018, pp. 51–55. DOI: 10.1109/WF-IoT.2018.8355182.
- [50] L. Carlozo, “What is blockchain?”, *Journal of Accountancy*, vol. 224, no. 1, p. 29, 2017.
- [51] S. Singh and N. Singh, “Blockchain: Future of financial and cyber security”, in *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, 2016, pp. 463–467.
- [52] G. Foroglou and A.-L. Tsilidou, “Further applications of the blockchain”, in *12th Student Conference on Managerial Science and Technology*, 2015.
- [53] A. Kiayias, A. Russell, B. David, and R. Oliynykov, “Ouroboros: A provably secure proof-of-stake blockchain protocol”, in *Annual International Cryptology Conference*, Springer, 2017, pp. 357–388.
- [54] A. B. Ayed, “A conceptual secure blockchain-based electronic voting system”, *International Journal of Network Security & Its Applications*, vol. 9, no. 3, pp. 01–09, 2017.
- [55] B. Koteska, E. Karafiloski, and A. Mishev, “Blockchain implementation quality challenges: A literature”, in *SQAMIA 2017: 6th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications*, 2017, pp. 11–13.
- [56] J. Lindman, V. K. Tuunainen, and M. Rossi, “Opportunities and risks of blockchain technologies—a research agenda”, 2017.
- [57] N. Nunes, G. Ambler, X. Foo, J. Naftalin, M. Widschwendter, and D. Jurkovic, “Use of iota simple rules for diagnosis of ovarian cancer: Meta-analysis”, *Ultrasound in Obstetrics & Gynecology*, vol. 44, no. 5, pp. 503–514, 2014.
- [58] Y. Takefuji and H. Szu, “Blockchain is vulnerable against classic database approach”, *MOJ App Bio Biomech*, vol. 2, no. 5, pp. 102–103, 2019.

- [59] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Y. Dong, “Distributed blockchain-based data protection framework for modern power systems against cyber attacks”, *IEEE Transactions on Smart Grid*, vol. 10, no. 3, pp. 3162–3173, 2018.
- [60] D. Anton and M. McFall, *Uniqueness and auditing of a data resource through an immutable record of transactions in a hash history*, US Patent App. 15/230,422, Nov. 2016.
- [61] V. L. Lemieux, “Trusting records: Is blockchain technology the answer?”, *Records Management Journal*, vol. 26, no. 2, pp. 110–139, 2016.
- [62] M. Liu, F. R. Yu, Y. Teng, V. C. Leung, and M. Song, “Joint computation offloading and content caching for wireless blockchain networks”, in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, IEEE, 2018, pp. 517–522.
- [63] A. Baliga, “Understanding blockchain consensus models”, in *Persistent*, 2017.
- [64] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, “A review on consensus algorithm of blockchain”, in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2017, pp. 2567–2572.
- [65] K. Park and H. Lee, “On the effectiveness of route-based packet filtering for distributed dos attack prevention in power-law internets”, in *ACM SIGCOMM computer communication review*, ACM, vol. 31, 2001, pp. 15–26.
- [66] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, “Denial-of-service attack-detection techniques”, *IEEE Internet computing*, vol. 10, no. 1, pp. 82–89, 2006.
- [67] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, “On the security and performance of proof of work blockchains”, in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, ACM, 2016, pp. 3–16.
- [68] I. Bentov, A. Gabizon, and A. Mizrahi, “Cryptocurrencies without proof of work”, in *International Conference on Financial Cryptography and Data Security*, Springer, 2016, pp. 142–157.
- [69] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, “Controlling data in the cloud: Outsourcing computation without outsourcing control”, in *Proceedings of the 2009 ACM workshop on Cloud computing security*, ACM, 2009, pp. 85–90.
- [70] M. Castro, B. Liskov, *et al.*, “Practical byzantine fault tolerance”, in *OSDI*, vol. 99, 1999, pp. 173–186.
- [71] S. Ølnes, J. Ubacht, and M. Janssen, *Blockchain in government: Benefits and implications of distributed ledger technology for information sharing*, 2017.
- [72] S. Huckle, R. Bhattacharya, M. White, and N. Beloff, “Internet of things, blockchain and shared economy applications”, *Procedia computer science*, vol. 98, pp. 461–466, 2016.
- [73] V. Buterin *et al.*, “A next-generation smart contract and decentralized application platform”, *white paper*, vol. 3, p. 37, 2014.

- [74] T. Yang, Q. Guo, X. Tai, H. Sun, B. Zhang, W. Zhao, and C. Lin, “Applying blockchain technology to decentralized operation in future energy internet”, in *2017 IEEE Conference on Energy Internet and Energy System Integration (EI2)*, IEEE, 2017, pp. 1–5.
- [75] P. De Filippi, “The interplay between decentralization and privacy: The case of blockchain technologies”, *Journal of Peer Production, Issue*, no. 7, 2016.
- [76] X. Liu, W. Wang, D. Niyato, N. Zhao, and P. Wang, “Evolutionary game for mining pool selection in blockchain networks”, *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 760–763, 2018.
- [77] R. Qin, Y. Yuan, and F.-Y. Wang, “Research on the selection strategies of blockchain mining pools”, *IEEE Transactions on Computational Social Systems*, vol. 5, no. 3, pp. 748–757, 2018.
- [78] D. Dolev and A. Yao, “On the security of public key protocols”, *IEEE Transactions on information theory*, vol. 29, no. 2, pp. 198–208, 1983.
- [79] S. Delgado-Segura, C. Pérez-Sola, G. Navarro-Arribas, and J. Herrera-Joancomartí, “Analysis of the bitcoin utxo set”, in *International Conference on Financial Cryptography and Data Security*, Springer, 2018, pp. 78–91.
- [80] H. J. Yoon, J. H. Cheon, and J. H. Sohn, *Private key generation apparatus and method, and storage media storing programs for executing the methods*, US Patent 9,036,818, May 2015.
- [81] M. Bellare and B. Yee, “Forward-security in private-key cryptography”, in *Cryptographers’ Track at the RSA Conference*, Springer, 2003, pp. 1–18.