International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 03 | Mar 2019 www.irjet.net p-ISSN: 2395-0072

Decentralized E-Voting System

Venkata Naga Rani B[1], Akshay S[2], Arun kumar M[3], Ishwar Kumar M A[4]

[1]*Assistant Professor, Easwari Engineering College, Bharathi Salai, Ramapuram, Chennai – 600089*
[2,3,4]*Student, Easwari Engineering College, Bharathi Salai, Ramapuram, Chennai – 600089.*

Abstract - *In a parliamentary government like India, when the electronic voting system overtook the traditional pen and paper based Ballot voting system, it had the potential to increase security and reduce fraudulence making the voting process traceable and verifiable. But, this system is centralized and has many limitations like proxy casting, recasting, and is not verifiable. These limitations can be handled by one disruptive innovative technology called Blockchain which is a distributed, immutable, incontrovertible, public ledger. It has its applications on almost various domains of Finance, Stock Market, Governance, IoT, Protection of Intellectual Properties, Crowdfunding, File Storage etc. The main concept is to combine the technology of blockchain with Cryptographic Hash Function and Digital Signatures in order to realize the decentralized e-voting system with all the requirements of voting process without a trusted third party. Our novel electronic voting system evaluates some of the popular blockchain frameworks for the purpose of developing E-voting protocol that utilizes the blockchain as a transparent ballot boxes linked using cryptographical techniques to solve the issue of unauthorized altering of data, provide voter privacy, total transparency and independently verifiable output. Implementation of a voting system as a smart contract deployed on Remix IDE using Solidity language running on Ethereum and node server which creates nodes for every user to store encrypted vote details in each block and thus providing a transparent and robust system for medium size elections.*

***Keywords*:- E-voting - Blockchain - Decentralized environment – Cryptographic Hash Function - Digital Signatures – Distributed Consensus – Ethereum - Smart contracts – Merkle Trees – Node JS – Remix - Solidity**

1. INTRODUCTION

After ballot based voting system was replaced with electronic voting machines, the potential to increase security and fraudulence was addressed. This made the voting system to be traceable and verifiable but by the election authorities only. Since then, Electronic voting systems have been the subject of active research for decades, with the goal to minimize the cost of election process, while maintaining the election reliability by satisfying the security, privacy and compliance requirements. But, it is centralized by design; i.e. a single authority (third party) controls the code base, the database, and the system outputs. Also it is not open-source, and not independently verifiable output with physical security concerns in Electronic voting machines. Addressing all these limitations our aim is:

(a)To provide a decentralized e-voting system with all the requirements of voting process and will be able to solve the issue of unauthorized altering of data and provide voter privacy and total transparency

(b)To create a secure and reliable voting protocol which the voters control as a network of peers and allow all voters to contribute in both the validating and confirming of ballots.

The blockchain technology is therefore considered to be the ideal tool, to be used to create the new modern democratic voting process. This new technology has three main features:

(i) Immutability: Any proposed "new block" to the ledger must reference the previous version of the ledger. This creates an immutable chain, which is where the blockchain gets its name from, and prevents tampering with the integrity of the previous entries.

(ii) Verifiability: The ledger is decentralized, replicated and distributed over multiple locations. This ensures high availability (by eliminating a single point of failure) and provides third-party verifiability as all nodes maintain the consensus version of the ledger.

(iii) Distributed Consensus: A distributed consensus protocol to determine who can append the next new transaction to the ledger. A majority of the network nodes must reach a consensus before any new proposed block of entries becomes a permanent part of the ledger.

Blockchain has become the backbone of modern internet with its applications ranging from finance to micro grids such as Banking, Stock Market Predictions, Crowdfunding, Governance, Supply chain auditing, File storage, Protection of Intellectual properties, Internet of Things, Neighborhood Micro grids, Land Registration

and so on. On account of these many applications, many frameworks for developing and implementing blockchain were developed and tested in virtual networks like Ethereum and Ropsten networks. They are blockchain based distributed computing platform for deploying contracts for blockchain transactions

This paper evaluates the existing researches and frameworks used for constructing blockchain architecture and the purpose of using it for secure voting systems. Fig 1.1 shows a simplified view of blockchain. The idea is to combine the blockchain technology with Cryptographic Hash Function and Digital Signatures in order to realize the decentralized e-voting system with all the requirements of voting process without a trusted third party. Our novel electronic voting system develops an E-voting protocol that utilizes the blockchain as transparent ballot boxes linked using cryptographical techniques. It is implemented as a smart contract which runs on Ethereum network and uses node js for creating nodes for every user to store encrypted vote details in each block and thus providing a transparent and robust system for medium size elections.

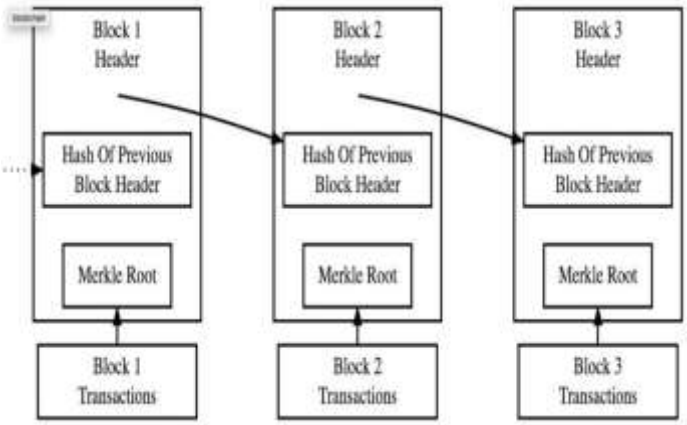

Fig – 1.1: Simplified Blockchain

2. LITERATURE SURVEY

Bin Yu et al [2] proposed that cryptographic techniques are employed to make sure that the security of voting systems in order to increase its wide adoption. However, in such digital e-voting systems, the public bulletin board that is hosted by the third party for publishing and auditing the voting results must be trusted by all participants. Currently, a number of blockchain-based solutions have been proposed to address this issue. However, these systems are not practical to use due to the limitations on the voter and candidate numbers supported, and their security frameworks, which heavily depends on the underlying blockchain method and typically suffers from potential attacks (e.g., force-abstention attacks). To handle with two above mentioned issues, they introduce a practical platform-independent secure and verifiable voting system that can be implemented on any blockchain that supports the execution of smart contracts. Fig 2.1 shows Blockchain e-voting system.

In this phase, they first provide an overlook of the whole voting protocol and then discuss each step in details. Except for the smart contract administrator who sets the voting smart contract, three terms are involved: voters, smart contract, and voting administrator. They take Bob as a valid voter in this phase to show how the voting protocol works. First, the smart contract is started to begin voting. Then, the voting administrator uploads the voting parameters. After all the voters register themselves in the voting process and provide their SLRS public key to the blockchain, the administrator begins the start of the voting phase. Bob as a voter casts his ballot vote before the administrator triggers the tallying phase. It is optional for Bob to verify the tally result before the administrator acquires the encrypted tally result. The administrator needs to upload the voting result and the proof to the blockchain to show the scope and validity of the result to the voters and all the stakeholders involved. The smart contract checks whether the obtained result matches the proof uploaded by the administrator and finally publishes the decrypted voting result on the blockchain. For downloading the parameters of 1 million voting, it takes about 4ms for a 100MB network and is approximately 15ms and the average value of in their test system is 776.60ms. In conclusion, it consumes about 1s for a voter to cast his vote in a setting of million voters voting. The time spent on publishing the result is less than 2s as they optimize the decoding algorithm by using shifting operation. To solve the problems that the present blockchain voting system cannot provide the comprehensive security features, and most of them are platform dependent, they have proposed a blockchain-based voting system that the voter's privacy and voting correctness are guaranteed by linkable ring signature, homomorphic encryption between the voter and blockchain. They analyze the correctness and the security of their voting system. The experimental results show that their voting system achieves a reasonable performance even in a large scale voting.

the main idea of cryptographic voting schemes is to provide transparency while protecting the ballot secrecy and to enable a fast tallying function. In this paper, they address three major issues of cryptographic voting schemes.

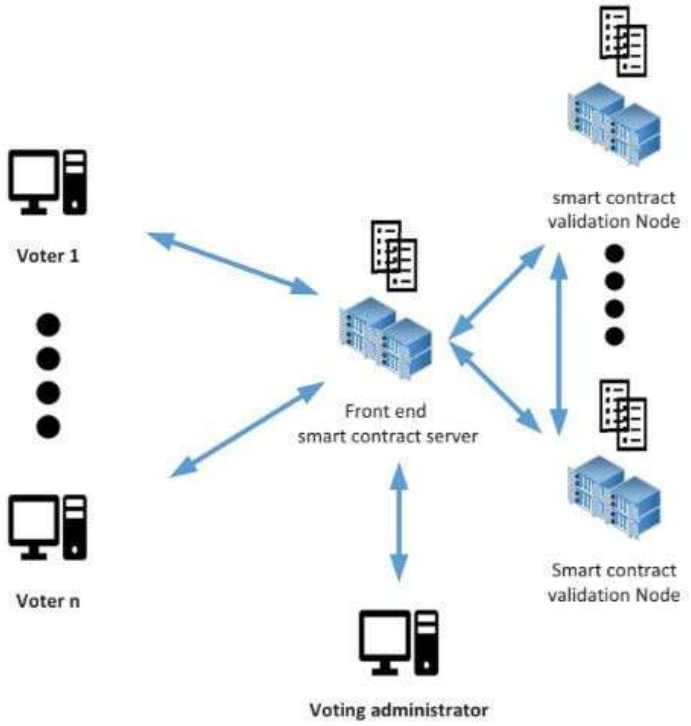

Fig-2.1 : Blockchain e-voting system

Fernando Lobato Meeser et al [5] proposed that the first implementation of a voting system as a smart contract running on Ethereum that uses threshold keys and linkable ring signatures to provide a robust and transparent system that could be implemented for medium size elections. Each voter is responsible for his/her vote and can monitor his/her vote while remaining anonymous amongst a set of users. The protocol minimizes the existing centralization by the use of threshold cryptography, allows the voting to be tallied by anybody and does not require each user to vote for the tallying to be exact. All the execution of the protocol is well ensured by the safety of the Ethereum protocol. They deployed the contract on the Ethereum test network and provided some analysis on feasibility and costs. In their scheme, they assume an election authority that will be in charge of coordinating the third parties holding the secret shares, publish the election contract to the blockchain and registering users into the contract. Before uploading the contract to the blockchain, the election authority and n third parties must get together in a public gathering, generate a key-pair. Split the key pair into n secrets where each third party will take a secret with them and destroy the original secret without keeping a copy. Next, the election authority must submit the contract with the election rules to the blockchain, present the code and provide the address of the contract. This way any voter can compile the code and verify whether it's the same contract. In this paper, they presented a scheme for running an election on the top of a blockchain so that any interested party can have a cryptographic assurance of the outcome of the election. They did this by implementing a smart contract on the top of Ethereum that enforces the correct execution of our voting scheme. Although our contract can't currently cast votes on the Ropsten test network or the Ethereum test network, we have implemented a blockchain voting scheme that could handle elections. This also shows us some of the current limitations by the existing blockchain infrastructure. While the idea of blockchain brings us an ideal decentralized trustless environment where we can host important human processes in which we wish to minimize the trust we place on individuals and institutions, it is far away from being the robust infrastructure needed to solve world problems. Nevertheless, given the spark of its inception, it hasn't even marked the decades there are probably still a lot of interesting advances yet to see in the coming years. In future work, they will investigate other types of ring signatures such as Borromean ring signatures and how to effectively use Ethereum as a dedicated computing environment on a separate network just for election purposes. Given the nature of Ethereum open source community, security and bugs issues are brought to light very fast, this makes it attractive from a security perspective. However, the Ethereum main network can become confectioned with the daily flow of financial transaction and existing smart contract execution. Future research will look at how to run the election in the cloud, where anybody can deploy a node in a Proof-of-Authority network where they can trust particular institutions to mine votes but anybody can be listening and relaying. The block size can be easily increased and the gas limits could also be raised to make this a feasible system.

Friðrik Þ. Hjálmarsson et al [7] states that building an electronic voting system that satisfies the legal requirements of legislators has been a challenge for a long time. Distributed ledger technologies are an exciting technological advancement in the information technology world. Blockchain technologies offer an infinite range of applications benefiting from sharing economies. Their paper aims to evaluate the application of blockchain as service to

implement distributed electronic voting systems. The paper elucidates the requirements of building electronic voting systems and identifies the legal and technological limitations of using blockchain as a service for realizing such systems. Their paper starts by evaluating some of the popular blockchain frameworks that offer blockchain as a service. They then propose a novel electronic voting system based on blockchain that addresses all limitations we discovered. The protocol proposed was focused on the first class, where strong voter privacy was the primary objective which had two challenges. First challenge was that there exists no trusted third party. With a trusted third party, many security problems can be easily solved, but could lead to the 'trusted' third party to become the one who breaks the security policy. The goal therefore was to eliminate the use of a trusted third party altogether. The second important challenge was that there would be no voter-to-voter private channels to ensure dispute freeness, i.e. everybody could check whether all voters had followed the protocol faithfully. These challenges were fulfilled in the AV-net, but the new protocol proposed a new solution which solved the downside of the AV-net, heavy computational load for each voter, which increased linearly with the number of voters. The new protocol proved as secure as the AV-net and can be seen as a generalization of the AV-net protocol, but significantly more efficient. The first round in the two-round protocol consisted of every participant to publish his public key and a zero-knowledge proof (ZKP) for his private key. When the round finished, each participant checks the validity of the ZKPs and computes. In the following round, each participant needs to demonstrate that the encrypted vote was one of the valid voting choices without revealing which one. The self-tallying function then works in a way that the public keys of all participants are combined in such a structured way that the random factors of the private keys immediately vanish, thus revealing the tally. There are limitations with this protocol concerning national elections. The protocol requires a collaborative effort from all voters; otherwise, the protocol will not work. For example, if some voters refuse to send data at the end of round 1, the tallying process will fail, and everyone will know who those voters are and can expel them and restart the protocol. This leads to the voting process to be delayed, which would be severely costly for a large scale election. Another limitation is the lack of resistance of coercion because of the remote voting; a voter can be forced to reveal his secret value, therefore revealing how he voted. The idea of adapting digital voting systems to make the public electoral process cheaper, faster and easier, is a compelling one in modern society. Making the electoral process cheap and quick, normalizes it in the eyes of the voters, removes a certain power barrier between the voter and the elected official and puts a certain amount of pressure on the elected official. It also opens the door for a more direct form of democracy, allowing voters to express their will on individual bills and propositions. In this paper, they introduced a unique, blockchain-based electronic voting system that utilizes smart contracts to enable secure and cost-efficient election while guaranteeing voters privacy. They have outlined the systems architecture, the design and security analysis of the system. By comparison to previous work, they have shown that the blockchain technology offers a new possibility for democratic countries to advance from the pen and paper election scheme to a more cost- and time-efficient election scheme, while increasing the security measures of today's scheme and offer new possibilities of transparency. Using an Ethereum private blockchain, it is possible to send hundreds of transactions per second onto the blockchain, utilizing every aspect of the smart contract to ease the load on the blockchain. For countries of greater size, some measures must be taken to withhold greater throughput of transactions per second, for example, the parent & child architecture which reduces the number of transactions stored on the blockchain at a 1:100 ratio without compromising the network's security. Their election scheme allows individual voters to vote at a voting district of their choosing while guaranteeing that each individual voters vote is counted from the correct district, which could potentially increase voter turnout.

Clement Chan Zheng Wei et al [13] states that current electronic voting protocol requires a centralized system to control the whole procedure from ballot inputs to Output results and election monitoring. Meanwhile, blockchain provides a decentralized system which opens across the whole network of untrusted participants. Applying blockchain into electronic voting protocol through a proper architecture can instill characteristics such as data confidentiality, data integrity, and data authenticity. In their paper, they discuss a proposed method on how to leverage the advantages of blockchain into an

electronic voting protocol. Their blockchain-based electronic voting protocol promises to provide a secure electronic election process given the proposed system works. They implement a protocol using blockchain technology to turn election protocol into an automated control system without relying on any single point of the entity. Lastly, they discuss the characteristics of our proposed blockchain-based electronic voting protocol in their paper. In term of architecture design for the proposed system, the application server and e-voting server is separated and divided. Only a network node is set up between the application server and e-voting server to communicate and exchange voter credentials information and ballot ID. The application server stores all the voter credentials information and ballot information, while the e-voting server stores the ballot spreadsheet which will contain the voter result. The purpose of separating the server is to prevent a single point of failure and ensure voter anonymity. Each system (the Application server and the E-voting server) will have their public-private key pair to secure messages transmission and communication. During an e-voting session, the voter can cast their ballot through e-voting server. The ballot spreadsheet contains the voter vote, voter public address signing the pseudonymous the identity of the voter on the spreadsheet, and the digital signature of the voter. Because only the pseudonymous the identity of the voter is signed on the ballot, no particular individual will be linked to their vote result directly, only the owner themselves know their vote belonging. Before the voting server accept the ballot into the voting chain, it will verify the voter credentials whether it is tally with the ballot ID generated previously during

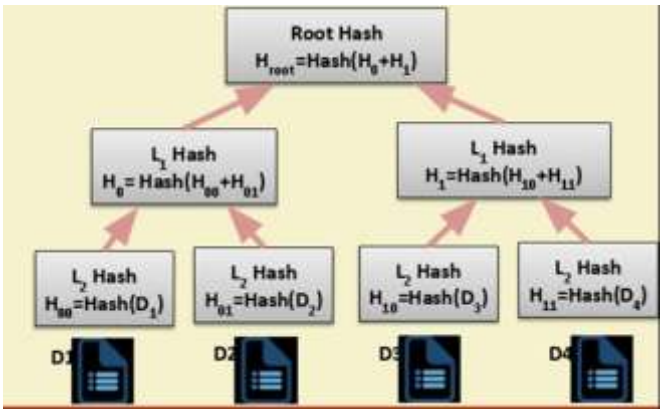

Fig – 2.2 : Merkle Tree

the registration process. Once the ballot ID is verified, the ballot will finally be accepted and recorded. The voter can access the voting server and cast their vote if the election is still going on. Each ballot submitted is also encrypted with the application server public key before adding into the voter result chain. The purpose of encrypting the ballot is to prevent the voter result from being exposed during the voting phase. As we know blockchain spreadsheet is a publicly shared record, everyone can access the chain if they are intended to do so. Hence, knowing the real-time result beforehand might affect the voter decision. Encrypting the voter result chain with the application server public key during the voting phase is to protect the voter result being exposed before the election end. In conclusion, they proposed a blockchain-based electronic voting protocol in their paper. The electronic voting system makes use of blockchain technology properties to enhance its security features. The system can secure the identity of every voter and ensure that all the vote results recorded are tamper-proof. Blockchain provides advantageous properties for e-voting systems such as authenticity, integrity, verifiability, anonymity, availability and a general consensus from every participant. The system does not rely on human trust but on computational cryptographic trust. Fig 2.2 shows Merkle tree structure

Olaniyi Olayemi M et al [3] proposed that the performance assessment of an imperceptible and robust secured stegano cryptographic model of electronic voting. The Performance analysis was achieved to an extent based on the degree to which the model meets the criteria of generic and functional requirements of secured e-voting system: authentication, integrity, confidentiality, and verifiability as well as other functional security requirements of secured voting using five-point psychometric analysis. The result achieved of the quantitative evaluation of the model assert that the model possessed capacity to guarantee and validate voter's for who they said they are, guarantees the integrity and validity of elections, ensures privacy of the voters, guarantees the confidentiality of the vote and therefore, provides mechanism for fraud detection after the election process in developing country where digital divide is significant. The architecture provides greater application high efficiency and flexibility, since each tier runs on a separate machine to improve system performance. The pre-election phase of the system involves the registration of all the entities that will

enable the result of the current election, such entities are: Voters information, administrators, Candidates and Parties information, which is all stored in the database. The election steps involves ballot submission of electronic casting and security of vote, such that the vote entered is encrypted with the RSA encryption algorithm specifically for the provided poll site and kiosk e-voting scenario while the ECC encryption algorithm is specifically meant for a remote mobile voting scenario. The encrypted text (handled as bit) is embedded in common multimedia graphics generated by the system using both video and image steganographic algorithms/techniques and then sent to the server. The voter's fingerprint and accurate response of the voter to both dynamically generated questions regarding grid and mobile short message service (SMS) is used for authentication of the voter to validate the original identity of the voter. The post-election phase is where the vote cast is extracted from the stego object and then decrypted via the extraction technique of the Image and Video steganographic technique as well as and decryption definitions of an appropriate cryptosystem (RSA or ECC based on voting, scenario), and the final conclusions are retrieved, collated, and processed. At this phase, e-votes are counted and final conclusions are retrieved and displayed for voters. The fundamental technically well-built secured e-voting requirements of voter's authentication, vote confidentiality, vote integrity and verification as well as functional requirements of secure electronic-voting systems like a scope for rigging, democracy are preferential assessing factors to impact electorate choice decision. The findings of their paper will make the steganographers, software developers and government in making a sound decision on what to consider in designing, developing and administration of secure electronic-voting systems for future free, fair and credible e-democratic decision making through e-voting.

3. PROPOSED DECENTRALIZED SYSTEM

This section discusses proposed simple e- voting system based on blockchain to record and evaluate the result of an election. This system architecture requires user to install a mobile app that is connected to the Internet for voter's convenience. This system is locally tested in ethereum network using testrpc, and can be extended to largescale using geth.

A: Registration Phase

In the registration phase, the admin who is in charge of elections, logs in to system with admin password and registers candidate and voter details. In candidate registration, credentials of candidate like candidate name, aadhaar number , mobile number, age, email and so on are stored in database with aadhaar number as primary key. Election details of candidate like standing party and constituency also is stored. The application also checks if candidate with same credentials are present and saves the data.

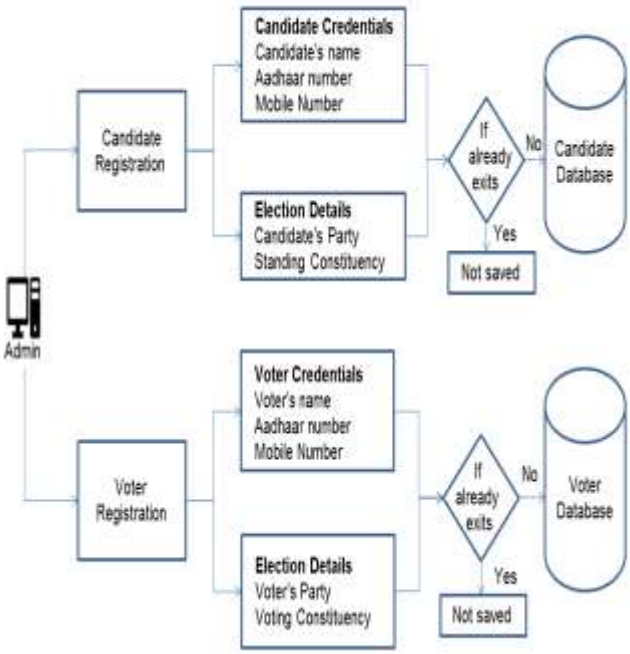

Fig – 3.1 : Registration Phase

In candidate registration, credentials of candidate like candidate name, aadhaar number , mobile number, age, email and so on are stored in database with aadhaar number as primary key. Election details of candidate like standing party and constituency also is stored. The application also checks if candidate with same credentials are present and saves the data. Authentication is done while voter is voting in next phase using OTP sent to linked mobile number of voter. These details get stored in mySql database which can be accessed using my Sql Query Browser. Fig 3.1 shows registration phase.

B: E-voting Phase

In E-voting phase, after the testRPC is setup and Ethereum environment is created using Remix Solidity, node server is run which creates nodes for every user who logs in to vote using mobile application. Note that the system which runs blockchain and mobile using the mobile app must be connected to the same network. First the voter logs in to the mobile app using aadhaar number and ip address of currently connected network. After

logging in OTP is sent to the linked mobile number which is to be entered for authentication. If authentication is successful, vote menu is displayed, else login failed.

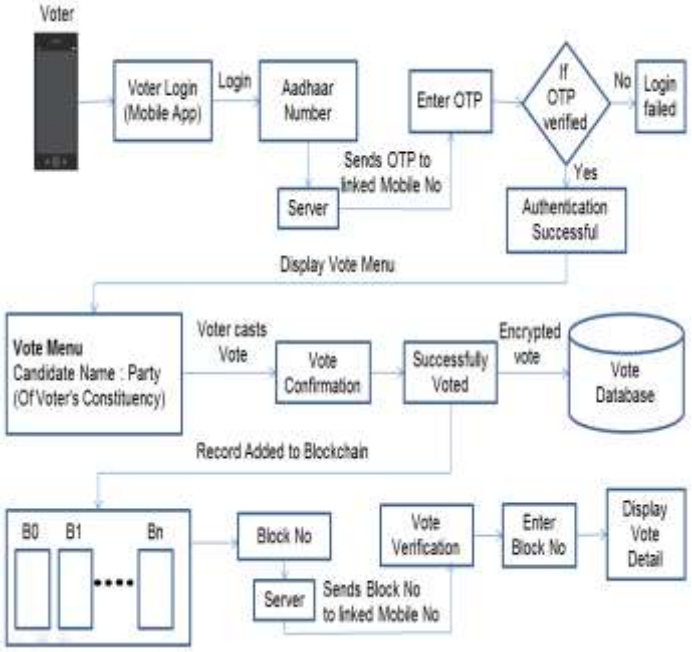

Fig – 3.2 : E-Voting Session

After successful authentication, vote menu is displayed from which user has to select a candidate as their vote for their constituency. After vote is cast, confirmation message is prompted and vote is encrypted and stored in vote database. Now the voting session is over. In the backend, the vote details are stored as a block in blockchain. For each vote, a record is inserted into blockchain along with the record number. After successful recording of blocks into blockchain, the block no is sent to server and it is sent as SMS to linked mobile number using which a voter can verify their vote anytime. For this, the user must again log in to the app using aadhaar number and OTP and in the next menu select "View vote" to verify their vote. This provides independently verifiable output. Moreover only the voter can view their vote using block no and OTP through their mobile. Even admin cannot see others vote and thus providing Voter privacy and Vote privacy and the whole system is transparent also providing transparency enhancing confidence of voters. This system also prevents user from trying to vote again (Recasting) by prompting that the user has already voted. Thus it provides prevention of Recasting and Proxy casting. Fig 3.2 shows E-voting phase.

C: Blockchain phase

After the registration phase, and before the voting phase, blockchain has to be started in Ethereum environment. First testRPC is started. It is a Node.js based ethereum client for testing and development in which uses ethereum js to simulate full client behavior. Now, the smart contract is run on Remix IDE written in Solidity language. It is connected to the Web3 component using localhost port. The node js server is run which creates node for every logged in voter for storing vote details. The first vote creates a genesis block (Block 0) from which consecutive blocks are added. Each block contains block no, transactions root, block hash, merkle hash, difficulty, nonce, timestamp. Each block is connected to previous block using previous block's hash which makes it immutable. Encrypted vote details are stored in JSON file and at last the result is tallied and announced Fig 3.3 shows Blockchain implementation phase.

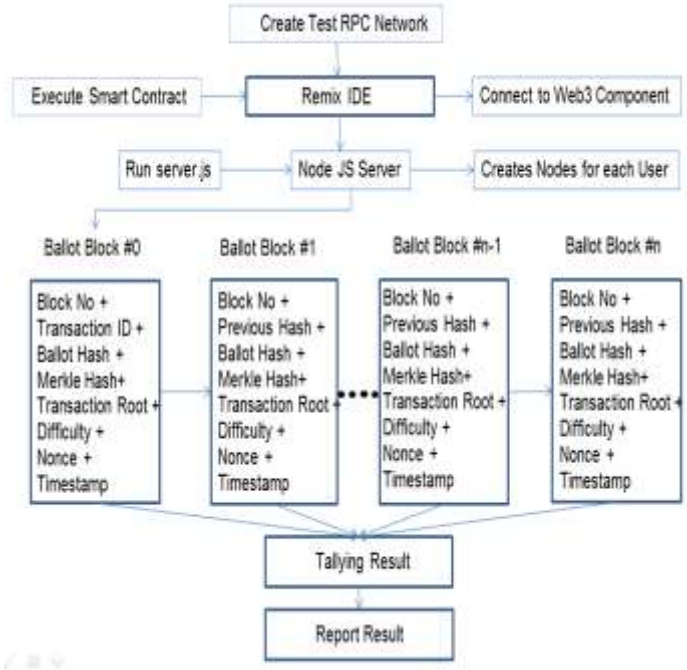

Fig – 3.3 : Blockchain Phase

4. SYSTEM IMPLEMENTATION

By design, the blockchain technology and its applications are always decentralized. Anything that happens on blockchain is a function of the network as a whole. Some important implications root from this. By creating a new way to verify transactions, the aspects of traditional commerce could become unnecessary. Stock market trades can become almost simultaneous on the blockchain technology, for instance — or it could make

types of record maintaining, like a land registry, fully public. And decentralization is already a reality. A globally interconnected network of computers eventually uses blockchain technology to jointly manage the database that records Bitcoin transactions. That is, Bitcoin is controlled by its entire working network, and not by any one single central authority. Decentralization means that the entire network operates on a user-to-user (or peer-to-peer) basis. The forms of mass collaboration are what make it possible for us to maintain and achieve Immutability, Verifiability, and Distributed Consensus.

In our paper, the overall architecture describes the following: Initially, the admin will register the user details which consist of various fields such as name, age, aadhar number, phone number and constituency and then candidate details along with their aadhar number and which party they represent is filled on the user interface. Those details will be stored on the database and these details can be later viewed by the admin. Once the registration is complete all the other phases can follow it. The architecture consists of Registration, Voting server, Blockchain formation and Verification. The complete architecture is explained in the below figure 4.1.

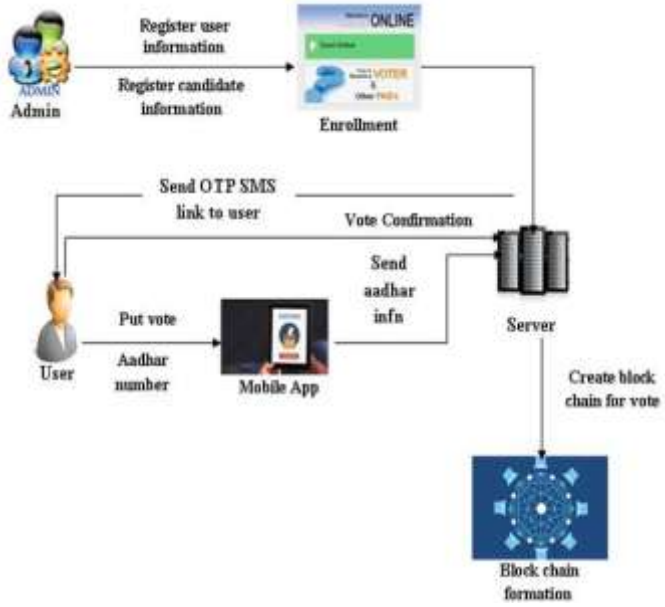

Fig – 4.1 : System Architecture

From the user side they have a mobile application on their mobile or they can even use the corresponding web application. They will log in into the application and give their aadhar number.

USER REGISTRATION:

Once the User creates a particular account, they are allowed to login into their account to access the application. Based on the User's request, the corresponding Server will respond to the User. All the particular User details will be stored in the Database of the Server. Users and candidates have to register their details along with aadhaar number.

CANDIDATE REGISTRATION:

In this module, the admin will register the candidate using their aadhaar number. Candidate registration will be made entirely using the aadhaar number and constituency of that candidate. If a user candidate provides improper information system will discard those details in registration process

BLOCKCHAIN FORMATION:

A block is a container type of data structure. The average size of a block seems to be 1MB (source block). Here every certificates number will be created as a block. For every block, a hash code will be generated for security. Here each vote information will be stored on the blockchain. If we store the information on blockchain it is more secured and every block is created based on constituency. Below Fig 4.2 shows 10 available accounts in testrpc and their addresses. This testrpc is a Node.js based Ethereum client used for the purpose of testing and development.

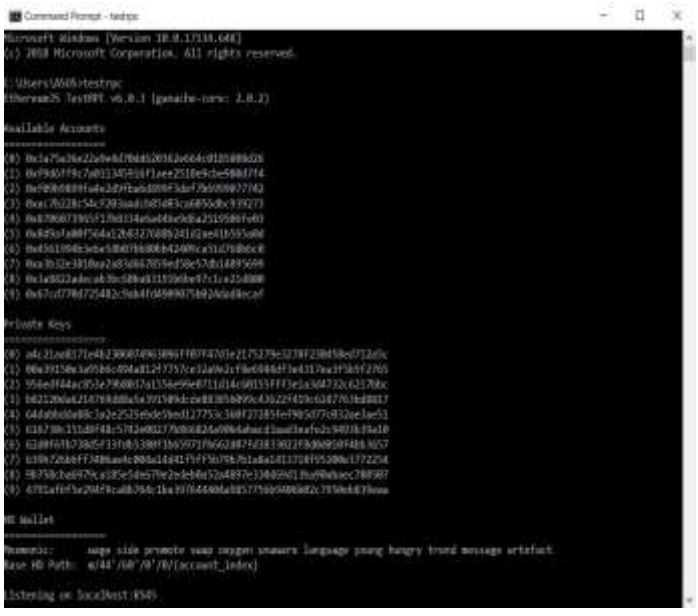

Fig – 4.2 : TestRPC

VOTING SERVER:

The Server will completely store voter's information in their database and verify them if required. Moreover, the Server has to establish a connection to communicate with the Users. The Server will update each new voter's details in the database. The Server

will authenticate and verify each voter by aadhar before they access the Application so that the user can access the Application. Fig 4.3 shows voting server listening on ip address.

Fig – 4.3 : Voting Server

REMIX SOLIDITY:

Remix is a Solidity IDE that's used to compile, write and debug Solidity code. Solidity is a completely contract-oriented, high- level programming language for writing the smart contracts. It was developed for standard languages such as C++, Python, and JavaScript. Fig 4.4 shows the Solidity code that connects sever and ethereum.

```
pragma solidity ^0.4.0;

contract Foo{
    bytes32 foo;

    function setFoo(bytes32  _foo) {
        foo = _foo;
    }
    function getFoo() constant returns (bytes32) {
        return foo;
    }
}
```

Fig – 4.4 : Solidity Code

USER VOTING:

1) In order to vote, a voter or the user must provide a valid aadhar number and enter the corresponding IP address of the network. Fig 4.5 shows the login process.

2) An OTP number is sent to the user's mobile number and this number is used to verify the user.

3) Finally, the user or voter chooses his/her party and registers the vote. Fig 4.6 shows the voting process.

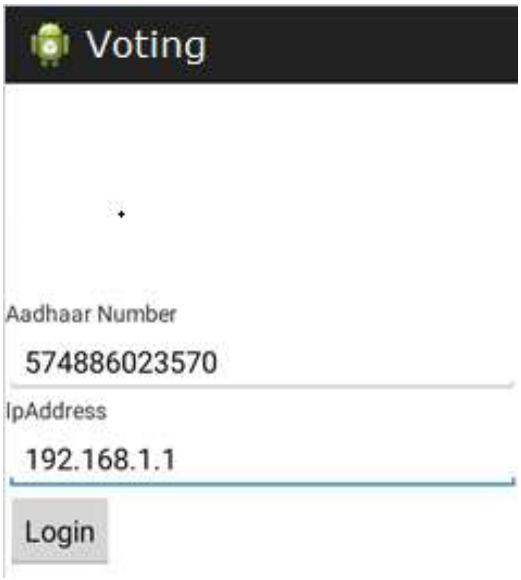

Fig – 4.5 : Login Screen

4) User vote details get stored in the block. Each block contains details like block no, block hash, parent hash, sha3uncles, transactions root, difficulty, gas limit, gas used, nonce, timestamp etc. Fig 4.7 showing vote details of a user get inserted.

Fig – 4.6 : Voting Screen

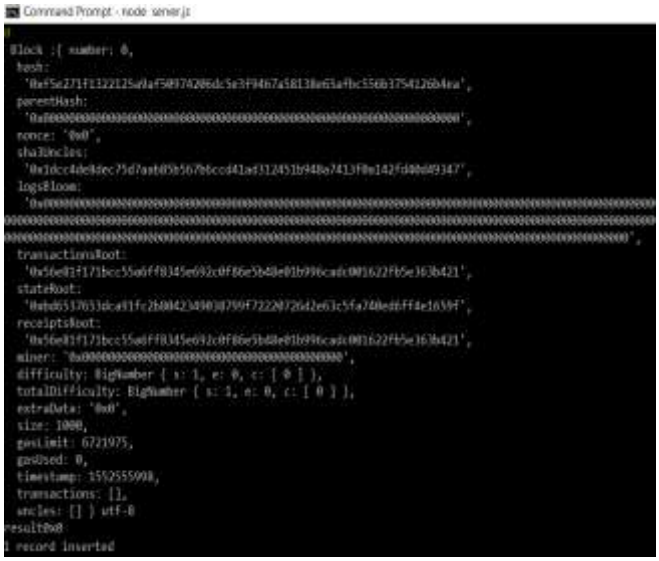

Fig – 4.7 : Record Added to Blockchain

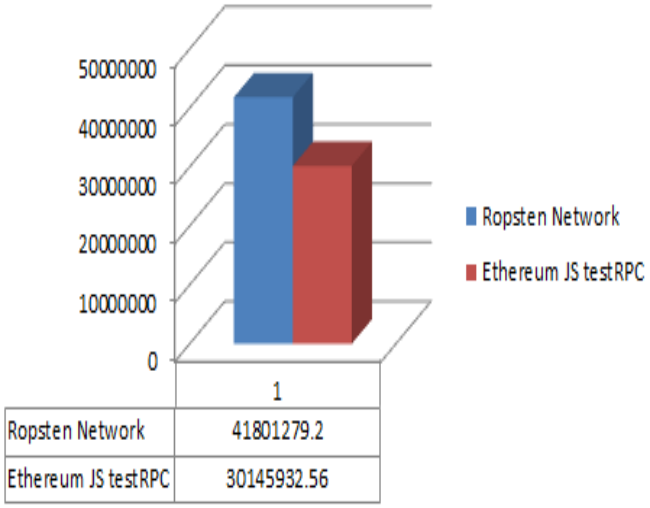

Fig 5.1 Average Gas Cost

5. PERFORMANCE ANALYSIS

Our work here is implemented using ethereum JS testRPC. The parameter that we have considered for calculating the performance of our proposed system is gas cost and processing time. All previous experiments carried on ethereum network yielded the following results shown in table 2

Table -5.1: Previous Achieved Result

METHOD	PLATFORM	AVERAGE GAS COST	VOTING TIME	PROCESSING TIME
Smart Contracts	Ropsten Networks	41801279 .20	1 sec	2 sec

Table – 5.2 Proposed System Result

METHOD	PLATFORM	AVERAGE GAS COST	VOTI NG TIME	PROCESS ING TIME
Smart Contracts	Ethereum Networks	30141932. 16	1 sec	2 sec

Ropsten Network and Ethereum JS testRPC network is compared in the following fig (graphs) 5.1, 5.2 and 5.3.

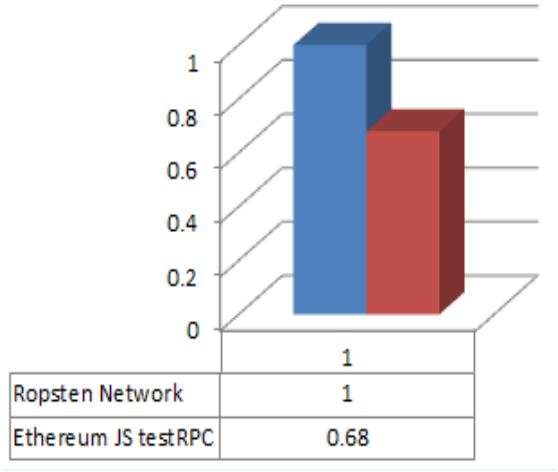

Fig 5.2 Voting Time(In Secs)

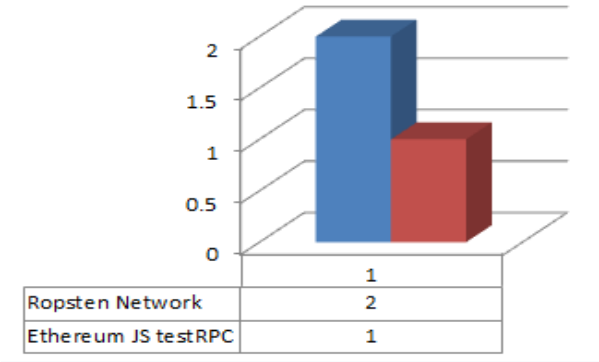

Fig 5.3 Processing Time(In Secs)

6. CONCLUSION AND FUTURE WORKS

Centralized nature of E-voting and security for E-voting is one amongst the essential problems faced by governments and organizations. Blockchain brings decentralized feature for E-voting application with minimum trusted third party. Generally- voting systems are prone to tampering, use of blockchain reduces the risk of tampering and errors. Use of encryption algorithms allows us to encrypt the final vote details which provide coercion resistance. Thus the paper infers that every people can poll their vote through a mobile application. Aadhar number is the authentication number to poll the vote on the application and for confirmation; we use OTP (one-time password). As a future work, the researchers may try to investigate the blockchain tools and techniques more deeply and develop this module into large scale use for better voting schemes.

REFERENCES

[1] Adrià Rodríguez-Pérez, [2017], "Secret suffrage in remote electronic voting systems" , 2017 Fourth International Conference on eDemocracy & eGovernment (ICEDEG)

[2] Bin Yu, Joseph Liu, Amin Sakzad, Surya Nepal, Ron Steinfeld, Paul Rimba, and Man Ho Au, [2018], "Platform- independent Secure Blockchain-Based Voting System", Springer International Publishing (2018)

[3] Clement Chan Zheng Wei, Chuah Chai Wen, [2018], "Blockchain-Based Electronic Voting Protocol", International Journal on Informatics Visualization, Vol 2(2018), No 4-2.

[4] Emre Yavuz ; Ali Kaan Koç ; Umut Can Çabuk ; Gökhan Dalkılıç, [2018], "Towards secure e-voting using ethereum blockchain" , 6th International Symposium on Digital Forensic and Security (ISDFS)

[5] Fernando Lobato Meeser, [2017], "Decentralized, Transparent, Trustless Voting on the Ethereum Blockchain", 2018 IEEE International Multidisciplinary Conference on Engineering Technology

[6] Freya Sheer Hardwick, Apostolos Gioulis, Raja Naeem Akram, and Konstantinos Markantonakis, [2018], "E-Voting with Blockchain, An E-Voting Protocol with Decentralisation and Voter Privacy", ArXiv 2018

[7] Friðrik Þ. Hjálmarsson, Gunnlaugur K. Hreiðarsson, [2018], "Blockchain-Based E-Voting System", 2018 IEEE 11th International Conference on Cloud Computing

[8] Hazem El-Gendy, Magdi Amer, [2017], "Towards a Fraud Prevention E-Voting System",International Journal of Advanced Computer Science and Applications, Vol. 4, No. 4

[9] Jens-Matthias Bohli, Christian Henrich, Carmen Kempka, Jörn Müller-Quade, Stefan Röhrich, [2009] , " Enhancing Electronic Voting Machines on the Example of Bingo Voting" , IEEE Transactions on Information Forensics and Security(2009)

[10] Lucie Langer ; Axel Schmidt ; Johannes Buchmann ; Melanie Volkamer ; Alexander Stolfik, [2009], "Towards a Framework on the Security Requirements for Electronic Voting Protocols", First International Workshop on Requirements Engineering for e-Voting Systems

[11] Mahmoud Al-Rawy, Atilla Elçi, [2018], "A Design for Blockchain-Based Digital Voting System", The 2018 International Conference on Digital Science

[12] Melanie Volkamer , Margaret McGaley, [2007], "Requirements and Evaluation Procedures for eVoting", The Second International Conference on Availability, Reliability and Security (ARES'07)

[13] Olaniyi Olayemi M, Arulogun Oladiran T, Omidiora Elijah O, Okediran Oladotun O, [2016], "Performance Assessment Of An Imperceptible And Robust Secured E-Voting Model", International Journal Of Scientific and Technology Research Volume 3, Issue 64

[14] Robert Stein, Gregor Wenda, [2014], "The Council of Europe and e-voting: history and impact of Rec(2004)11", 6th International Conference on Electronic Voting: Verifying the Vote (EVOTE)

[15] Shalini Shukla, A. N. Thasmiya, D. O. Shashank, H. R. Mamatha, [2018], "Online Voting Application Using Ethereum Blockchain", International Conference on Advances in Computing, Communications and Informatics (ICACCI 2018)

[16] Tohari Ahmad, Jiankun Hu, Song Han, [2009], "An Efficient Mobile Voting System Security Scheme Based on Elliptic Curve Cryptography", Third International Conference on Network and System Security