Blockchain using Hashing Algorithm: in Reference of Election Smart Contract

J. Dhiviya Rose,
Assistant Professor (SG),
Cybernetics Cluster, SOCS,
University of Petroleum and Energy Studies (UPES), Bidholi, Dehradun. INDIA 248007.

S.Christalin Nelson
Assistant Professor (SG),
Systemics Cluster, SOCS,
University of Petroleum and Energy Studies (UPES), Bidholi, Dehradun. INDIA 248007.

Vishal Kumar,
Student, School of Computer Science,
University of Petroleum and Energy Studies (UPES),
Bidholi, Dehradun. INDIA 248007.

Abstract - **The proposed system aims at taking a step into the field of Blockchain Technology and Cyber Security by developing a blockchain model which can be used for smart voting or any other purposes. Blockchain being a decentralized ledger is scalable as well as benefits businesses through greater transparency. It will be helpful in creating a better understanding of the process of creating a block and its verification. It will be further helpful in performing all large computations in a shorter execution time span with efficient speed for scenarios such as smart voting or any other smart contract. Verification of the block containing the data inside the block can be used for storing sensitive information that should never be threatened. They can be voter information for casting votes and electing their representative. Therefore, this project will give a better understanding of blockchain internal processes and features. In the future, this can be developed well with the concepts of file handling and object-oriented programming.**

Keywords - Blockchain, Security, Smart voting, Decentralization

I. INTRODUCTION

Business developers, entrepreneurs, technology evangelists, and computer engineers are inspired by the technologies of the modern period. Blockchain as a technology holding huge potential is an inspiration to all these individuals[1]. To acquire the technology and its needs, they have to be through with the implementation and understanding[2]. Blockchain is a decentralized distributed ledger that stores data, holds immutable property, and verified transactions[3]. Similarly talking about a concept known as the hashing algorithm, majorly focuses on generating a fixed-length result for an input value. Ethereum blockchain already has SHA256 hashing algorithm implemented and performs high computations[4]. Ledger in blockchain is cryptographically-secure, which means that cryptography has been used to provide security services which make this ledger secure against tampering and misuse. These services include nonrepudiation, data integrity, and data origin authentication. Two main cryptographic primitives are: Hash Function and Digital Signatures. Blockchain hashes are deterministic; which means that the input data will produce the same result each time[5]. They are designed to be immutable.

Following an outlook to the similar pattern, our project aims to showcase the inner processing of a secure blockchain network model. The idea has to be implemented for low constraint devices that utilize blockchain technology. As we know there are many problems with the existing manual voting system like: Expensive and Time-Consuming, too much paperwork, Errors during data entry, Loss of registration forms, the short time provided to view the voter register. The use case for our model is an e-voting system using smart contracts. Most of the co-relations in the project would be done with respect to this system in our model.

Therefore, our primary objective is to understand the block parameters working of a simple blockchain model in C language using linked list. Analyzing and comparing its hash parameters in the blocks for the verification insisting the security.

The basic objective of the proposed system is to develop a blockchain model for understanding its parameters functioning for verification and applying SHA256 hashing algorithm. The input data for the blocks will be entered through user, file handling concept is a good option for input due to which our file contents, irrespective of the quality and quantity of the content will be entered in the block making it more secure. And to develop a smart contract for a simple e-voting. For this project, major focus is going to be on the development of a blockchain structure based on its parameters in C programming. Use SHA-256 algorithm to calculate hash. SHA256 function is present in openssl/Crypto.h header file in C. We will use this library for the hashing mechanism of the blockchain.

The entire implementation of this project is explained in the following points:

- The development of the basic blockchain structure will be proceeded on any coding platform.
- Number of blocks in blockchain will be entered via user.
- In our model, block's data will be introduced using file handling in future but for now it can be either asked manually by the user or entered randomly by random function in C. Therefore, data of file will be stored in the block.
- Applying SHA256 hashing algorithms (via openssl/crypto.h) to generate hash values.
- These hash values will help in verifying about our blockchain structure hashes that are generated.
- The block is mined and is added with the other blocks in the existing structure of the block.
- If the hash of the previous block and this block is similar it will be verified.
- Therefore, the blockchain structure gets validated.

- And finally in the second module of the project which is to implement a smart contract for voting will be designed in the Solidity on the online IDE, remix Ethereum.

Keeping all the advantages and disadvantages into consideration the implementation of development of blockchain model will be done. The report is organized as follows: Section two is the problem statement for the project in reference to a e-voting smart contract. Section three is about Literature Review based on various blockchain models and their insights. Section four is about Objectives of the development of the blockchain model. Section five is dependent on the Methodology of the project. Section six is initiates the System requirements and Schedule for the model. Section seven is based on the design and algorithms and is classified into its sub sections for separate flowcharts and algorithms. Section eight is about the outputs of our implementation. Last section introduces References for the system.

II. Background Studies

The upcoming time is based on Blockchain Technology. It will be used in several places (as in for the e-voting, healthcare, Education and IOT devices. Access control, interoperability, provenance and data integrity are all issues which being faced by many technologies and many sectors. And Blockchain has that ability to solve all these issues in various sectors. Secondly, there are many problems of the existing manual voting system like: Time Consumption, too costly activities with more manual and paper work that can cause error while data entry, Loss of registration forms and short time provided to view the voter register.

The distributed nature of block chains helps to solve these issues if used properly. In block chain the advantages includes decentralization, transparent because of the transparent ledger and the efficiency towards fault tolerance. Because of the distributed structure of the blockchain, a Bitcoin electronic voting system reduces the risks involved with electronic voting and allows for a tamper-proof for the voting system. The blockchain uses hashing as its basic mechanism. Hashing is a technique or process of mapping keys, values into the hash table by using a hash function. It is done for faster access to elements. The efficiency of mapping depends on the efficiency of the hash function used.

A hash function takes a group of characters (called a key) and maps it to a value of a certain length (called a hash value or hash). The hash value is representative of the original string of characters, but is normally smaller than the original. In blockchain, decentralization refers to the transfer of control and decision-making from a centralized entity (individual, organization, or group thereof) to a distributed network. Decentralized networks strive to reduce the level of trust that participants must place in one another, and deter their ability to exert authority or control over one another in ways that degrade the functionality of the network. Smart contracts are simple programs which runs using blockchain with some conditions with the objective to automate some task. Smart contracts uses Ethereum for its working functionality.

This study discusses the recording of voting data using blockchain technology. On the study towards the implementation of Smart Contracts, few systems has shown it's efficiency. In voting, Ethereum Blockchain is implemented to create this voting system[6]. Various cryptographic hash functions and recalls merkle damgard security properties of iterated hash functions focuses on SHA-256 implementation and inner part architecture has proved less area utilization and improved security[4]. A network Reliable, safe, flexible system using blockchain was also proposed with the capabilities to support real-time services[7]. Another study explains about the IOT devices facing a challenge of data tampering which is recovered or solved by the blockchain technology and hashing algorithms[8].

III. Proposed System Design

A blockchain-based electronic voting system requires a wholly distributed voting infrastructure. Electronic voting based on blockchain will only work where the online voting system if fully controlled by no single body, not even the government. The flow process of the proposed system is given in Figure 1.

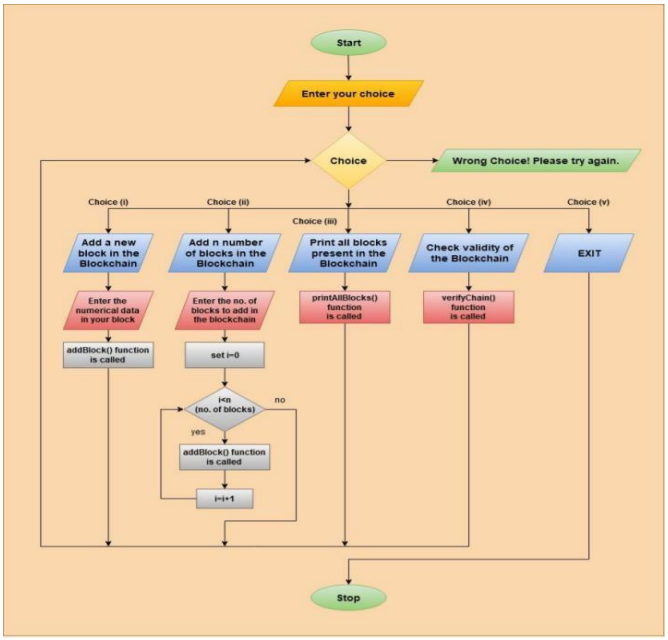

Figure 1. Process Flow

The algorithm followed for the proposed system is given below. The choice to perform operation in blockchain including is provided. Add a new block in Blockchain, add 'n' numbers of blocks in Blockchain, print all the blocks present in the Blockchain, check validity of the Blockchain and Exit (Terminate the code). Depending upon the various choices entered the input are provided and call addBlock() user define function and the functions shown in figure 2. Input the number of blocks that you want to add in Blockchain and iterate a for loop (number of blocks that you want to add) times. In every iteration call addBlock() user define function and after for loop iterations. In the choice of printing all blocks call printAllBlocks() user define function than goto for loop iterations. Call verifyChain() user define function than goto for loop iterations and last call and terminate the program as per the

exit choice. In add block/blocks function's separate algorithm after it receive block data by user it check's if Blockchain head is empty then create first block i.e. Genesis Block else Add a new block in existing Blockchain. Calculate the hash value for this particular block and store the block data in particular block. Manage addresses of blocks (nodes) according to concept of linked list and stop. During the Print blocks function algorithm it iterate every block in Blockchain (nodes in linked list) and call printBlock() function as shown in figure 3. Check if the head of the block is Null or not. If null the blockchain structure is empty else print parameter to every block. <<optional>>. Print all the blocks parameters (Block Hash Value, Previous block hash value, Block data, address of next block).

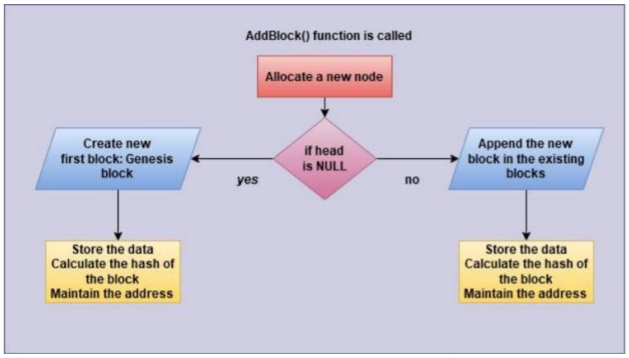

Figure 2. Add block/blocks function

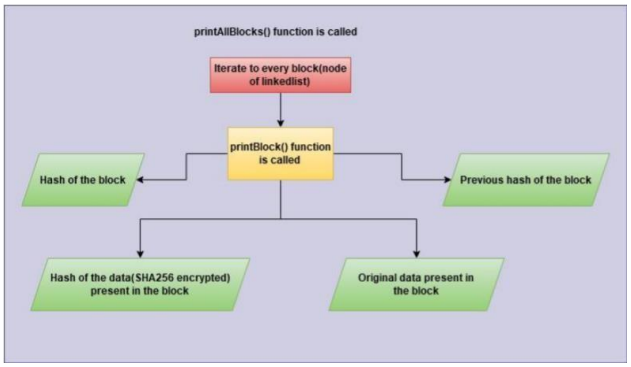

Figure 3. Print blocks function

During the verification of blocks algorithm works by comparing if head is equal to Null. If yes, print "Hey, Blockchain is empty! Please try after adding some block" and if no set two counter variable count=1 for iterating and flag=0 for verification. Since the current node is not Null, the serial number is printed, along with previous hash and current has in SHA256 form, if both are verified then the block is valid. If the block is not valid flag turns 1 and verification is denied. If flag stays 0 for all the blocks validation, then the verification is confirmed as shown in figure 4.

The proposed system follows the smart contract as given in figure 5. The first task is to **c**reate a contract named Election. Have to structure types named candidate and votes, as a voter to vote for a candidate. These structures have attributes as name, votecount, autho, voted after which the mapping of voters has to be done as public voters for the election followed with setting a collection and counter parameter for vote. We will have certain function for each working as listed below.

- add candidate - for candidate in election
- getNum candidate - for number of candidates
- authorize - for authorizing a voter
- vote - for voting in authorized manner and count of votes for a candidate
- end - to complete the election process

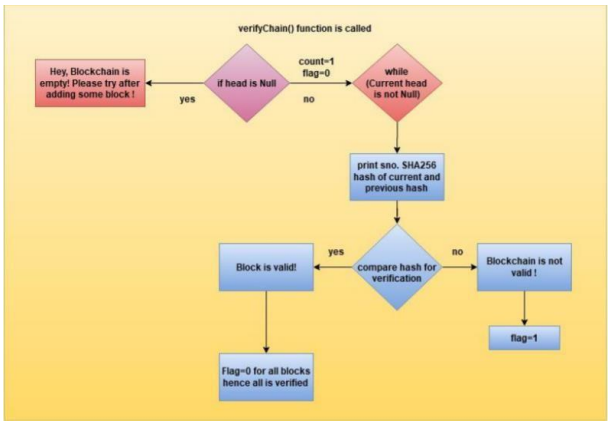

Figure 4. Verification of blocks function

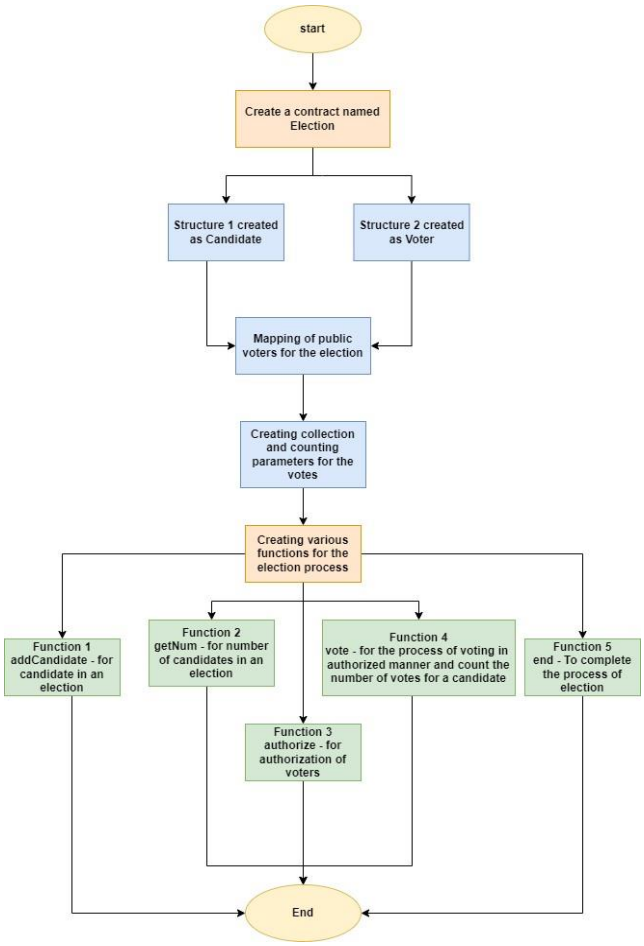

Figure 5. Smart Contract flowchart

IV. System Implementation

The system was implemented in command line user interface with i3/i5/i7 processor with 8 GB RAM, 1 TB Hard-disk. The software used for the implementation includes gcc compiler for C programming/coding platform or IDE, Remix.js online

compiler for Ethereum smart contacts using windows operating system. LevelDB database is used with openssl/Crypto.h library. Basic performance extended to Industry level model were involved in checking the various parameters of SHA256 hashing, block miners and avoidance of hash collision. After running the code, we get various options for the user-input.

1. Add a new block in Blockchain
2. Add n numbers of blocks in Blockchain
3. Print all blocks present in Blockchain
4. Check validity of Blockchain
5. Exit (Terminate the code)

The figure 6 shows the screenshots of the proposed system interface and the smart contract. The smart contact is performed by deploying the election, adding candidates and casting vote after authorizing the account.

Figure 6. Interface Output Screenshots

V. CONCLUSION

Our blockchain model implementation in C and solidity language is successful. In this project implementation of structures, pointers, linked lists, user define functions, control and jumping statements was initiated. For further project work file handling and a user-friendly GUI concept will be applied in our future work. We know that since the starting of the era when blockchain technology was launched via bitcoin, it was evolving into a general-purpose technology with use cases in many industries. The objective of the study behind this model was to identify the blockchain technology use cases in smart voting, the example applications that have been developed for these use cases, the challenges and limitations of the blockchain-based smart contract applications, the current approaches employed in developing these applications and areas for future research. Since in our model we can create a block, add into the block and check the validity of the block. This will improve in maintaining access control, scalability and the content or transactions information stays secure. Further research is also needed to supplement ongoing efforts to address the challenges of better scalability, latency, interoperability, security and privacy in relation to the use of blockchain technology in e-voting.

REFERENCES

[1] S. Al-Megren *et al.*, "Blockchain Use Cases in Digital Sectors: A Review of the Literature," *Proc. - IEEE 2018 Int. Congr. Cybermatics 2018 IEEE Conf. Internet Things, Green Comput. Commun. Cyber, Phys. Soc. Comput. Smart Data, Blockchain, Comput. Inf. Technol. iThings/Gree*, no. July, pp. 1417–1424, 2018, doi: 10.1109/Cybermatics_2018.2018.00242.

[2] D. B. Rawat, V. Chaudhary, and R. Doku, "Blockchain Technology: Emerging Applications and Use Cases for Secure and Trustworthy Smart Systems," *J. Cybersecurity Priv.*, vol. 1, no. 1, pp. 4–18, 2020, doi: 10.3390/jcp1010002.

[3] M. Niranjanamurthy, B. N. Nithya, and S. Jagannatha, "Analysis of Blockchain technology: pros, cons and SWOT," *Cluster Comput.*, vol. 22, no. 2, pp. 14743–14757, 2019, doi: 10.1007/s10586-018-2387-5.

[4] A. Gowthaman and M. Sumathi, "Performance study of enhanced SHA-256 algorithm," *Int. J. Appl. Eng. Res.*, vol. 10, no. 4, pp. 10921–10932, 2015.

[5] I. A. Omar, R. Jayaraman, K. Salah, I. Yaqoob, and S. Ellahham, "Applications of Blockchain Technology in Clinical Trials: Review and Open Challenges," *Arab. J. Sci. Eng.*, vol. 46, no. 4, pp. 3001–3015, 2021, doi: 10.1007/s13369-020-04989-3.

[6] A. Susanto, "Implementation of Smart Contracts Ethereum Blockchain in Web-Based Electronic Voting (e-voting)," *J. Transform.*, vol. 18, no. 1, p. 56, 2020, doi: 10.26623/transformatika.v18i1.1779.

[7] A. M. Al-Madani, A. T. Gaikwad, V. Mahale, and Z. A. T. Ahmed, "Decentralized E-voting system based on Smart Contract by using Blockchain Technology," *Proc. 2020 Int. Conf. Smart Innov. Des. Environ. Manag. Plan. Comput. ICSIDEMPC 2020*, pp. 176–180, 2020, doi: 10.1109/ICSIDEMPC49020.2020.9299581.

[8] D. Li, W. Peng, W. Deng, and F. Gai, "A blockchain-based authentication and security mechanism for IoT," *Proc. - Int. Conf. Comput. Commun. Networks, ICCCN*, vol. 2018-July, 2018, doi: 10.1109/ICCCN.2018.8487449.