CHAPTER 1

INTRODUCTION

Securely conductance of an election is a subject of National security in every democracy. There are only two types of systems that allow people to cast their votes for the polls. The polling systems are either electronic-based or paper-based. The paper ballot process is very time consuming and also counting of the paper ballots is prone to errors, and many parties don't believe that the electronic voting machines are hackproof. The hazards of the electronic polling are so much that many governments avoid it from the use. Electronic Voting machines have been seen as a flaw by the security agencies because of the threats due to physical access to the devices. Anyone having physical access to such a machine can hack the tool which finally affects all the votes cast on that machine. Thus, replacing the existing system with a new voting system is crucial to restrict fraud and having the polls process verifiable and traceable.

To minimize the cost of the method of the national election and to enhance the security circumstances of an election, the computer security field has studied various possibilities of electronic voting system for a decade. If any tampering with the polling device occurs, the potential expenses are far unfavorable. As we know that our current voting systems, whether they are electronic or paper-based, include deficient levels of transparency which becomes very challenging and unendurable for electors to guarantee that the calculation of the votes done by the election commission is accurate. As an example, Brazilian polls used Direct Recording Electronic polling system which does not give any information regarding the results of the poll and statistics related to it, aside from the counting of the votes. Further, the balloting system used by the Virginia government lead to several security-related questions which results in the discontinuation of the polls by the Virginia Information Technologies Agency. This signifies that if recounting of the ballots is required, then it is done by government delegates only which results decrease the confidence of the voter in the election results. However, data from the DRE System can be traced back for re-examining the links of the tickets with the electors. But this would create crucial concerns about the confidentiality of the polls which is not appropriate in a Republican ballot.

We can also consider creating the voting app as a Web App, instead of DApp. But there are some problems of designing voting app as a Web App.

(i) Votes can change.

(ii) Election Rules can vary.

The assurance, confidentiality, and transparency predicaments of an online voting scheme can now be inscribed by adopting an emerging and latest technology which is called blockchain. Blockchain technology owns very encouraging potentials. Let's take an example to demonstrate how a blockchain system works to deliver a secure and safe event without the need for a central authority.

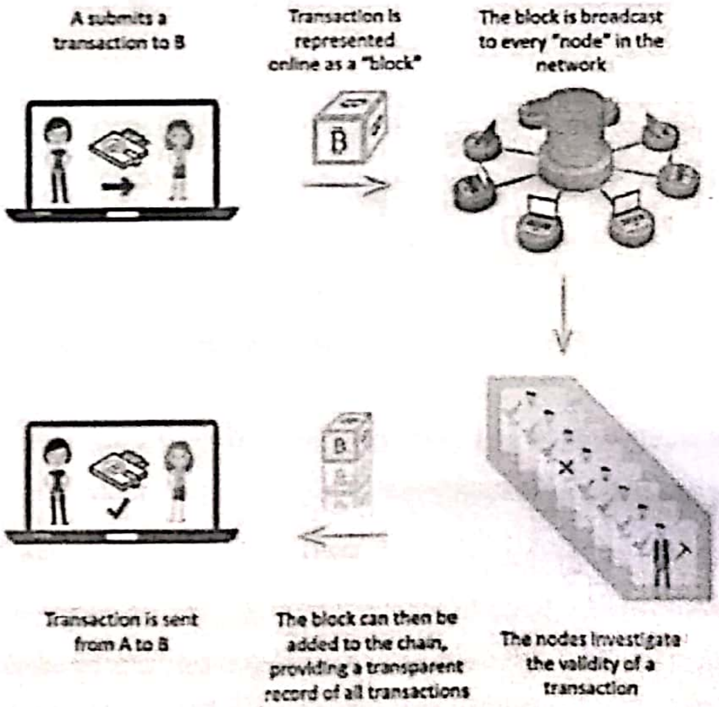

A submits a transaction to B
Transaction is represented online as a "block"
The block is broadcast to every "node" in the network
B
The nodes investigate the validity of a transaction
The block can then be added to the chain, providing a transparent record of all transactions
Transaction is sent from A to B

Figure 1-1 The schematic of blockchain network

Consider the case of Figure 1. A wants to send some money to B. The transaction in the network is served as a block. After A has submitted the block, it is transferred to each node present in the system which is known as a miner. Once the majority of the miners have approved the block, the event can be treated as completed and authorized. This mechanism provides transparency and reliability for transmitting and accepting transactions. The same idea of blockchain can also be extended to similar concepts such as elections.

An unidentified person named Satoshi Nakamoto[1] introduced blockchain technology. The main aim of the blockchain at the time of introduction was to develop a payment platform which allows transaction of money in the network without the need of any central authority. "Craig Steven Wright," an Australian businessman and computer scientist, openly declared to be the central element of the organization that was responsible for forming the blockchain.

A blockchain system is an open source system, which is impervious to any information adjustments. Since a blockchain network is stringently straightforward, transparent, and conveyed just as an agreement based the protected and one of a kind structure of this framework shows that meddling is outlandish when properly executed. The usage of surveying with the help of blockchain systems[15] has accomplished progressively unprecedented acknowledgment, after, the 2016 US presidential surveys in which the electronic casting ballot frameworks were viewed as interceded with by remote programmers. For instance, President Obama's emphatically needs to reject the 35 Russian representatives of the US because of the issues of Russia's altering the 2016 decision, finishes up the helplessness of the surveying frameworks to outside altering. A balloting framework which is fitted with blockchain innovation gives an important dimension of clearness by continuing an opened vault of votes while protecting the mystery of the voters. The accord from adjacent every one of the hubs is mandatory for an exchange to gets acknowledged in square chain innovation. This makes the balloting gadget a widely, reliable stage.

Using blockchain for decisions is worth something beyond a trial, nonetheless. Portable casting a ballot utilizing a sheltered and tried interface could dispense with voter extortion and lift turnout. It will make it increasingly helpful for natives to cast a ballot while abroad, regardless of the separation and time. It is likewise an advantageous apparatus for the race commission to keep up straightforwardness in the constituent procedure, limit the expense of leading decisions, streamline the way toward checking cast a ballot and guarantee that all votes are tallied.

Under the innovation that was utilized in the West Virginia decisions, a voter's personality is checked utilizing biometric apparatuses like a thumbprint filter before casting a ballot on a cell phone. Each vote structures some portion of a chain of votes, where it is numerically demonstrated by the outsider member. Utilizing blockchain, all information of the race procedure can be recorded on a freely evident record while keeping up the obscurity of voters, with results accessible in a split second.

Potential mediations become all the more challenging exponentially to deal with, when a satisfactorily high number of hubs are executed, if not completely unthinkable. In the accompanying, we intend to survey the previous works that have been directed on blockchain-based e-casting a ballot system.

CHAPTER 2

BACKGROUND AND MOTIVATION

In this chapter, we are going to discuss how blockchain technology works. We will take bitcoin as an example while considering it. We also discuss other important concepts of blockchain technology. We also throw light on the idea of smart contracts.

2.1 Blockchain

A blockchain is an immutable, distributed, public, indisputable ledger. This brand-new technology operates over four main features:

(i) One ledger exists in various locations: The same ledger is available at many places at the same time. So, maintenance of the distributed ledger is effortless, and there is no single point failure.

(ii) There is no central authority. So anyone can append a new block or transaction to the ledger.

(iii) For a proposed block to be a part of the permanent ledger, the majority of the nodes in the network must reach a consensus.

(iv) Any "new block" that is added to the ledger must have reference to the previous block of the ledger, building a changeless chain and thus limiting tampering with the integrity of prior entries.

A blockchain is a database which is distributed in nature. It records all the transactions as a ledger format in the form of blocks. Because of the decentralized nature of the blockchain, each peer in the network has a replicated database. In other words, we can also say that blockchain is a data structure which constrains how data is stored and constructed. Different types of data structures are CSV, images, text files, databases, and others.

The blockchain is an organized list of blocks. Each block has its identification and can be identified using a hash. Each block refers to the block that appeared before it. This results in the series of blocks. The blockchain is immutable. This means that if a block is formed and added to the blockchain, the transaction in the associated block cannot be reverted or changed. We can also say that the blockchain should be write-only and updating and deletion of the blockchain is not allowed for anyone. This is to prevent the double spending problem and also to ensure the integrity of the data. Blockchain can also be referred to as DLT(Distributed Ledger Technology). If someone wants to alter the chain, then the time is taken by the culprit to modify the network is much more than the time taken to recognize the attack. So, considering this we can say that blockchain is immutable or non-alterable. In a blockchain, the users of the blockchain network can interact with the other users of the blockchain anonymously without revealing their identities.

There are two kinds of networks, i.e., peer-to-peer(P2P) and server-based network. A peer-to-peer network(Fig 2-1(b)) is also known as a decentralized network whereas server-based network(Fig 2-1(a)) known as a centralized network. In P2P network data is distributed among all the nodes, whereas in the centralized network data resides on a central server. In a peer-to-peer network, all the nodes have equal rights to take part in the network. Peer to peer network, functionalities, and responsibilities divides among peers. The server-based network works directly opposite to do the peer-to-peer network, which means that on server-based network data is completely kept on servers, and only some people have access to that data. Nowadays, most of the applications or websites use a centralized architecture.

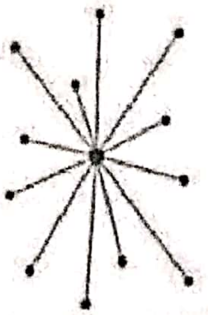


Fig 2-1(a) Centralized architecture

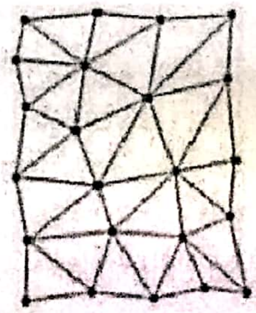


Fig 2-1(b) Peer-to-peer(Decentralized) architecture

Because all the information is placed on a single server, the data is vulnerable to attacks or failures. But in a peer-to-peer network, each peer in the network have access to all the data.

When there is an update in the data, the update is shared to all the peers in the network. The data in the peer-to-peer network is duplicated many times in the network. This is why this type of is considered less efficient than the traditional network. Even though each node has a copy of data in the network, but they are independent of each other. Because of the no central authority, the peer-to-peer networks are more robust than the traditional server-based networks.

Fig 2-2 shows the structure of the traditional network

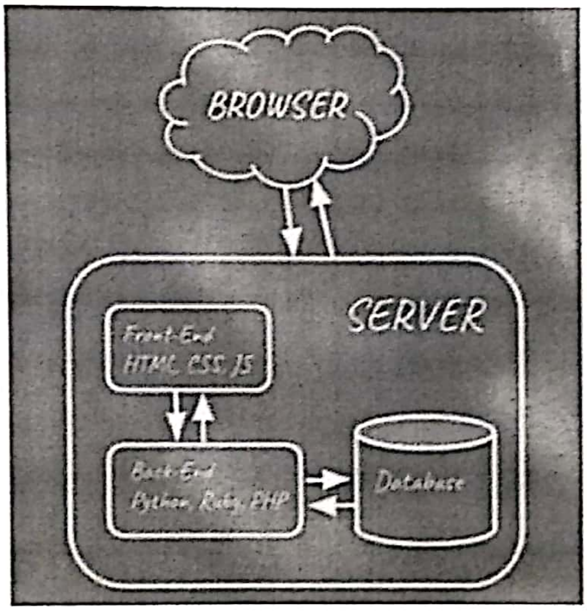


Fig 2-2 Schematic of Server-based betwork

2.1.1 History and Applications

Recently, Blockchain technology because of cryptocurrency has attracted massive attention from both academics and industry. Bitcoin is known as the first cryptocurrency. Bitcoin is recognized as the first publicly used application of blockchain technology. Bitcoin facilitates transactions between the users or nodes of the network without the need of any central authority. Verification of the transactions in the bitcoin network is done by an extensive network of nodes known as "miners." These transactions are written in a distributed ledger, which is publically available. Blocks are used to keep track of each transaction ever happened in the network. From the beginning, all the transactions of the system are being recorded. This is the reason that the ledger of the bitcoin is constantly growing.

The typical structure of the blockchain system(bitcoin) is shown in Fig 2-3

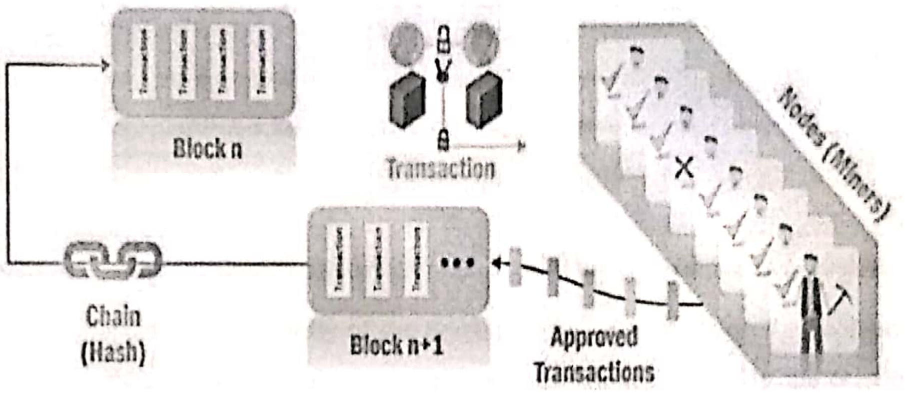
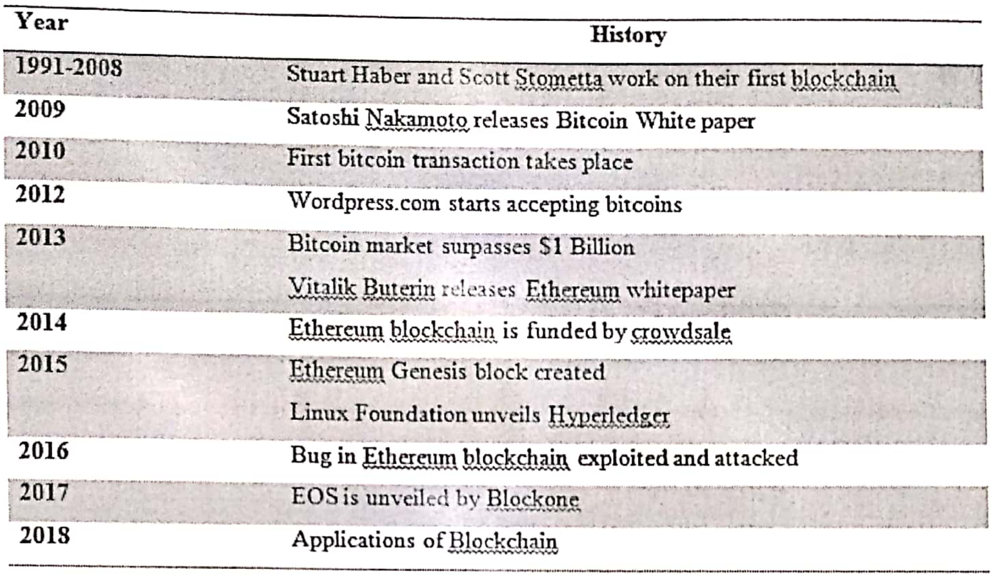


Fig 2-3 Structure of bitcoin blockchain

The bitcoin distributed ledger is an open source database in which you do not need permission from anyone to write things. Therefore, it is not necessary to log in or register for users to access. The expansion is finished by running an open source programming through which a PC associates with different PCs on the system employing the Web. The product makes it conceivable to send or get exchanges, or to add information to the overall record by taking care of a perplexing estimation issue that is hard to confirm. This is broadly known as "mining". It is critical to remember that numerical problems become a test when utilizing capacities called "hash."



Year	History
1991-2008	Stuart Haber and Scott Stometta work on their first blockchain
2009	Satoshi Nakamoto releases Bitcoin White paper
2010	First bitcoin transaction takes place
2012	Wordpress.com starts accepting bitcoins
2013	Bitcoin market surpasses $1 Billion Vitalik Buterin releases Ethereum whitepaper
2014	Ethereum blockchain is funded by crowdsale
2015	Ethereum Genesis block created Linux Foundation unveils Hyperledger
2016	Bug in Ethereum blockchain exploited and attacked
2017	EOS is unveiled by Blockone
2018	Applications of Blockchain

Table 1-1 History of blockchain

In Bitcoin, hash capacities are a piece of the cryptographic calculation used to compose new exchanges to the framework through the revelation procedure. Mining is an essential and fundamental piece of the blockchain that ensures fairness while keeping up the solidness and security of the system. Minors get a specific measure of bitcoins in return for their administrations. This makes an impetus stage to draw in more individuals to the mine. The more prominent the number of miners, the more the system can be made and verified. When assessing a Bitcoin record, one can without much of a stretch find which record has what number of Bitcoins and which Bitcoin record gets from whom. This dimension of straight forwardness is the thing that permits Bitcoin exchanges to be checked by anybody, anyplace on the planet. Subsequently, if somebody attempts to connect a fake transaction, the miners would effectively know and don't affirm it.

Further, the applications of blockchain technology can be divided into four classes as described in Fig 2-4.

(i) Digital Currency

(ii) Smart Contracts

(iii) Security

(iv) Record Keeping

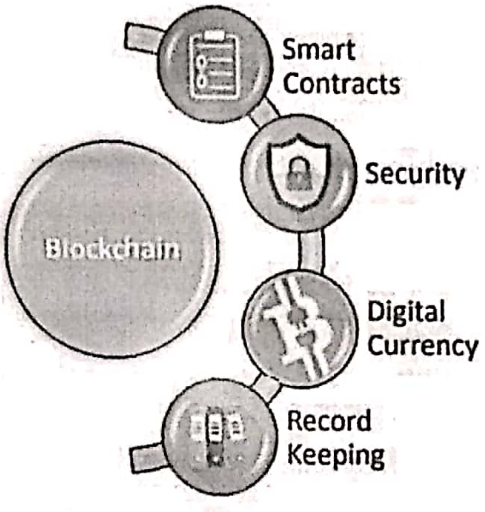


Fig 2-4 Categories of applications of blockchain

Based on the above categories, the e-voting system, which is based on blockchain technology, would fall under the "record keeping" category. Bitcoin and ether falls under the category of digital currency.

2.1.2 Types of Blockchain

Blockchain can be classified into 3 categories:

(i) Private Blockchain
(ii) Consortium Blockchain
(iii) Public Blockchain

In private blockchain, a particular user with specific permissions can join the network. It can send or write transactions in the system. Examples of private blockchain are Eris, Ripple. In a public blockchain network, any unidentified user can enter the publically available blockchain network, send a new transaction to the system, read the content of the blockchain network, and also check the exactitude of the blocks. Examples of public blockchain are Ethereum[17], Bitcoin. Consortium blockchain can be referred to as the hybrid blockchain which includes the characteristics of both the private blockchain and public blockchain. We can assume the meaning of these blockchain from their names itself.

Public Blockchains are the blockchains where anybody can take an interest from the world. Assume 1000 individuals are occupied with a public blockchain network; at that point, more individuals can join since it is public.

Consortium Blockchains are partially decentralized in nature. Assume out of 1000 individuals 100 individuals are associated with consortium blockchain and out of those 100 just 30 individuals are associated with a transaction then just those 30 needs to sign a transaction for its finish. It will be refreshed on records of each of the 100 people. An individual must be included upon the understanding of other 100.

Private blockchain, on the other hand, is private which means no outside substance can join that blockchain over that assume there are ten individuals in blockchain and they are offering items to one another then no other individual can know at what value items are being sold. Ex - A needs to provide an item to B, C, D. A can offer the same thing to B, C, D for 10, 20, 30 tokens individually, as it is private. At the point when an exchange between A to B will happen C, D will realize that a trade has occurred however won't become more acquainted with about subtleties of exchange. Hyperledger fabric is case of private or permissioned blockchain.

2.1.3 Structure of Block

Each block in the blockchain contains, in addition to other things, the present time, a record of a few or every ongoing exchange, and a reference to the block that came preceding it. It likewise contains a response to a hard to-unravel scientific riddle - the response to which is extraordinary to each square. New squares can't be submitted to the system without the right answer - the way toward "mining" is the way toward contending to be the alongside discovering the appropriate response that "tackles" the present square. The scientific issue in each square is amazingly hard to unravel; however, once a legal arrangement is discovered, it is simple for the remainder of the system to affirm that the arrangement is right. There are several legitimate answers for some random block just one of the arrangements should be observed for the square to be settled.

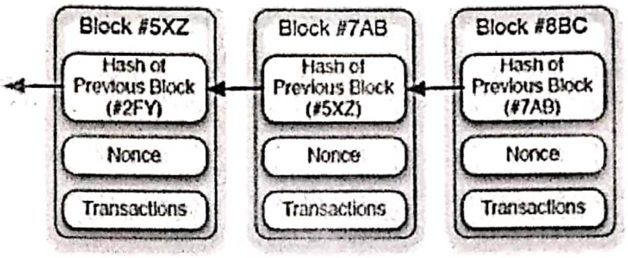

Block #5XZ
Hash of Previous Block (#2FY)
Nonce
Transactions
Block #7AB
Hash of Previous Block (#5XZ)
Nonce
Transactions
Block #8BC
Hash of Previous Block (#7AB)
Nonce
Transactions

Fig 2-5 Structure of a block in blockchain

2.1.4 Consensus Mechanisms

Consensus mechanisms are the etiquettes that ensure all the nodes in a blockchain are synchronized with each other and agree on transaction legitimacy. Some of the consensus mechanisms are explained below.

Proof of Work

Proof of work is one of the most crucial parts while checking the authenticity of a newly generated block. The bitcoin blockchain uses the proof of work named Hashcash. It is just a piece of code but which is very difficult to create and easy for others to confirm. For anyone to tamper a blockchain, they have to change all the succeeded blocks starting from the tapered block. For limiting this, the bitcoin limits the time for the creation of a new block in the chain to 10 minutes. For the acceptance of any new block in the chain, miners must have to complete the proof of work.

Proof of stake

Proof of stake is an alternative approach to the proof work. Proof of stake also provide consensus and try to avoid the problems of selfish mining and double spending. In proof of stake, the resources of the miners are also taken into account. This means that mining of proof of stake blocks is directly proportional to the resources of the miners. If a particular miner holds 5% of the bitcoin, then he can mine only 5% of the proof of stake blocks.

Proof of Burn

Proof of burn is an alternative to both the proof of work and proof of stake. The idea of the proof of burn is that miners must give proof of the coins burnt. This seems expensive, but the burnt assets only are used for the verification.

Proof of Capacity

Due to high energy consumption in proof of work, proof of capacity appeared as one of the alternative solutions to the mining. Proof of capacity gives miners freedom to use the free space available on their hard disks to mine the crypto coin.

There are two steps included in the proof of capacity.

(i) Plotting

(ii) Mining

Advantage of using proof of capacity over others is that it is very efficient as compared to others. Example of the cryptocurrency using proof of capacity as its mechanism is Burstcoin.

Proof of Importance

Proof of importance was proposed by NEM(New Economic Movement). Proof of Importance is a consensus mechanism which generally used to identify whether the nodes are eligible to add the block to the chain or not.

It overcomes the problem of proof of stake in which "the rich get richer." Proof of stake supports the idea of giving a chance to those who already have a higher stake of cryptocurrency. But in proof of importance, it is taken into account which account is supporting the network.

The three important factors are

(i) Vesting

(ii) Transaction Partners

(iii) Number and size of transactions in the last 30 days

There is one more problem of the blockchain, which is known as "fork." The problem is that sometime because of the latency in the network, miners added more than one block to the chain. Now, in this situation, the system will automatically detect the valid block with the help of consensus rule "longest chain."

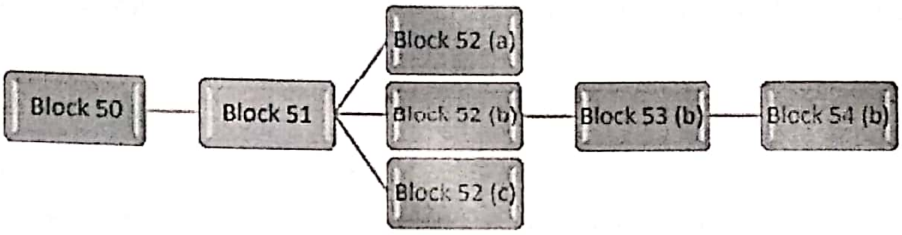


Fig 2-6 Structure of blockchain tree, which represents acceptance of longest chain

Let's take an example of Fig 2-6. In this example, we are assuming that all the miners are synchronized at block 51. After that, the problem of fork arises. Now, there are three different blocks in the network i.e., block 52(a), block 52(b), block52(c). These blocks appear because of network latency. These blocks are somewhat different from each other. In this case, block 52(b) is considered valid because of the "longest chain" consensus rule followed as we can see the block 52(b) is followed by block 53(b) then block 54(b).

2.1.5 Challenges of Blockchain Technology

2.1.5.1 Immutable bugs

As we know that the blocks in any blockchain once created cannot be altered. Same goes for the smart contract. Once the smart contract is formed, it is finalized and cannot be modified. If there is a flaw in the smart contract, one cannot fix it instantly. We need to update it with a proper mechanism followed by blockchain.

2.1.5.2 Lack of Regulations and Standardization

There is no standardization for blockchain technology and not even for smart contracts. So one doesn't know about the best methods for smart contracts. Also, the lack of regulations creates a lot of problem for the government regulations agencies. This becomes the primary reason for the banning of cryptocurrencies by India. Many illegal settlements can happen with the help of smart contracts. So it's a high-security risk which needs to be addressed by proper regulation protocols.

Juels et al. [27] introduced the idea of criminal smart contracts (CSCs). Some typical CSCs are various real-world crimes, leakage of confidential information, and theft of cryptographic keys.

2.1.5.3 Eclipse Attack

Eclipse attack is another misleading action in blockchains in which an untrustworthy hub(node) assumes responsibility for the sufferer's internal and outward associations, henceforth isolating the sufferer from the remainder of the hubs in the system. The intruder would then be able to obstruct the sufferer's permeability of the system, and commit them to spend their computing power on review an out-of-date variant of the blockchain arrange, or far and away more terrible occupy the ability to the upside of his/her unjust exercises. Other than hindering and harming the uprightness of the blockchain organize, eclipse attacks could be the beginning of and heighten other potential assaults, for example, selfish mining.

2.1.5.4 Double Spending

Double spending is one of the most common problems of the blockchain. In digital currency, double spending means a certain number of coins are being used in more than one transactions. If we consider the usual scenario, then we can say that double spending is when we spend rupees 10 in two places. This means we are paying the same 10 rupees instead of two different 10 rupees. This can also be possible while voting. A single ballot of a voter can be used to vote for more than once.

2.1.5.5 Selfish Mining

Selfish mining happens when a group of deceptive excavators conspires to expand their mining reward income. In such a situation, excavators can gain more reward by hiding the recently delivered squares from the center chain and making an unmistakable fork. To more readily comprehend selfish mining, how about we investigate Bitcoin blockchains. Bitcoin mining works dependent on a gathering of diggers who disentangle cryptographically entangled issues and get boosted, for example, get advanced coins, in return. Such salary relies upon various factors, for example, the dimension of the trouble of the cryptographical issue, mining cost, web speed, and association quality.

Bitcoin is masterminded in a manner to boost diggers corresponding to their mining yield. With such system set up, regardless of whether large gatherings of excavators endeavor to conspire, they can't get a more significant number of mint pieces joined than what they exclusively and on the whole created in the open record. In any case, if exploitative hubs disguise the new squares and make them accessible just in their private system, they can rise a lot of the system's general reward. Selfish mining[16] is such a significant issue, that can even risk the decentralization idea of blockchains, causing the centralization of the blockchain activities. In contrast to twofold spending, selfish mining ought to be painstakingly treated with regards to blockchain-empowered casting a ballot.

Selfish-mining assaults could effectively affect the trustworthiness of the blockchain framework. Whenever fruitful, exploitative enemies can without much of a stretch transform into more productive hubs than the fair hubs. Benefits from selfish mining can rise if the foes use increasingly computational power. This can make the assaults exponentially increasingly viable, until to a point where over half of the power in the system is held for the aggressors. This can happen at extreme power ordinary hubs out of the system. In such a case, the deceptive segment of the system would be not just ready to assemble all the square rewards, yet additionally to hinder any tally from being tallied decently.

2.2 SMART CONTRACT

The phrase "smart contract" was minted by Nick Szabo, around 1993. Szabo represented a smart contract as “a set of promises, specified in the digital form, including protocols within which the parties perform on these promises [2].” One of his most notable examples was to associate the smart contract with the vending machine. A blockchain-enabled smart contract is just a piece of code which is self-executing when the parties involved in the agreement agrees[1].

For writing a smart contract, we have two different languages known as “Solidity” and "Serpent." But Solidity is very much popular and widely used. Smart contract eliminates the use of a middleman which results in reducing the cost. It involves only those parties that are part of the agreement. A blockchain-based smart contract is a computer etiquette designed to digitally verify, facilitate, or implement the agreement or performance of a contract. Smart Contracts eliminates the role of intermediaries which results in various benefits for the parties involved in the smart contract. Smart Contracts have come a very long way. From the '90s to now, the original idea of the Szabo has been revived with the arrival of blockchain technology.

2.2.1 Working of Smart Contract

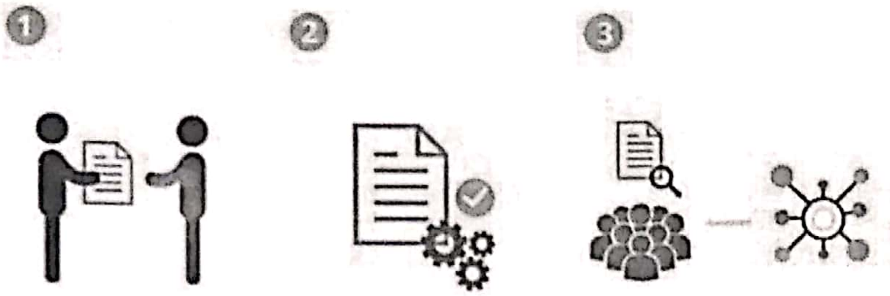


Fig 2-7 Working of smart contract

The working of a smart contract is divided into three necessary steps as described in Fig 2-7.

The first step is to write the option contact between the parties in the form of code into the publically available blockchain. The identity of the individuals available in the contract is not known to each other, but the contact is a public ledger.

The second step in the working of the smart contract is a triggering event. Triggering event includes an expiration date and delivery of the belongings. The smart contract executes itself according to the coded terms of the contract.

In the third step, assets are released to the participating parties. For further understanding, regulators can study the immutable transaction ledger to understand all the activities that have taken place.

2.2 Challenges and Issues of Smart Contracts

As smart contracts are progressing at a brisk rate, it currently faces a lot of difficulties and problems. Based on the research, this section will illustrate the challenges of smart contracts.

2.2.1 Contract Adoption Challenges

Adoption Curve

The adoption of smart contract for the businesses is a massive task. They have to transform the whole business from the regular transaction system to the entire new blockchain enabled smart contract system. As security issues and many vulnerabilities are present, the adoption rate among the businesses is meager[18]. We can also say that the adoption rate is directly proportional to the learning rate.

Learning Curve

As we know that blockchain is new to the market, there are not many kinds of research present right now[18]. So the learning rate is not that much. Lack of standardization and regulations are also a significant cause for the low learning rate.

Data Privacy

When we talk about blockchain, we know that it's a decentralized database which is transparent. But when we talk about smart contracts, it may be possible some parties want to hide the deal from others. But by using smart contracts, it is not possible till now. Research is going on over it.

2.2.2 Smart contract vulnerabilities

Callstack Size

As the maximum size of the callstack is restricted to 1024 frames to the Ethereum. When a contract requests another contract, then the associated call stack is incremented by one frame. When the size of the call stack reached to its limit, then an adversary throws an exception. If the thrown exception is not controlled correctly, then the odds of an attack on adversary are quite high.

Mishandled exception

When a caller contract calls a callee contract if somehow callee contract runs abnormally and terminates and return false to the caller. The caller must have to verify the transaction has been happened successfully or not because the linked exception of the callee may or may not be transferred to the caller. If the caller contract doesn't confirm the operation, it will bring potential threats. Luu et al.[19] highlighted mishandled exception which results in providing importance to the return value checking. To detect this type of vulnerability, 'OYENTE' tool[19] can be used.

TOD(Transaction Ordering Dependence)

When multiple dependent transactions call the same contract, then TOD occurs in which miner can form the order in which transaction will occur by manipulating it. To handle this issue mentioned above, Natoli et al.[20] recommended the adoption of Ethereum-based functions to implement the ordering of the transactions. Luu et al.[19] developed a tool termed 'OYENTE' which can be used to identify the smart contracts that are vulnerable to TOD.

Re-Entrancy issue

In this issue, a caller can go into infinite loops state issuing the recurring transactions occurred and conclusively empty the bank balance. This is caused when a caller contract calls a callee contract, and the intruder may use the in-between position of the caller contract to lead repetitious calls. In June 2016, an intruder used this issue to steal around 60 million US dollars. 'OYENTE,' developed by Luu et al.[19], can be used to detect this kind of vulnerability.

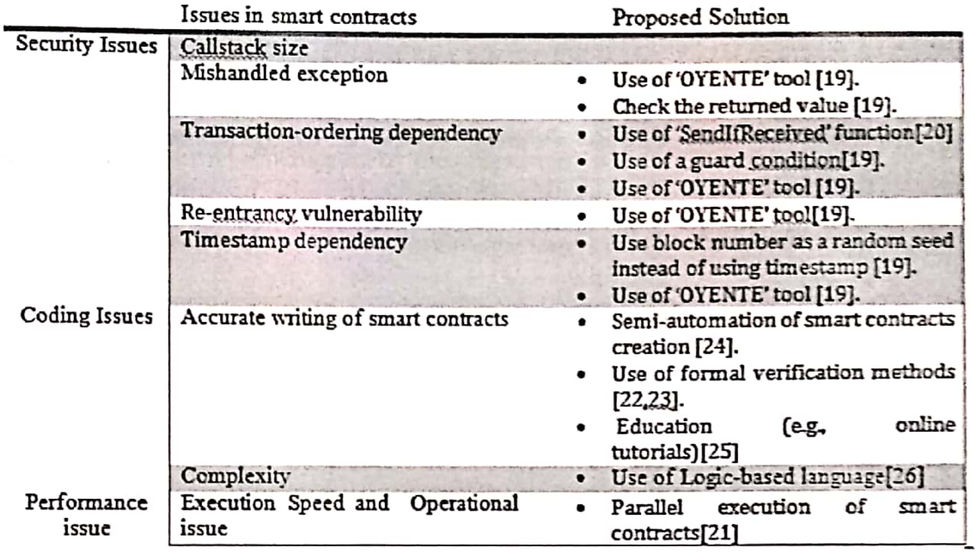
Timestamp Dependent

The miners have the power to alter the timestamp with a few timestamps. This may cause a problem to those smart contracts which take the timestamp as a triggered condition. Luu et al.[19] suggested the use of block number as a random seed for smart contracts rather than accepting the block timestamp. 'OYENTE' tool introduced in [19] can be used to recognize this vulnerability in smart contracts.

2.2.3 Performance issue:

Execution Speed and Operational issue

One of the major problem in the performance of blockchain-enabled smart contracts is the serial execution which degrades the performance of the system and limits the performance up to a certain extent. As businesses are adopting smart contracts, the number of smart contracts will increase day by day. Because of the sequential execution, it will not be able to finish the execution on time which will ultimately make the smart contract worthless. Vukolic[21] proposed parallel execution of the smart contracts instead of sequential execution ehich results in the improvement in the performance.



	Issues in smart contracts	Proposed Solution
Security Issues	Callstack size	
	Mishandled exception	• Use of 'OYENTE' tool [19]. • Check the returned value [19].
	Transaction-ordering dependency	• Use of 'SendIfReceived' function[20] • Use of a guard condition[19]. • Use of 'OYENTE' tool [19].
	Re-entrancy vulnerability	• Use of 'OYENTE' tool[19].
	Timestamp dependency	• Use block number as a random seed instead of using timestamp [19]. • Use of 'OYENTE' tool [19].
Coding Issues	Accurate writing of smart contracts	• Semi-automation of smart contracts creation [24]. • Use of formal verification methods [22,23]. • Education (e.g. online tutorials)[25]
	Complexity	• Use of Logic-based language[26]
Performance issue	Execution Speed and Operational issue	• Parallel execution of smart contracts[21]

Table 2-1 Issues in smart conracts with proposed solutions

2.2.4 Coding Issues

Accurate writing of smart contracts

As we have already discussed that a smart contract contains some agreement. The agreement can be in the form of policy or in the way of some transaction. So, accurate coding of the smart contract is critical because if the contract is not executed as the way it proposed, it may result in some loss. To handle this issue, semi-automation is prosposed[24]. The second solution is the selection of formal affirmation techniques to recognize the unidentified behavior of smart contracts[22,23]. The last resolution is to provide developers with guidelines which helps them to write a correct blockchain-enabled smart contract. Delmolino et al.[25] published online materials for this purpose.

Complexity

Complexity is also a significant issue while writing a smart contract. For writing a smart contract, we use Solidity. Solidity is a procedural language which makes the writing of smart contract a very hectic task because the developer has to mention each and everything while writing the code and also have to state how to do it. Idelberger et al. [26] suggested the use of logic-based language rather than using procedural language. Logic-based language can also reduce the complexity of writing of smart contracts. Also, there is no need to specify the sequential order for the smart contract by the programmers.

CHAPTER 3

RELATED WORK

In this particular section of the report, we are going to discuss the previous proposed methods by other authors by reviewing their previous papers.

3.1 Related Work Reviewed

Daniel[13] was the first to propose the use of blockchain technology for an online voting system in 2015.

In 2016, the idea of adopting the blockchain-enabled e-voting earned more attraction during the presidential election of the US. This happened just after Sep 2016, when the director of the FBI informed the House Judiciary Committee about the investigation of the Russian hackers who were endeavoring to intervene in the 2016 polls. He also told that FBI investigator there were multiple attempts by the hackers to hack the elector enrollment database[12].

Ryan Osgood[11], in 2016, explained the design of the blockchain and its advantages. He also talked about the challenges of adoption and progress of the system.

In 2017, Kartik Hegadekatti [10] drafted the idea underlining balloting on the blockchain and interpreted the benefits of such a system. He also examined the consequences of polling through the blockchain technology.

In Jan 2017, Ivo Kubjas[9] explained the use of blockchain technology to make the web protocols for voting more secure.

Ahmed Ben Ayed[8], in 2017, proposed an e-voting system design by utilizing the nature of the blockchain, i.e., open source for making voting more reliable, secure and anonymous which results in the trust gaining of people in their governments and also to increase the number of voters.

Moura and Gomes[7] tried to resolve the issues regarding the confidence and transparency associated with the elections with the use of blockchain.

Bartolucci et al.[6], in 2017, proposed a new kind of system for the implementation of the fair and secure elections. They called it SHARVOT protocol, which introduced a secret share-based polling system. Their protocol also used Circle Shuffle, a shuffling technique, used to de-link the voters from their votes.

Kaan Koç et al.[5], in 2017, executed and examined a smart contract based e-voting application for the Ethereum blockchain network, which uses the Solidity language to write the smart contract and Ethereum wallets. They also take care of those who don't have Ethereum wallets and also allow the Android platform for voting. In their recommended approach, users can submit their votes directly from their Ethereum wallets or through their Android device. After the election is over, the Ethereum blockchain network will take care of all the records of votes and ballots. The transactions of Ethereum wallets can be handled with the help of the consensus mechanism of the Ethereum network. This creates a trustworthy and transparent ecosystem for e-voting.

After that, Casado-Vara and Corchado [4] introduced a new voting system which uses the blockchain to minimize and prevent the imperfections of the balloting system. In his approach, blockchain is being used for transmits the votes digitally to a voting station. After that voting station transmits a smart contract to each elector and asks them to registers their vote on a chain. After this step, the side chain would be assigned to the standard balloting blockchain. To prevent any malicious activity, smart contract platform being used to conduct voting.

In the same year, Wang et al.[3], came with a new approach for e-voting on the blockchain platform, which is based on ring signature and homomorphic ElGamal encryption. The main features of this system are self-management, free-receipt, non-interactive, and decentralization. Further, with the help of one-time ring signature, the atomicity of the vote can be assured. Moreover, voting integrity can be confirmed with the verifiable public billboards.

Further, Akbari and Zhao et al.[14] examined many existing suggested methods of e-voting and drafted potential enhancements. They also discussed further obligations of web-based balloting as compared to the typical monetary transactions. They proposed a model which uses the parallel processing of the blockchain technology which further removes the scalability deficiency of the blockchain technology. This increases the overall speed of the blockchain. They also introduced the use of biometric authentication of the voters at the time of voting. Also, to secure the secrecy of the votes, a scheme was suggested.

Despite many papers published over the blockchain-enabled e-voting, the feasibility of this platform at a huge scale has yet to be explored.

CHAPTER 4

FROM BLOCKCHAIN TO VOTING DAPP

In this section, we discuss the requirements for our blockchain-based e-voting system. We also discuss the proposed method in this section. We also review what we are going to use for this project.

4.1 DApp

First of all, we will discuss what DApp is? DApp is an acronym for Decentralized application. There might not be one definition of DApp, but there are some standard features.

(i) Open Source

(ii) Decentralized

(iii) Incentivized

User and developer should decide all changes. It's code based should be accessible to scrutiny. All records of the application operations must be stored on a public and decentralized blockchain to encourage transparency, trust, and efficiency. Anyone that helps to secure the blockchain should be rewarded with cryptographic tokens.

The first DApp is bitcoin itself. It allows the transfer of money without the need for an authority to deem transactions valid. This lack of central authority makes bitcoin a decentralized application.

Both Bitcoin and Ethereum are DApps. But the ethereum can be think of a like a DApp library.Ethereum enable developers to write programs called smart contracts that are stored on the ethereum blockchain. These smart contracts then stored and executed across every node in the network making them decentralized apllications. Rather the needing to develop the entirely new blockchain for every application, Ethereum created a secure platform for DApps to be built and deployed.

Why should we use DApp?

(i) Trust(Open Source)

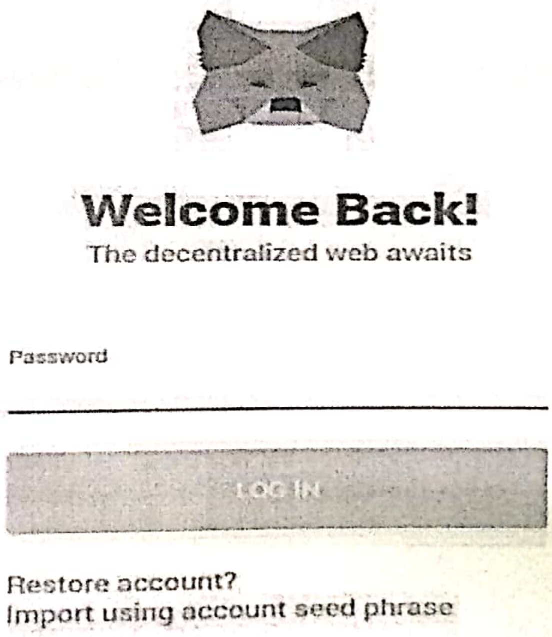
(ii) Guaranteed Execution

(iii) Censorship Resistance

4.2 Dependencies for our DApp

Metamask

We will require metamask extension for google chrome. So as to utilize blockchain, we should connect with it. We need to introduce an extraordinary program extension to have the option to utilize the ethereum blockchain network. We will most likely associate with our local ethereum network to connect with our own account and collaborate with our smart contract by utilizing metamask. Fig 4-1 shows the interface of the metamask login page.



Welcome Back!

The decentralized web awaits

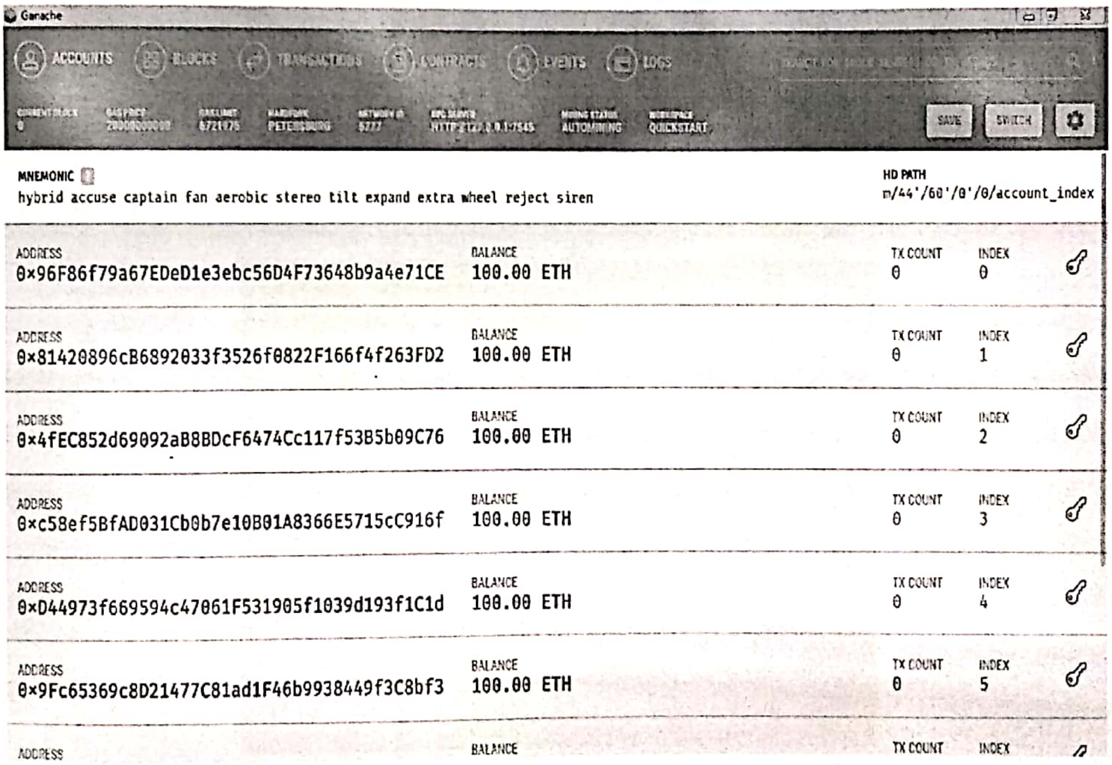
Restore account?
Import using account seed phrase

Fig 4-1 Metamask login interface

Fig shows the interface of the metamask login page. We can login to our account by entering password and connects to the blockchain. It also provides the information about the ethereum coins.

Ganache

Ganache is a local in-memory blockchain that we will use for development purposes. One can download ganache by going to the truffle framework website. Once we have ganacge started, we have a local blockchain running, and ganache gave us ten accounts whenever it started. Each of the accounts has a unique address and each account has been credited with 100 ether as shown below in Fig 4-2. These ethers are fake and they actually don't worth anything on the real ethereum blockchain. The account addresses are unique identifiers that are going to represent the voters in our project.



CURRENT BLOCK	GAS PRICE	GAS LIMIT	HARDFORK	NETWORK ID	RPC SERVER	MINING STATUS	WORKSPACE
0	20000000000	6721975	PETERSBURG	5777	HTTP://127.0.0.1:7545	AUTOMINING	QUICKSTART

MNEMONIC
hybrid accuse captain fan aerobic stereo tilt expand extra wheel reject siren

HD PATH
m/44'/60'/0'/0/account_index

ADDRESS	BALANCE	TX COUNT	INDEX
0x96F86f79a67EDeD1e3ebc56D4F73648b9a4e71CE	100.00 ETH	0	0
0x81420896cB6892033f3526f0822F166f4f263FD2	100.00 ETH	0	1
0x4fEC852d69092aB8BDcF6474Cc117f53B5b09C76	100.00 ETH	0	2
0xc58ef5BfAD031Cb0b7e10B01A8366E5715cC916f	100.00 ETH	0	3
0xD44973f669594c47061F531905f1039d193f1C1d	100.00 ETH	0	4
0x9Fc65369c8D21477C81ad1F46b9938449f3C8bf3	100.00 ETH	0	5
ADDRESS	BALANCE	TX COUNT	INDEX

Fig 4-2 Ethereum accounts in Ganache

Truffle

Truffle is a framework that's going to allow us to create decentralized applications on the ethereum network. It's going to give us a suite of tools that will enable us to write our smart contracts with the solidity programming language. It also provides us with a framework for testing our smart contracts, and it provides us with a set of tools to deploy our smart contracts to the blockchain. We can also develop our client-side application inside the truffle.

4.2 Structure of Dapp

In this section, we are going to discuss about the general structure of our decentralized application. In this application, there are three important parts.

(i) Front-End

(ii) Back-End

(iii) Browser

For the front-end, we are using a straightforward interface. We are going to use HTML, CSS, and JS for our front-end. At the back-end, we are going to use ethereum blockchain network for storing the data and also we will deploy our smart contracts here. We are going to use google chrome as our browser which will link both the front-end and the back-end.

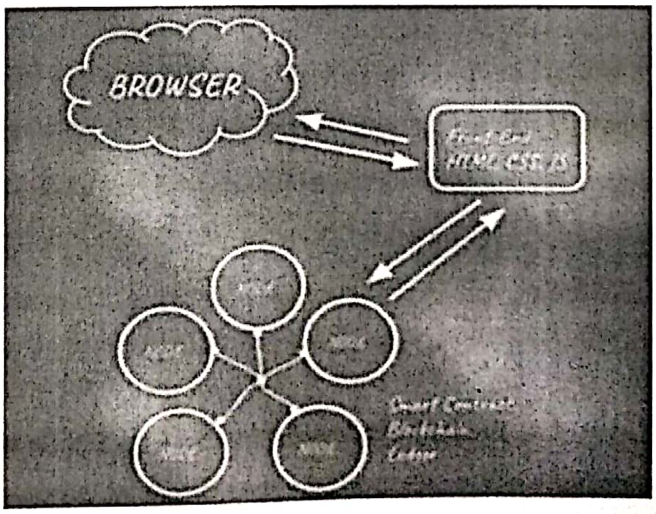


Fig 4-3 Structure of Decentralized App

Fig 4-3 defining the basic structure our decentralized application and also showing the interaction between the browser, front-end and back-end.

4.3 Proposed Methodology

In the previous methodologies, we have seen many versions of the blockchain-based e-voting app. Some use the fundamental blockchain technology, i.e., thinking of the votes instead of bitcoins. But the main problems of the fundamental blockchain is the chance of attacks high. We are going to propose a decentralized voting application which is based on ethereum blockchain technology with the use of smart contract in it. This is the improved version of the proposal which uses only a single smart contract for all the work. In our project, we are going to use a wide variety of smart contracts which will further reduce the chances of malicious attack and other flaws of the present voting system. The main purpose of the smart contract is to prevent malicious activities in the system.

Smart Contracts include:

(i) All the code of the DApp, that going read and write with the ethereum blockchain.

(ii) List all the candidates that run in the election.

(iii) Going to keep track everyone who voted in the election.

(iv) Writing of business rules which govern our election.

4.4 Working of our Blockchain

In our blockchain network, we are using smart contracts for keeping the chances of malicious activities low. We are using variety of smart contracts which will help in govern the rules for the election and many other things. Our blockchain network consists of Candidate's wallets, main blockchain, voter's wallet, sidechain. From the main blockchain, control is passed to the voter's account. A valid voter's account has one coin acting as a vote in the election in his wallet. A voter can select any one of the valid candidates. Then this decision of the voter is deployed to the sidechain with the help of smart contract. After all the validations and confirmation of the vote, this sidechain is deployed to the main blockchain. Then from the main blockchain to the candidate's wallet. Fig 4-4 gives the overview of the design of the blockchain network for our project.

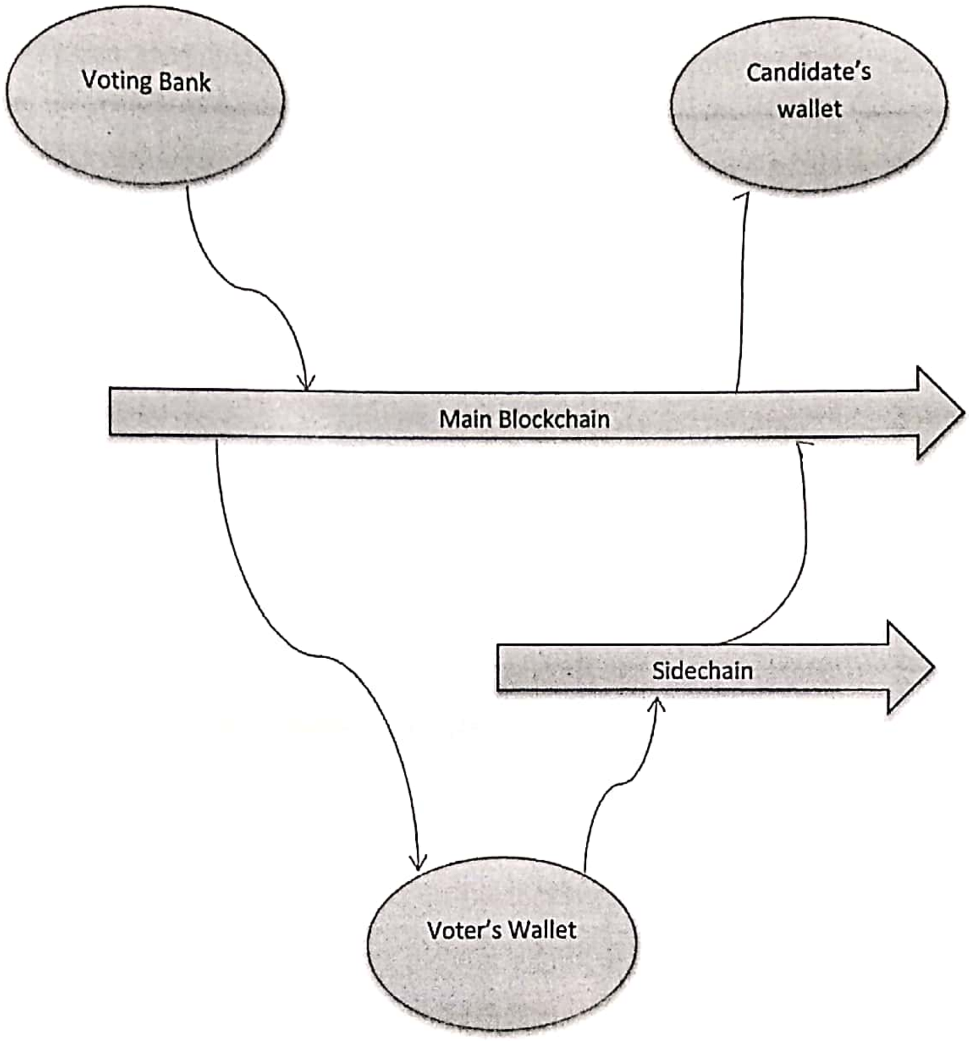


Fig 4-4 Structure of blockchain network for election

CHAPTER 5

SIMULATION AND ANALYSIS

In this chapter, we are going to discuss about the simulation of our project and analyze the output of the some of the operations performed.

5.1 Simulation

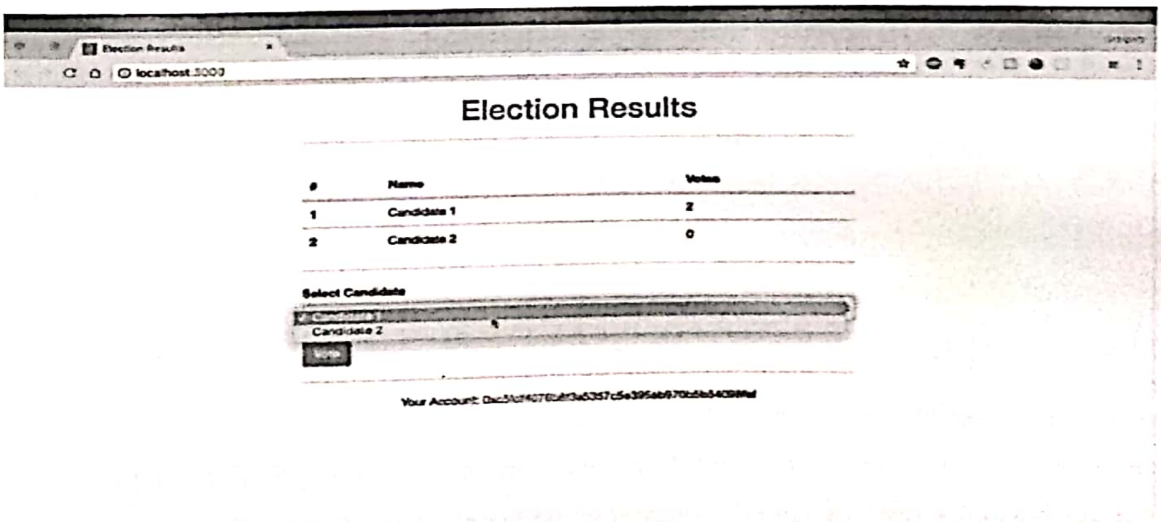
As we know that the implementation of the blockchain in real-world with millions of nodes present in the network is very challenging in some of the cases. So, a powerful simulator can be of imperative significance for practically concentrating the blockchain execution as an element of system parameters.

Following are the simulation requirements:

(i) Chrome Browser

(ii) Metamask extension enabled

(iii) Running Ganache



Election Results

#	Name	Votes
1	Candidate 1	2
2	Candidate 2	0

Select Candidate

Candidate 1
Candidate 2

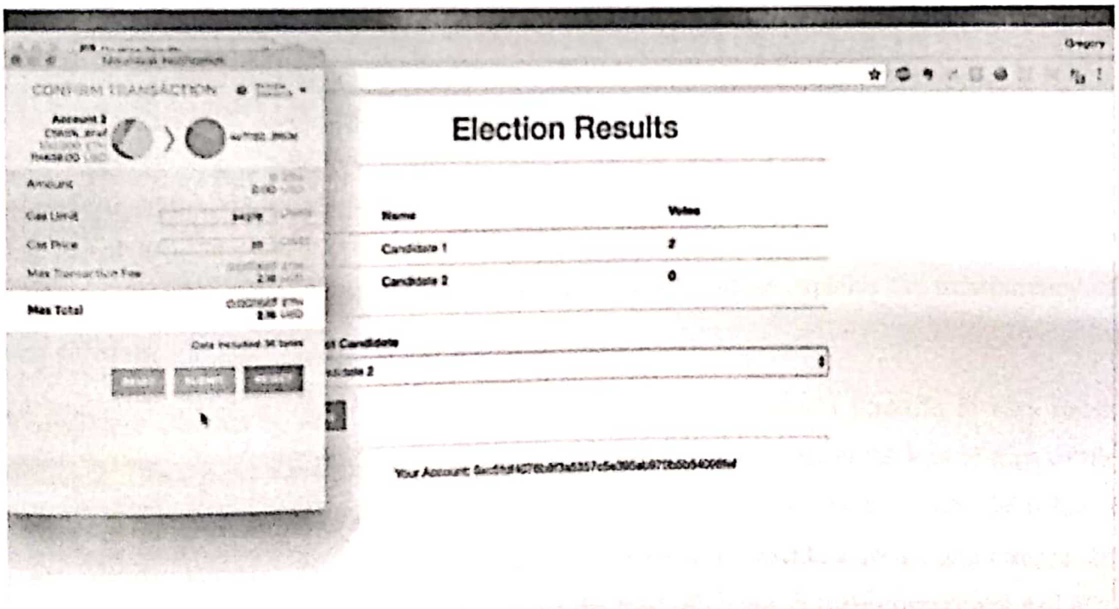
Your Account: [illegible]

Fig 5-1 DApp voter's interface

First of the all, we started our project which is running on the local host network.

Fig 5-1 shows the user interface shown to a voter. In our project, we are keeping the transparency of the system alive and shows the candidates with their particular number of votes. There, if you haven't cast your vote, then you can select the candidates from a drop-down list present just below the table of the name of candidates with their votes. Below that there is a button for the casting of the ballot. Below that is your account id which is a unique identifier.

Now, if someone want to cast his/her vote in this blockchain based e-balloting system, then after the selection of the candidate from the drop-down list and then click on the vote button. For validation, metamask prompt opens. Now, it will ask for the confirmation of the casting of the vote. It will charge some gas from your account which includes different types of fees which are applicable on a transaction.



Election Results

Name	Votes
Candidate 1	2
Candidate 2	0

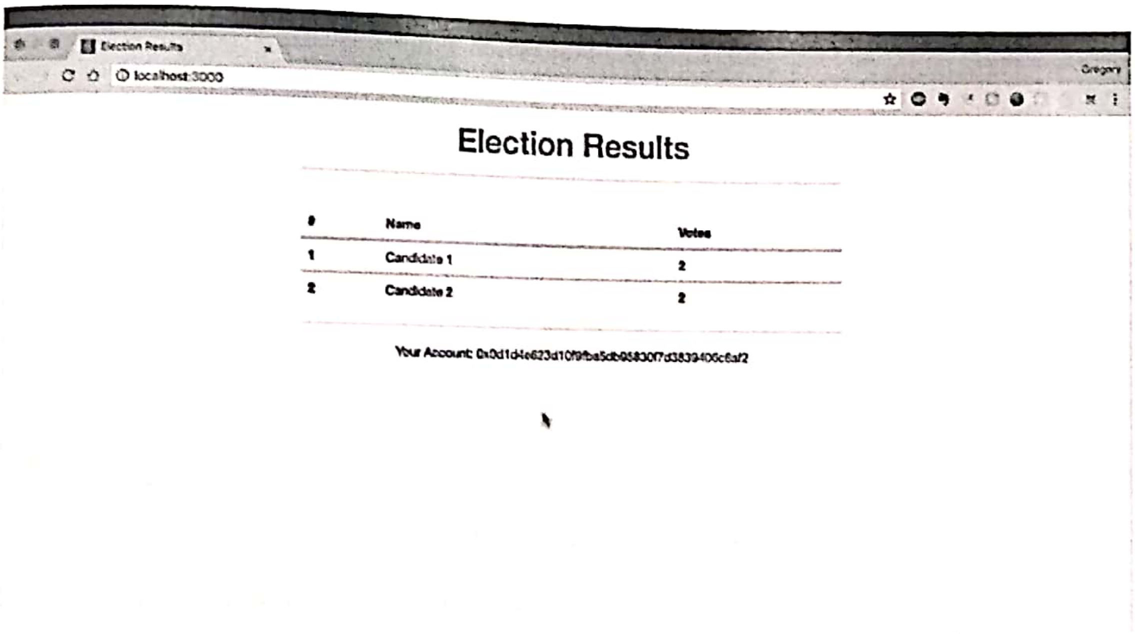
t Candidate

ndidate 2

Your Account: [illegible]

Fig 5-2 Metamask confirmation prompt

Fig 5-2 shows the metamask prompt for the confirmation of the transaction. Metamask extension is very important for the connection of our system with the ethereum blockchain. Without the metamask, we can't connect our system to the blockchain. We can say that metamask act as a bridge between the user and the blockchain. We can also perform other functions in the metamask too. For example the buying and selling of the ethers.



Election Results

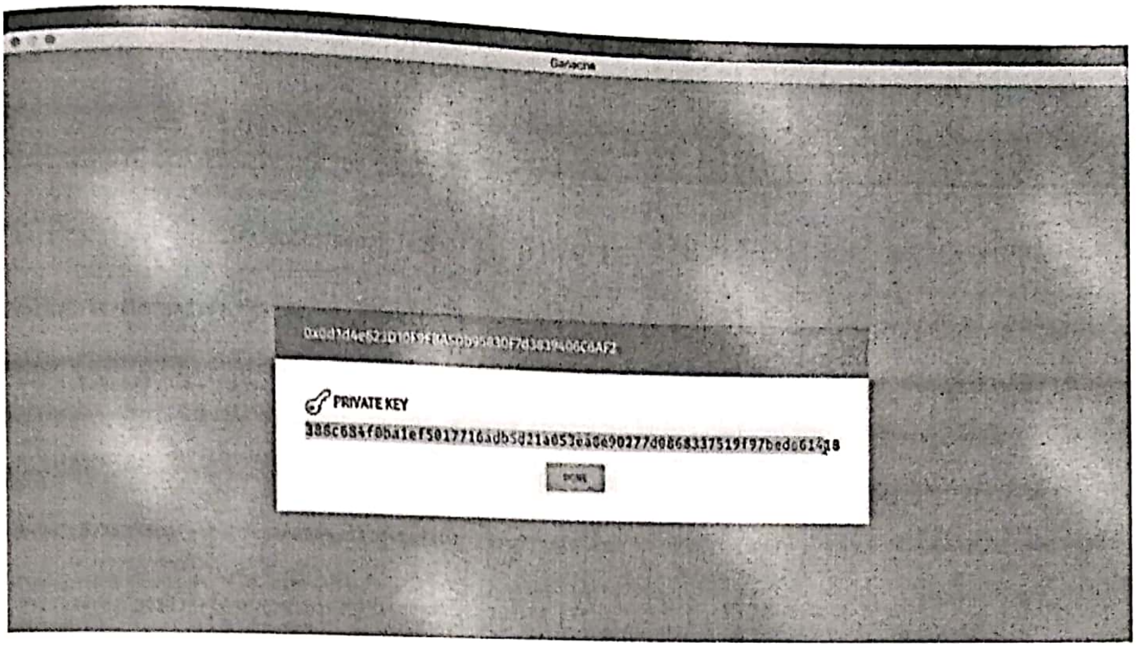
#	Name	Votes
1	Candidate 1	2
2	Candidate 2	2

Your Account: 0x0d1d4e623d10f9fba5db958307d3839406c6af2

Fig 5-3 voter's user interface after casting the vote

Fig 5-3 shows the screen to the voter after the casting of his/her ballot. Also after casting the vote, the counting of the candidates also increases, which further explains the transparency of this election.

Transparent election is very important in democracy. But the today's scenario is very much different. There is no transparency in the election which further results in the loss of trust of the people on their ruling government. Also, the EVMs used for the election are under the radar of many security vulnerabilities. So, in that case our system will provide a secure and transparent system for the election. This helps in increasing the trust of people in their government and also it will increase the vote casting percentage. With transparency, it also provides a very much secure system because of the use of wide variety of the smart contracts for this project.



0xdd7d4e621D10f9FBA5db95830F7d3839406C6AF2

PRIVATE KEY

386c684f0ba1ef5017716adb5d21a053ea0e90277d0868337519f97bede61418

Fig 5-4 Ethereum account private key

Fig 5-4 shows the private key of a particular account. This private key can be used to access the account of a particular user. This is also treated as a unique identifier for the account. For our testing purposes, we are using these kind of keys to gain the access of different account on the ethereum blockchain provided by the Ganache for the testing environment.

5.2 Analysis

In this section, we review our project on the basis some fundamental features of the blockchain technology.

Transparent

Our implemented project is fully transparent. As we have seen in Fig 5-1 and Fig 5-3 , it is also showing the results of the candidates with their names and numbers of votes casted to each candidate against their name.

Vote Secrecy

Vote Secrecy is maintained in our system. As there is no details present in the system that which voter casts their vote to whom. Vote secrecy is very important in terms of elections.

Vote Only Once

In our system, a voter can cast their ballot only once in the election. He/She can't update his/her choice after the casting of the vote. In this way, our system is handling the problem of double spending.

Security

As we are using multi-smart contracts for our system. The chances of malicious attacks is very less as compared to the other system. Smart contracts holds code of the DApp, governing rules of elections, candidates information and many other things.

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this project, we started with reviewing the challenges of the blockchain technology and issues of smart contracts. We also discuss the challenging of using blockchain technology for the voting purpose. Researchers have been studying various proposed methodologies and the challenges associated with it for the successful implantation of the blockchain-based e-voting. We provide a methodology and it addressed different concerns associated with the balloting like transparency etc. We focused on the most important part of the election i.e. security. Securing the elections from the malicious attacks is one of the biggest concerns. One malicious attack can change the whole situation of the election and government loses faith.

We proposed an enhanced version of the previously proposed model for e-voting. In our model, instead of one smart contract, we are using multiple smart contracts for multiple purposes, which results in enhancing the security of the system. We are using blockchain network for our back-end and created an UI for voters for casting their vote.

Future works include studying the application of advanced cryptographic techniques for enhancing the security further more. And also studying various methods to increase the efficiency of the blockchain network.

CHAPTER 7

REFERENCES

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] N. Szabo. (1996). Smart Contracts: Building Blocks for Digital Markets. [Online]. Available: http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html

[3] B. Wang, J. Sun, Y. He, D. Pang, and N. Lu, “Large-scale Election Based On Blockchain,” Procedia Computer Science, 129, pp. 234 – 237, 2018.

[4] R. Casado-Vara1 and J. M. Corchado, “Blockchain for Democratic Voting: How Blockchain Could Cast of Voter Fraud,” 2018.

[5] A. K. Koç, E. Yavuz, U. C. Çabuk, and G. Dalkılıç, “Towards Secure E-Voting Using Ethereum Blockchain,” 2017.

[6] S. Bartolucci, B. Pauline, and J. Daniel, “SHARVOT: secret SHARe-based VOTing on the blockchain,” arXiv preprint:1803.04861, 2018.

[7] T. Moura and A. Gomes, “Blockchain Voting and its effects on Election Transparency and Voter Confidence,” In Proceedings of the 18th Annual International Conference on Digital Government Research, pp. 574-575. ACM, 2017.

[8] A. B. Ayed, “A Conceptual Secure Blockchain-Based Electronic Voting System,” International Journal of Network Security & Its Applications (IJNSA) ,Vol. 9, No. 3, May 2017.

[9] I. Kubjas, “Using blockchain for enabling internet voting,” 2017.

[10] K. Hegadekatti, “Democracy 3.0: Voting Through the Blockchain,” 2017.

[11] R. Osgood, “The Future of Democracy: Blockchain Voting,” 2016

[12] U.S. Official: Hackers targeted voter registration systems of 20 states, Associated Press, Sep. 2016.

[13] M. Daniel, "Blockchain Technology: The Key to Secure Online Voting", Bitcoin Magazine, Jun. 2015.

[14] A. Lewis, "A Gentle Introduction to Blockchain Technology," 2015. Available: https://bitsonblocks.net/2015/09/09/a-gentle-introductionto-blockchaintechnology/

[15] Blockchain voting: The end to end process, https://followmyvote.com/blockchainvoting-the-end-to-end-process/

[16] A. Sapirshtein, Y. Sompolinsky, and A. Zohar, "Optimal selfish mining strategies in bitcoin," arXiv preprint arXiv:1507.06183, 2015.

[17] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project Yellow Paper, 2014.

[18] https://blog.pwc.lu/smart-contracts-adoption-challenges/.

[19] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, pp. 254-269, ACM, 2016.

[20] C. Natoli and V. Gramoli, "The blockchain anomaly," in 15th International Symposium on Network Computing and Applications (NCA), 310-317, IEEE, 2016.

[21] M. Vukolić, "Rethinking permissioned blockchains," in Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, BCC '17, pp. 3-7, ACM, 2017.

[22] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Gollamudi, G. Gonthier, N. Kobeissi, N. Kulatova, A. Rastogi, T. Sibut-Pinote, N. Swamy, et al., "Formal verification of smart contracts: Short paper," in Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security, pp. 91-96, ACM, 2016.

[23] G. Bigi, A. Bracciali, G. Meacci, and E. Tuosto, "Validation of decentralised smart contracts through game theory and formal methods," in Programming Languages with Applications to Biology and Security, pp. 142-161, Springer, 2015.

[24] C. K. Frantz and M. Nowostawski, “From institutions to code: Towards automated generation of smart contracts," in 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W), pp. 210-215, IEEE, 2016.

[25] K. Delmolino, M. Arnett, A. Kosba, A. Miller, and E. Shi, “Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab," in International Conference on Financial Cryptography and Data Security, pp. 79-94, Springer, 2016.

[26] F. Idelberger, G. Governatori, R. Riveret, and G. Sartor, “Evaluation of logic-based smart contracts for blockchain systems," in International Symposium on Rules and Rule Markup Languages for the Semantic Web,167-183, Springer, 2016.

[27] B. Marino and A. Juels, “Setting standards for altering and undoing smart contracts," in International Symposium on Rules and Rule Markup Languages for the Semantic Web, pp. 151-166, Springer, 2016.