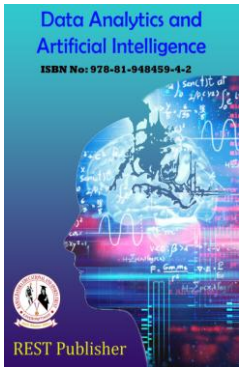
Data Analytics and Artificial Intelligence

Vol: 3(2), 2023

REST Publisher; ISBN: 978-81-948459-4-2

Website: http://restpublisher.com/book-series/daai/

DOI: https://doi.org/10.46632/daai/3/2/16



Block Chain for IOT Security Using Consensus Algorithms

***P. Kalpana, I. Anusha Prem**

St. Joseph's College of Arts Science for Women, Hosur, Tamilnadu, India

*Corresponding Author Email: kalpana912001@gmail.com

Abstract. *The first distributed recordkeeping system with a built-in trust structure is the block chain. It creates a dependable architecture for decentralized control through information redundancy across multiple nodes. Based on this, this study suggests a minimal block chain-based IoT information exchange security framework. The framework uses a double-chain approach that combines the data block chain and the transaction block chain. Distributed storage and tamper-proof data are implemented in the data block chain, and the consensus process is improved using the improved practical Byzantine fault-tolerant (PBFT) mechanism. Data registration efficiency, resource and data transfers, and privacy protection are all enhanced by better partial blind signature-based algorithms in the transaction block chain. This article focuses on how well the consensus algorithms employed in a block chain system for the Internet of Things perform (IoT). Such systems' time requirements to accomplish. Consensus ought to be minimal. The three most popular consensus algorithms—modified proof of work, realistic byzantine fault tolerance, and binary consensus—are assessed under various conditions, including mote type, number of participating nodes, and radio propagation model. To enable an IoT node to switch between different consensus algorithms, a comprehensive solution is put forward. The Contiki IoT operating system simulations display strong performance (time to achieve consensus less than seconds)*

Keywords: *Internet of Things, block chain, consensus algorithm*

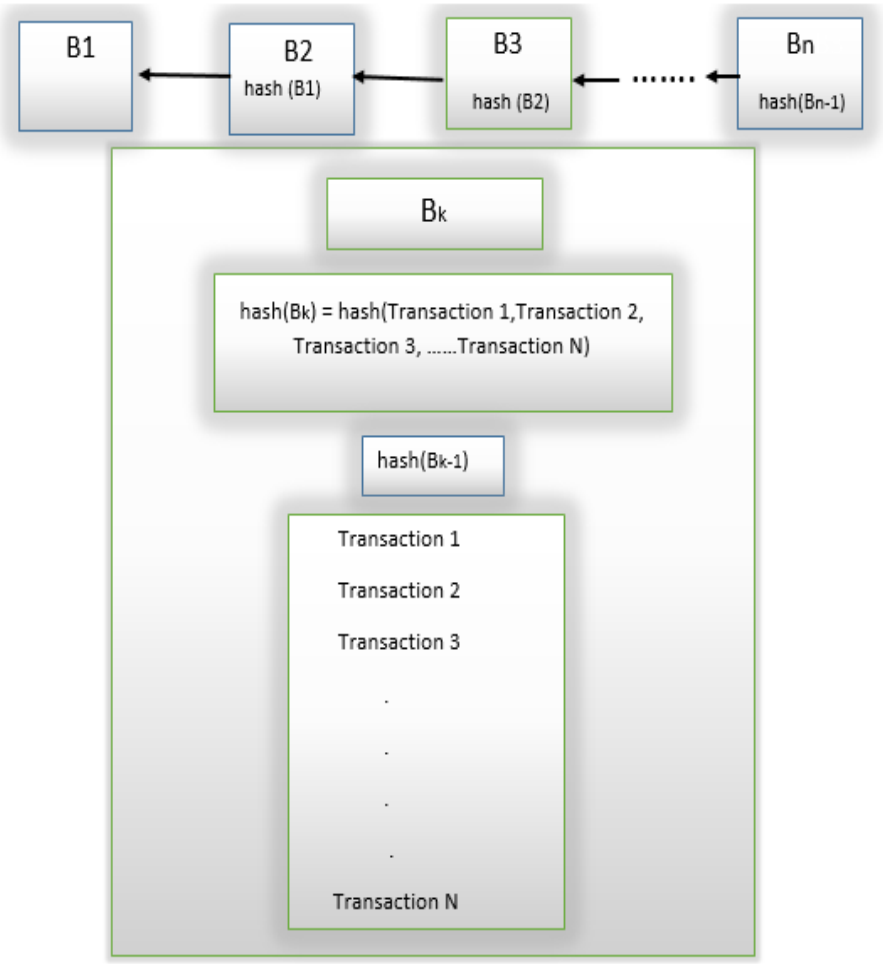
1. INTRODUCTION

The Internet of Things is now expanding steadily [1]. Smart cities, intelligent homes, and other applications require communication between many nodes, which are often composed of microcontrollers with low to medium processing capacity and little memory. The majority of Internet of Things (IoT) systems have a server-client communication model, in which servers with lots of processing power and storage are used to identify, authenticate, and connect objects. The devices are linked over the Internet and each has an own identification. Due to the massive increase in IoT device numbers, this communication paradigm will no longer be used in the near future. The cost of centralized solutions increased since they require ongoing maintenance and a large number of servers and networking devices to maintain connectivity across billions of devices. Block chain technology integrates encryption, peer-to-peer transmission, consensus, distributed storage and other technologies. Its potential application value has become a hot topic of discussion among many international organizations and national governments, attracting a wide range of research and development interests in industry [1–4]. At present, block chain technology has been initially applied to many fields such as the IoT, digital asset trading, and supply chain management, which may trigger a new round of technological innovation and service mode change [5–7]. The core problem of the block chain solution is to establish an ecosystem that satisfies the user's trust conditions in the case of information disclosure.

2. BLOCKCHAIN TECHNOLOGY FOR THE INTERNET OF THINGS

Block chain technology might offer a more effective remedy. The block chain allows an IoT network to run without a centralized node and maintains an immutable record of device history. Block chain technology for IoT will lower costs and unpredictability of functioning devices [3]. The "transactions" Created over a specific period of time are contained in a block. Except for the first block, the data in each block is encrypted using a hash function and contains a reference to the block before it. This is seen in Figure 1.Because updating a block

alters its hash and breaks references in a block chain, the earlier blocks cannot be changed. In a block chain system, there are two different kinds of nodes: nodes who want to add new blocks to the chain, and nodes that receive new blocks and use a consensus algorithm to add them to the chain.

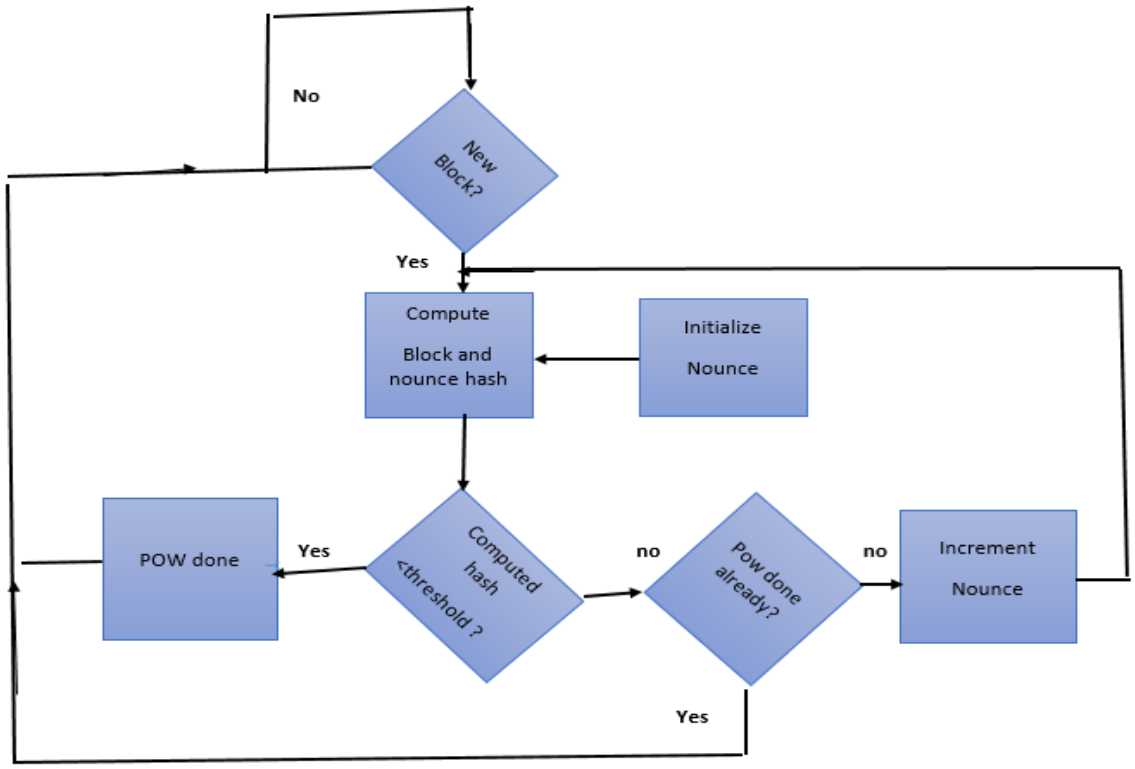


The Block chain

3. CONSENSUS ALGORITHMS

We'll examine three consensus algorithms: the traditional Byzantine Fault Tolerance (BFT) algorithm, the proof of work (PoW) technique and a new form of PoW, and the binary consensus (BC) approach. The following characteristics of each of these consensus algorithms will be assessed: computation time (including all processing steps and transmission time), the number of messages exchanged in the network to implement the algorithm, and the average time interval in which the consensus is reached while taking into account two radio propagation models that increase the number of messages in the network because of the need for retransmissions.

Proof of Work: The proof of work consensus algorithm [3, 4] is illustrated in Figure 2.



The POW algorithm: The receiving nodes (referred to as miners) compute a hash function of the block data and an announcement when a node wants to create a new block. The calculated hash value must be less than an established value (this value indicates the work's complexity). If this criterion is false, the hash function is recalculated using the new nounce and the nounce is increased. The block will have a proof of work when a miner has finished, and other network nodes will verify it. When compared to POW calculations, the hash function has the advantage of making verification quick. The complexity of solving the cryptographic problem with a target beginning with k zeroes is (16) k O, but The complexity of checking a proposed block is only O(1).In IoT systems, a lower proof of work complexity is required to gain agreement more quickly. The block is added to each node's blockchain copy if the majority of these nodes successfully validate the POW. The majority is provided by 1.N nodes, where 1 0.66 1... And N is the total number of nodes that are involved in the consensus process. The entire system is since more than one-third of the system's total computing power is required to install a bogus block. The POW method can be distilled into the following four points [4]: - A miner will be required to wait a limited amount of blocks. Each miner computes a hash function of the product of concatenating its unique address identifier and its public key (denoted as MIN - Miner Identification Number) after its last proposed block and before beginning a new POW job (excluding the first block).Create a block score using the formula: 256 score H MIN = 2 | | or 256 score H MIN = 2 / | |, where H is either the previous block's hash value or the sum of the hash values of the previous K blocks.

4. BYZANTINE FAULT TOLERANCE IN PRACTISE

Workflows for the Byzantine Fault Tolerance algorithm [5] the use of synchronous systems and multiple voting cycles makes them occasionally unfeasible. Practical Byzantine Fault Tolerance (PBFT) is one such implementation where nodes share sequence nodes (also known as view or group). A leader (or principal node) for the group must be chosen through an election mechanism. The remaining nodes are known as replicas. A new leader will be chosen and a new group will be formed if the primary node fails. The technique is secure because the nodes don't alter their states (decisions), and since the requests are completely ordered. Every request will result in a replay for the client, making the algorithm active.

- The client node communicates a request to the main node. The message is verified by the main node, and supplies the request message with an identification number.
- All replicas receive "pre-prepare" messages from the primary node, which are used to validate client request messages and obtain identification numbers.

- The clones reach an agreement on the overall sequencing of these messages and send "prepare" messages to every other replica.
- A "commit" message was disseminated by the replicas. They acknowledged receiving the client request and decided on an order total.
- The client receives a "replay" message from every working replica (including the primary).
- After waiting for two legitimate messages, the client takes a decision. Figure 3 displays the algorithm.

5. IMPLEMENTATIONAND PERFORMANCE EVALUATION

The Contiki operating systems for the Internet of Things were used to implement these algorithms. As depicted in Figure 5, each node runs a control process that incorporates the various consensus procedures mentioned above. It is a solid assumption that there shouldn't be more than 25 nodes taking part in the consensus. The three above-mentioned consensus methods for two radio media (UDGM, Unit Disk Graph Medium, and MRM, Multi-path Ray Tracer Medium [7]) were evaluated in terms of computational effort. A multicast transmitter thread (MC TX Process) and a multicast reception thread (MC RX Process) must be generated in order to implement the control process graph.

6. CONCLUSION

The study assesses the implementation's computational effort. There are three consensus algorithms utilised in block chain systems for the Internet of Things, and their time to consensus should be relatively short. A comprehensive approach is put forth that automatically adjusts a node to the network's consensus mechanism. According to the simulation findings, the suggested implementation performs well (the computing time is less than seconds) and is resilient to a variety of situations, including the kind of mote, the number of nodes utilised in the consensus procedure, and radio propagation. Both Contiki's optimised primitives and the specification and arrangement of the suggested processes are responsible for the performance.

REFERENCES

[1]. Wortmann, Felix & Flüchter, Kristina. (2015). Internet of Things. Business & Information Systems Engineering. 57. 221-224. 10.1007/s12599-015-0383-3.

[2]. Zibin Zheng; Shaoan Xie; Hongning Dai; Xiangping Chen; Huaimin Wang. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends, Big Data (Big Data Congress), 2017, Honolulu, HI, USA, DOI: 10.1109/BigDataCongress.2017.85

[3]. Zhang, Y. & Wen, The IoT electric business model: Using blockchain technology for the internet of things J. Peer-to-Peer Netw. Appl. (2017) 10: 983. https://doi.org/10.1007/s12083-016-04561

[4]. Castro, M.; Liskov, B. (2002). "Practical Byzantine Fault Tolerance and Proactive Recovery". ACM Transactions on Computer Systems. Association for Computing Machinery. 20 (4): 398–461, doi:10.1145/571637.571640

[5]. Abdaoui, Abderrazak & El-Fouly, Tarek. Distributed binary consensus algorithm in wireless sensor networks with faulty nodes. 2013 7th IEEE GCC Conference and Exhibition, GCC 2013. 495-500. 10.1109/IEEEGCC.2013.6705829

[6]. Thomson, Craig & Romdhani, Imed & Al-Dubai, Ahmed & Qasem, Mamoun & Ghaleb, Baraq & Wadhaj, Isam. (2016). Cooja Simulator Manual. 10.13140/RG.2.1.4274.8408.

[7]. D. Li, Z. Cai, L. Deng, Information security model of block chain based on intrusion sensing in the IoT environment, Cluster Comput. 8 (Z1) (2018) 1–18.

[8]. P.K. Sharma, S. Singh, Y.S. Jong, and DistBlockNet: A distributed blockchainsbased secure SDN architecture for IoT networks, IEEE Commun. Mag. 55 (9) (2017) 78–85.

[9]. A. Ouaddah, A. About Elkalam, A. Ait Ouahman, FairAccess: a new Blockchain-based access control framework for the IoT, Secur. Commun. Netw. 9 (18) (2016) 5943–5964.

[10]. W. He, Computational neuroscience applied in surface roughness fiber optic sensor, Trans. Neurosci. 10 (1) (2019) 70–75, http://dx.doi.org/10. 1515/tnsci-2019-0012.

[11]. M.H. Miraz, M. Ali, Blockchain enabled enhanced iot ecosystem security, Soc. Sci. Electron. Publ. 9 (3) (2018) 38–46.

