Available online at http://ijim.srbiau.ac.ir/
Int. J. Industrial Mathematics (ISSN 2008-5621)
Vol. 14, No. 3, 2022 Article ID IJIM-1574, 13 pages
DOI: http://dx.doi.org/10.30495/ijim.2022.65634.1574
Research Article
Science and Research Branch (IAU)

The Application of Private Blockchain to Increase Security and Privacy in Internet of Things (IoT)

R. Mahmoudie *, S. Parsa †‡, A. M. Rahmani §

Received Date: 2021-08-24 Revised Date: 2022-03-02 Accepted Date: 2022-03-17

Abstract

BlockChain (BC) has the potential to address the Internet of Things' security and privacy issues (IoT). BC, on the other hand, is computationally expensive, has limited scalability, and incurs significant bandwidth overheads and delays, making it unsuitable for use in the IoT. In this study, we have proposed a method that optimizes BC for use in IoT environment. We have presented a hierarchical structure that uses a private BC to increase scalability, reduce network overhead and delay. In the proposed method, devices with high-level resources build a network referred to as the overlay network on devices with low-level resources. The members of the overlay network manage the BC. We used trust technique and voting from direct neighbors to reduce network traffic and overhead costs. The use of factors such as trust, encouragement and penalty of block managers in the overlay network ensures accurate transactions in IoT. We used a new authentication algorithm for authenticating block managers in the overlay network. The simulation results show that the proposed algorithm reduces packet overhead and delay in service delivery and increases the scalability of the BC in comparison to the system that uses the base BC. Furthermore, because in the proposed algorithm, the number of effective block managers in voting is limited to direct neighbors, the average time to confirm a block is significantly reduced.

Keywords : Scalability; Security; Blockchain; Internet of Things; Overlay Network; Block Manager.

1 Introduction

BC is distributed data storage and sharing system that uses an immutable timestamp ledger of blocks [1]. BC keeps track of transactions in a distributed digital ledger that is shared across all nodes. New transactions are validated and confirmed by other nodes in the network, obviating the need for a centralized authority. The process of adding a new block to the BC, known as mining, requires solving a computationally challenging, difficult-to-solve, and easy-to-verify puzzle. This problem is at the heart of a trustless consensus technique for untrusted nodes. The computing resources required to participate

*Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.
†Corresponding author. parsa@iust.ac.ir, Tel:+98(912)1000188.
‡Department of Computer Engineering, Iran University of Science and Technology, Narmak, Tehran, Iran.
§Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.

in the consensus procedure might be substantial, limiting the amount of blocks that a node can mine and so providing protection against malicious block mining.

To solve the puzzle, you must use a mechanism that introduces unpredictability among the nodes that want to connect [2]. One of the major issues standing as a barrier to adopting various IoT products is the security and privacy challenges. The growth of IoT devices creates new services and applications, but at the same time, it creates several security vulnerabilities that became more apparent. Manufacturers of IoT devices are not considering security in their priorities [3]. With low public awareness about security and privacy, IoT devices could lead to severe problems that could literally lead to losing our lives. The governments should encourage manufacturers of IoT devices to adopt new security measures in their products. Also, manufacturers should employ the concept of security by design to implement built-in security algorithms within their products to ensure minimum security and safety for various consumers.

IoT has acquired significant recognition and appeal as the major standard for Low-Power Lossy Networks (LLNs) with restricted resources, thanks to the increasing rise of smart gadgets and high-speed networks. It denotes a network in which *things*, or embedded devices with sensors, are linked via a private or public network [4, 5]. The gadgets in IoT can be operated remotely to execute the intended purpose. The information is subsequently shared across the devices via the network, which uses industry-standard communication protocols. The smart linked devices, often known as *things*, range in size from small wearables to massive machineries, all of which incorporate sensor chips. There is a demand for worldwide IoT access control systems that are also extremely secure and reliable. As a result, various IoT standards have established new ways to address these restrictions. Constrained protocols, such as the Constrained Application Protocol (CoAP) [6] are used to create these solutions. They use a server to centralize the management of IoT devices. However, a fundamental flaw in all existing IoT commercial solutions is that they are vulnerable to a single point of failure, which prevents IoT systems from scaling. Furthermore, centralized architectures are intended for systems in which controlled devices are stationary and belong to the same management community for the duration of their lives.

However, in the IoT, some of the assumptions of centralized designs are incorrect [7]. Many IoT devices will be mobile, as in IoT situations involving vehicle-to-vehicle communication [8]. They may be managed by several managers, as in supply chain scenarios [9], and they may belong to many management domains over the course of their lives. Authors in [10] have previously developed a blockchain-based approach for managing access to resources in IoT devices in order to give a more flexible solution. Managers are responsible for administering the access control permissions of a group of IoT devices in our system. The policy is kept on the blockchain, and the management interactions with the various IoT devices are also done on it. Current IoT ecosystems are based on centralized brokered communication models, in which all devices are recognized, verified, and connected via cloud servers. Because billions of devices are connected, this concept is unlikely to scale. Furthermore, cloud servers will continue to be a bottleneck and a point of failure for the entire network [11]. BC has been employed in a variety of non-monetary purposes, such as proving location [12]. In 2013, Ethereum [13], a new open-source BC-based platform, was suggested to support smart contracts, which are computer programs that enforce a set of rules. BlockCharge [14] is a BC-based electric car charging platform. It uses Bitcoin as the underlying payment mechanism, hence it inherits Bitcoin's high level of secrecy. Most IoT devices have limited resources, including bandwidth and memory, which is incompatible with the requirements of complex security solutions [11]. The multiple advantages of BC technology make it an appealing answer to the Internet of Things' challenges. Existing BC examples, on the other hand, cannot simply be applied to the Internet of Things for the figure (1).

1.1 Blockchain Solutions for IoT Security

There are several security threats to the Internet of Things. However, the integration of the BC into the IoT can be a solution to overcome some of these security threats. IoT consists of heterogeneous devices with latent sensors connected via a network [15]. Most of these devices have features such as low power, small memory and limited processing capacity, so the most important challenges of the IoT can be summarized in figure 2. We used Cooja and *Omnet++* to run comprehensive simulations to evaluate important performance characteristics including latency, processing time, and cyber-attack resilience. Our findings support different design decisions and show that the planned optimizations are effective. In this article, we have tried to control the authentication of objects in the IoT and block managers and improve the level of security in the IoT by using a private BC that has been implemented at the level of the overlay network. To achieve this goal, we used the authentication algorithm of block managers. The key contributions of this paper are summarized below:

1. Implementing a private *BC* on IoT.
2. Using the coverage network to optimally manage resources and increase the level of security in the IoT.
3. Authentication of objects under the control of each of the block managers.
4. Authentication of block managers in the overlay network.

The rest of the paper is organized as follows. Section 2 presents a literature review on IoT security and BC applications. The proposed method and an overview of transactions are discussed in Section 3. Detailed security analysis and performance evaluations are presented in Section 4. Section 5 discusses further aspects of the proposed method and finally Section 6 concludes the paper.

2 Literature Review

Smart grids, smart cities [16] and health management [17] are only a few of the applications for the IoT. However, in the midst of people's private lives, the increasingly undetectable, dense, and pervasive collecting, processing, and dissemination of data raises severe security and privacy problems. The lack of central control, heterogeneity in device resources, various attack surfaces, context-specific risks, and scalability are all inherent elements of IoT that endanger its security and privacy issues.

This paper suggests the BC technology underpinning Bitcoin is flawed [18], and the first cryptocurrency system can provide an effective solution to IoT privacy and security. The main source of BC security is a cryptographic puzzle known as Proof of Work (POW), which is used to append (mine) new blocks to the BC. BC also offers a high level of privacy by using a changeable Public Key (PK) as the users identity. BC has been adopted for a number of non-monetary applications, e.g. proof of location [19], distributed storage systems [20], and health care data [21]. These distinguishing characteristics of BC make it appealing for enabling distributed privacy and security in the IoT. The application of BC to IoT, on the other hand, is not straightforward. Several major obstacles must be overcome.

a) POW's use necessitates a lot of resources.

b) Scalability concerns arise from the necessity to reach consensus among miners.

c) POW causes a lot of delays

Due to the vast majority of devices' limited resource capabilities, massive volume, heterogeneity among devices, and lack of standards, IoT security is difficult. Furthermore, many of these IoT gadgets capture and share vast amounts of data from our private lives, raising serious privacy concerns. The authors of [22] designed distinct privacy zones for different categories of data to protect users' privacy.

In [23], the authors showed that a wide range of IoT devices lack basic security precautions.

The authors proposed a Security Management Provider that is in charge of limiting data and device access through fixed or dynamic content-based restrictions. However, the issue of preserving user privacy while disclosing personal information is not addressed. Authors in [24] have fully analyzed the IoT security. In their paper [24], the authors developed a multi-layered *BC* architecture for receiving data from IoT devices and sharing it with organizations and individuals. The suggested architecture has three primary components: a data management protocol, a data storage system, and a messaging service.

Intel has released Proof of Elapsed Time (*PoET*), a new consensus *BC* technique that interacts with Hyperledger [25]. *PoET* is a consensus mechanism that runs on Intel processors in a trusted execution environment (*TEE*). A node must wait as long as a random time from a trustworthy and truly random range before storing a block in a *BC*.

A solution for protecting users' privacy in the smart home was given in [26]. The concept was implemented using three separate components. The data collector module gathers data from users in the smart home and transfers it to the data receiver module, which divides it into two data sets. To protect privacy, the result-provider module checks end-user access to the data. This strategy ensures that only the real user has access to the data.

The authors of [2] employ an overlay network to answer the block unscalability problem. The nodes are clustered by the clustering method. In this technique, each group has a cluster head, and only the cluster heads are responsible for keeping the *BC* up-to-date. This method helps to cut down on network traffic. The number of transactions grows as the number of nodes grows; nevertheless, the number of confirmed transactions grows as well. The Lightweight Scalable *BC* (LSB) employs a *BC* that differs from IOTA's *DAG*. As a result, the *LSB* benefits from the inherent advantages of *BC*, such as dependability and immutability.

In order to manage smart meter data, smart contracts have also been used in [27]. Similarly, in [28] a blockchain-based system has been suggested to manage firmware updates of IoT devices. [29] makes use of *BC* to store access control data, as a data storage system in a multi-tier IoT architecture. In [30], *BC* and smart contracts are employed to secure authorization requests to IoT resources. The afore-mentioned articles exploit *BC* to either execute smart contracts or perform application specific tasks, but not to decentralize IoT systems and achieve autonomous application execution.

Figure 1: The most important challenges of the *BC*.

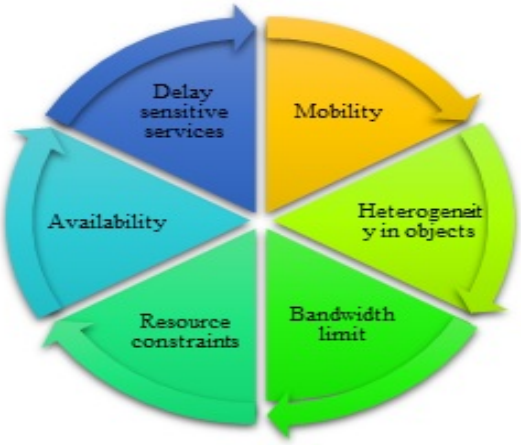

Figure 2: The most important challenges of the IOT.

3 The Proposed Method

In this article, we have used exclusive private *BC* and voting technique from direct neighbors. In the proposed method, IoT devices are divided into two layers. The first layer includes all low-level resource devices and the second layer is an overlay network with high-level resources which are called block managers. In the second layer, in order to detect the intrusion signs in each block manager, the neighboring block managers (block managers in the row and column of the block manager in question) participate in voting in a

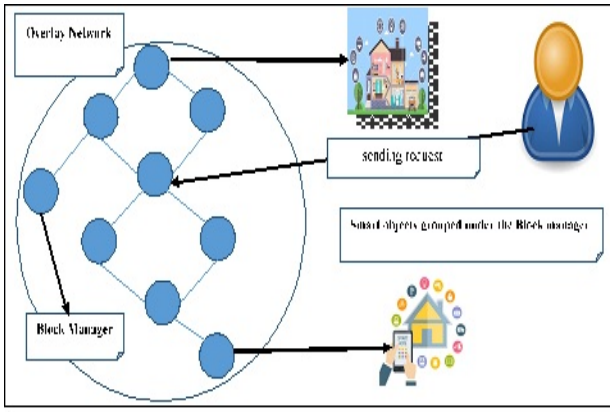

Figure 3: The arrangement of block managers in the overlay network.

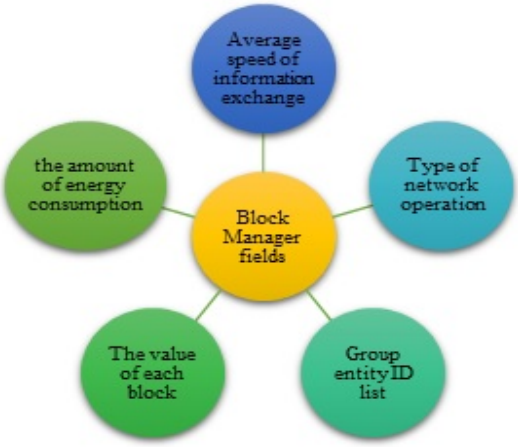

Figure 4: The control fields of each block manager

two-dimensional form. For this purpose, a block manager is considered as the manager of several devices in the first layer in order to increase the speed of detecting attacks, to reduce the overhead and network delay. The overall structure of the IoT, presented in this article, consists of two layers as shown in Figure 3. In the overlay network, devices with high-level resources are used to ensure scalability. The nodes in the overlay network are organized as block managers and only these block managers are responsible for BC management. A block manager contains the transactions of the members of its respective subset; simply, the transactions of a member will be processed at a specific time in the respective block manager affiliate relations. Thus, each member (an IoT device) must belong to a subset of a block manager. If a device experiences an excessive delay in receiving transactions from the relevant block manager, it can easily change its set. In addition, if a service provider or a requester sends a request to a block manager, the block manager will first verify the service providers public key (PK). If the PK is allowed to access, the information would be encrypted by the service providers PK and sent to the service provider. On the other hand, when a block manager receives a request from a service provider, if the request is not a subset of its services, the request will be sent to the block managers neighbors and will be registered in the block managers submission list.

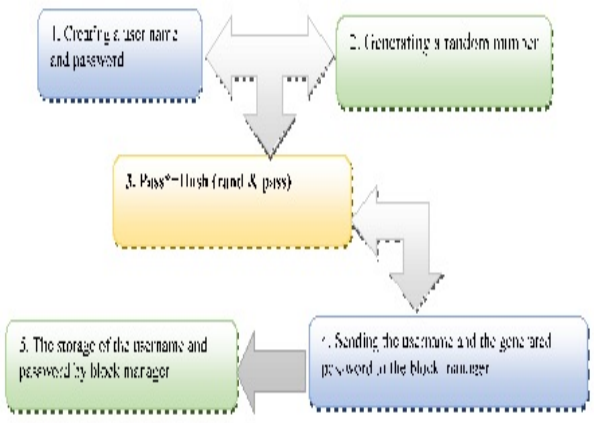

Figure 5: Registration algorithm steps

Figure 6: Common IoT attacks

3.1 Blockchain Structure in the Proposed Method

A block manager is an entity responsible for managing the BC. This management includes production, verification, and the storage of transactions block. Each node considered as the block manager will be identified with a public key. The overlay network may potentially contain a large number of block managers. In addition, each

Figure 7: Evaluation criteria of the proposed algorithm

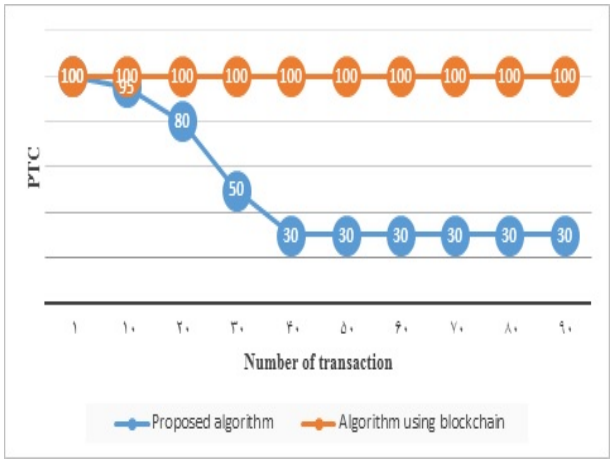

Figure 8: EvaluateThe percentage of transactions that need to be confirmed

bock manager monitors the input and output (I/O) transactions executed by its subset members. Transactions generated by a block manager are encrypted and protected with asymmetric encryption, digital signatures and hashing functions. The control fields of each block manager are shown in Figure 4. Block managers in the overlay network use digital signatures and asymmetrical encryption methods to manage the block, so that each block manager has two private and public keys for encryption. These keys have a mathematical connection. The public key is shared to receive messages from other users. The private key will also be kept private by the block managers. In digital signature, the data hash is encrypted by a private key and decrypted from the receiver side by the sender's public key, and finally compared to the data hash to verify the sender's identity and the integrity of the information. Generating a pair of private and public key is like creating an account on the block chain, without having to register somewhere. Every transaction executed on the *BC* is signed by the sender's private key.

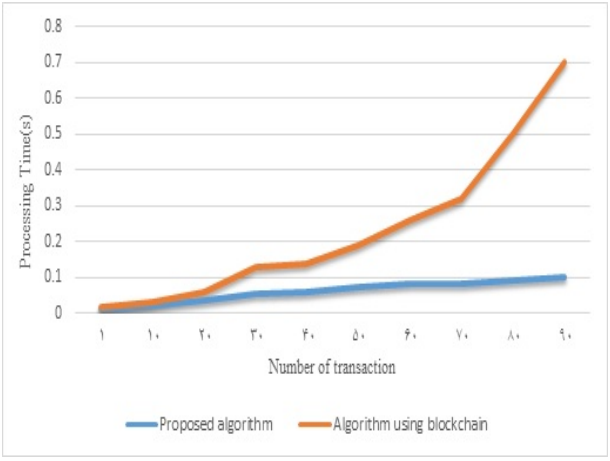

Figure 9: Evaluating the average processing time to validate a new block

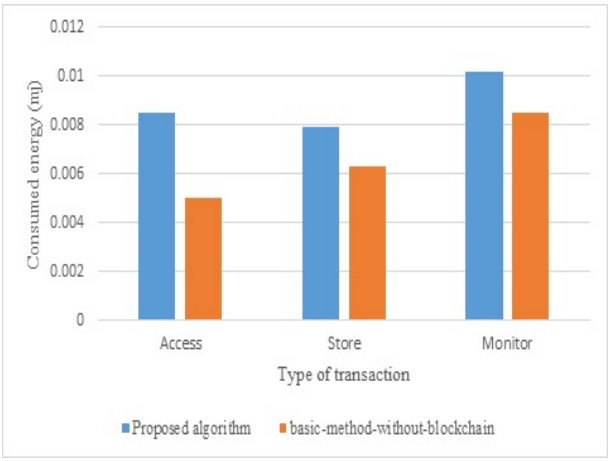

Figure 10: Evaluation of energy consumption

3.2 The Structure of the Proposed Overlay Network

Each block manager can be adjacent to N neighboring block managers based on the geographical location of the other block managers. As shown in Figure 3, the proposed network structure includes two-dimensional block managers. In this case, for each block manager, a maximum of four direct neighbors are considered. The *BC* applied in the overlay network is of exclusive type, so there is no need for the *BC* to be identical. Using this technique also reduces synchronization costs.

3.3 Steps of Implementing the Proposed Algorithm

Phase 1: Registration.

Phase 2: Recording the Block Managers Authorized Transactions at the Overlay Network.

Phase 3: Encryption of Block Managers Authorized Transactions.

Phase 4: Authentication of Block Managers.

Phase 5: Weighting Block Managers to Determine Voting Values.

Phase 6: The Voting Process to Determine the Validity of the Request or to Detect Intrusion.

Phase 7: Encouraging and Punishing Block Managers.

Phase 1: Registration

In this step, to authenticate each object in the corresponding block manager, we have used the algorithm shown in Figure 5.

Phase 2: Recording the Block Managers Authorized Transactions at the Overlay Network

To implement this phase, each block manager must perform the authorized transactions related to its subset and all neighboring managers must be aware of these transactions; therefore, each block manager's authorized transactions are stored in an array called $T[n, m]$. Actually, every block manager is responsible for carrying out a specific transaction, and that transaction has already been set out.

Phase 3: Encryption of Block Managers Authorized Transactions

Transactions created by each block manager are encrypted using light-weight one-way cryptographic hash algorithm ($LOCHA$). $LOCHA$ produces a hash with fixed and relatively small length. $LOCHA$ meets all the properties of one-way hash (basic properties such as collision resistance and pre-image resistance) [31].

Phase 4: Block Managers Authentication

We used the algorithm below to authenticate the block managers at the overlay network. The service requester first sends its request to its own direct block manager. This request includes the transaction number, the requester's public key and the block manager's public key. The block manager checks the received request. If the requested public key is located in the same block managers subset, the access permission is controlled. Otherwise, the request will be forwarded to neighboring managers for checking and responding. Finally, the KI key will be chosen as the session key. If a block manager receives multiple unsuccessful access requests from a specific PK, it can block that PK and reject any subsequent requests.

Block Manager Authentication Algorithm

1. Sending the access request $(T1, PKI, K_J)$.
2. Rejecting the request in the absence of the managers PKI in the access list.
3. Accepting the request in the presence of PKI in the access list (through the neighbors voting algorithm).
4. Sending request to neighboring managers in the absence of the managers PKJ in the access list.
5. Generating a random number Y and calculating $F(X)$ using KJ and $Hash(pass*, Y)$ by the server manager.
6. Sending Y to the transaction requesting manager.
7. Z calculating by transaction requesting manager $Z = Hash(T1, PKI, F(0))$ Sending Z to server block manager.
8. Calculating $Z' = Hash(T1, PKI, F(0))$ If $Z = Z'$ the requester is authenticated.

9. $KI = Hash(F(0))$.

Phase 5: Weighting Block Managers to Determine Voting Values

The majority of proposed IoT solutions presume the presence of a trusted environment, which is not always the case. As a result, trust management is a prerequisite for network security solutions. Under the proposed method, each block manager maintains only the trust values of its neighbors. The trust value for each block manager is in the range of $[0, 1]$. At this stage, based on the importance and value of block managers, each block manager is assigned a weight according to the following formula. This weight will be considered as the trust value of each block manager to its neighboring block manger.
$Cost[n, m]$, the value of n block manager, is provided that it is adjacent to m block manager. $Cost[n, m]$ is received as the input parameter. Hence, each block manager has a factor called block manager cost. For instance, the number of successful operations divided by the total number of operations for each block manager is considered as the value of the block manager which is initially performed as an input parameter and will be calculated in the next steps from the following formula.
The formula (3.1) for calculating $costs[n, m]$ is as follows.

$$Cost[n, m] = \frac{Number\ of\ successful\ operations}{Total\ number\ of\ operations}. \tag{3.1}$$

$Sum[m]$ is the neighbors total value of m block manager. Now, to calculate the weight of each block manager for each neighbor, the following formula (Formula 3.2) is used. It should be noted that each block manager has one certain weight for each neighbor.
Given that each block manager has a specific value for each of the adjacent blocks, which is placed in $cost[n, m]$, so the sum of the values associated with that block is also calculated in the $Sum[m]$ matrix. The weight of block n will be calculated in the form of formula (3.2) as long as it is adjacent to block m.

$$Weight[n, m] = \frac{Cost[n, m]}{Sum[m]}. \tag{3.2}$$

3.4 Time-based Consensus Algorithm

The block manager that received the request creates a transaction for a legitimate request. To boost block security, we introduced a Time-based Consensus Algorithm that chooses a neighbor block manager at random and adds the created block into the block chain. This algorithm aids in reducing the number of duplicate blocks that may occur at the same time. Furthermore, the waiting duration in the overlay network is limited to twice the maximum point-to-point delay. Neighboring block managers wait a random period of time after a valid request is discovered.

Phase 6: The Voting Process to Determine the Validity of the Request or to Detect Intrusion

At this stage, the users request is examined. Neighboring block managers vote using matching user requests and encrypted code in the second phase. At this point, the eligible ones encrypt and send their vote. According to the aggregated homomorphic principle, the encrypted sum of information is equal to the sum of the encrypted information [32].Thus, using this feature, the block manager selected to count the votes determines the result by decrypting the sum of votes. After the calculation of the total value weight of the positive and negative votes, if the sum of positive votes exceeds the sum of negative votes, the block is allowed to change the situation. However, if the total negative vote is more than positive ones, it is not allowed to change the situation.

Phase 7: Encouraging and Punishing Block Managers

When a number of block managers have voted incorrectly, they should be punished, while block managers who voted correctly should be encouraged. As a result, the percentage p of the weight of the nodes that voted incorrectly is reduced

and added to the weights of the managers of the blocks that voted correctly. (p is a coefficient to control the rate of decline in confidence, which is a factor in the range $[0, 1]$).
Assuming it is $p = 10$, the weight of the block managers who voted incorrectly will be reduced by 10% due to the wrong vote and will be divided among the neighbors who voted correctly. This is considered a penalty variable. The manner in which the penalty amount is divided among the winning nodes is calculated using the following formula. To solve this challenge, formula (3.3) is used. The variable $Weight[i, j]$ is the weight of node i relative to node j.

$$Weight[i, j] = \frac{Weight[i]}{Weight[i] + Weight[j]},$$
$$Weight[i] = Weight[i] + Weight[i, j] * penalty. \tag{3.3}$$

4 Evaluation of Suggested Algorithms

This section assesses the suggested algorithm's quality and performance in the face of security threats. A malicious node is said to be capable of acting as a smart home device or an overlay network node. A rogue node has the ability to interrupt network connections, erase transactions, create bogus transactions and blocks, and edit or remove data stored on the network. As a result, a malicious person's goal in some cases can be said to be to prevent the user from having legitimate access to the services. Furthermore, a malicious user tries to gain access to the system by posing as a legitimate user.
We employ the trust method for neighboring nodes to alleviate the computational cost involved with confirming new blocks. According to the legitimacy of the new blocks they create, each block manager collects evidence from other nearby managers. As the network grows, more transactions may be added to the BC, hence the suggested method will improve performance. Block managers keep track of transactions. If an attacker tries to change a previously saved transaction, the next block's matching block hash is no longer compatible. As a result, a change attack can be identified in this manner.
In the proposed technique, a transaction has the following structure: An identity field and a pointer field to the prior transaction of the requesting node are evaluated for each transaction, ensuring that all transactions made by a requester are linked. We utilize the requester's and the requestee's public keys and transaction signatures to do this. Finally, we consider an output field that contains the total number of accurate transactions received and responded to by a requester, as well as the total number of transactions refused by the requestee from that requester. We can examine the requester's legitimacy based on this information.
A transaction has the following structure in the proposed technique: For each transaction, an identification field and a pointer field to the requesting node's previous transaction are assessed, ensuring that all of the requester's transactions are connected. The public keys and transaction signatures of the requester and the requestee are used to accomplish this. Finally, we consider an output field that comprises the total number of correct transactions received and replied to by a requester, as well as the total number of transactions that the requestee has denied from that requester. Based on this information, we can assess the requester's validity.
Based on common IOT attacks, we will examine the following attacks to evaluate the proposed method. We analyze the flexibility of the method presented in this study against any attack and the probability of an attack occurring according to the risk analysis criteria of the European Standards Institute ($ETSI$).

4.1 Evaluation

On IoT devices, the blockchain-based framework assures security and privacy. Simulators such as Cooja [33] and $Omnet^{++}$ [34] are utilized to test the suggested technique. We may pick, setup, and test the performance of sensors and actuators in the node since Cooja is ideal for assessing low-resource devices and has the capacity to execute numerous IoT protocols.

In peer-to-peer network analysis, the $Omnet^{++}$ emulator is also employed. We created an alternate scenario that performed transactions without encryption, hashing, or blockchain to compare the overhead of the approach reported in this research. In the implementation, we employ the $basic-method-without-blockchain$ method. In the simulation, we will utilize the $IPv6$ protocol and $WLAN$ (6LoWPAN) as the primary communication protocols. During the simulation, the number of nodes changes from 10 to 90. During this period, the simulation is performed numerous times and the results are provided on average. The following criteria were used to make the decision shown in Figure 7. To evaluate the proposed framework's performance, we run a simulation with the $Omnet^{++}$ emulator, concentrating on the overlay network. We simulate a 100-node network with 15 block managers to analyze the proposed algorithm's traffic and processing overload. The simulation takes 180 seconds to complete and generates 1150 transactions.

4.2 Evaluate the Percentage of Transactions that Need to be Confirmed (PTC)

Figure 8 depicts the percentage of transactions that must be confirmed (PTC). Because the block managers have yet to gain trust in one another, the processing time for both techniques is the same when they first start up. However, when more blocks are generated and validated over time, the block managers develop direct trust in one another. As a result, processing time is reduced as compared to the $basicalgorithmusingBC$, which verifies all transactions within the block. Furthermore, as the number of confirmed blocks grows, the number of transactions that need to be verified decreases as trust in other block managers grows. When 40 blocks are formed, the level of trust among block administrators reaches its pinnacle.

4.3 Evaluate the Average Processing Time on Block Managers to Validate New Blocks

Block Managers must authorize transactions within a block in the BC structure. The average time for validating blocks grows exponentially as the number of IoT devices grows. For IoT devices, delays caused by blocking validation are unacceptable. Although the number of devices in the IoT grows, the average time to approve a block decreases and remains nearly constant, according to the suggested algorithm, because only neighboring managers are involved in block approval and registration. The proposed algorithm becomes stable for a long length of time on the network. The average processing time is shown in figure 9. The proposed algorithm achieves over 70% savings in processing time compared to the basic algorithm using blockchain. Reducing the authentication time of a block increases the speed of responding to IoT-level transactions, which is critical for IoT-level transactions.

4.4 Evaluation of Energy Consumption

Figure 10 outlines the energy consumption results. As is evident, the proposed algorithm increases the energy consumption. The proposed algorithm results in longer packets (due to encryption and hashing), which increases the transmission energy consumption as compared with the basic-method-without-blockchain. This increase in energy consumption is acceptable in contrast to the increase in the level of security resulting from the implementation of the algorithm.

5 Conclusion

Bandwidth, the complexity of consensus techniques, scalability, and packet overhead are the key roadblocks to integrating the BC with the Internet of Things. Our suggested Private BC eliminates these impediments and greatly improves IoT security. IoT necessitates high-speed network communications as well as a plethora of accessi-

bility rules. To solve the problem, we proposed an overlay network on high-level resources that can operate as block managers and communicate with their neighbors regarding accessibility regulations via voting mechanisms. By computing the trust value of each surrounding block manager, we present a voting-based trust approach for validating requests and block manager activity. The simulation results show that the proposed algorithm significantly reduces the average transaction confirmation time compared to previous algorithms.

References

[1] A. Kosba, A. Miller, E. Shi, Z. Wen, C. Papamanthou, Hawk: The blockchain model of cryptography and privacy-preserving smart contracts,*Security and Privacy (SP), 2016 IEEE Symposium on. IEEE* 14 (2016) 839-858

[2] A. Dorri, S. Kanhere, R. Jurdak, P. Gauravaram, LSB: A lightweight scalable BC for IoT security and privacy, *ArXiv preprint arXiv:* 1712.02969, 2017.

[3] H. F. Atlam, R. J. Walters, G. B. Wills, Internet of nano things: security issues and applications, *2nd International Conference on Cloud and Big Data Computing* 23 (2018) 7177.

[4] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, *Comput. Netw.* 54 (2010) 2787-2805.

[5] D. Giusto, A. Iera, G. Morabito, L. Atzori, The Internet of Things, *20th Tyrrhenian Workshop on Digital Communications, Springer Publishing Company, Incorporated*, 2014.

[6] Z. Shelby, K. Hartke, C. Bormann, The Constrained ApplicationProtocol (CoAP),*RFC 7252 (Proposed Standard), RFC Editor,Fremont, CA, USA,* (2014) 1-112.

[7] X. Sun, N. Ansari, Edge IoT: Mobile Edge Computing for theInternet of Things, *IEEE Communications Magazine* 54 (2016) 22-29.

[8] C. Wu, Connected vehicles and Internet of Things, *2nd International Conference on Telecommunication and Networks (TELNET)* (2017) 1-1.

[9] K. Pearsall, Manufacturing supply chain challenges - globalization andIOT, *6th Electronic System-Integration Technology Conference(ESTC)* (2016) 1-5.

[10] O. Novo, Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT, *IEEE Internet of Things Journal* 5 (2018) 1184-1195.

[11] Z. K. Zhang, M. C. Y. Cho, C. W. Wang, C. W. Hsu, C. K. Chen, S. Shieh, Iot security: ongoing challenges and research opportunities, *Service-Oriented Computing and Applications, IEEE 7th International Conference on. IEEE* (2014) 230-234

[12] G. Brambilla, M. Amoretti, F. Zanichelli, Using Block Chain for Peer-to-Peer Proof-of-Location, *ArXiv preprint arXiv:* 1607.00174 (2016).

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

[14] Inspiring, http://www.inspiring.co.uk/, accessed on Sep 20, 2017.

[15] DE. Kouicem, A. Bouabdallah, H. Lakhlef, Internet of things security: A top-down survey, *Computer Networks* 4 (2018) 199-221. http://dx.doi.org/10.1016/j.comnet.2018.03.012/.

[16] A. Gaur, B. Scotney, G. Parr, S. McClean, Smart city architecture and its applications based on IoT, *Procedia Computer Science* 52 (2015) 1089-1094

[17] M. Hassanalieragh, A. Page, T. Soyata, G. Sharma, M. Aktas, G. Mateos, B.

Kantarci, S. Andreescu, Health monitoring and management using Internet-of-Things (IoT) sensing with cloud-based processing: Opportunities and challenges, *Services Computing (SCC), 2015 IEEE International Conference on. IEEE* 285292.

[18] S, Nakamoto, Bitcoin: A peer-to-peer electronic cash system (2008).

[19] G. Brambilla, M. Amorei, F. Zanichelli, Using Block Chain for Peer-to-Peer Proof-of-Location, *ArXiv preprint arXiv:* 1607.00174 (2016).

[20] Sh. Wilkinson, J. Lowry, T. Boshevski, Metadisk a blockchain based decentralized file storage application, *Technical Report. Technical Report* http://metadisk.org/metadisk.pdf/

[21] X. Yue, H. Wang, D. Jin, M. Li, W. Jiang, Healthcare Data Gateways: Found Healthcare Intelligence on Blockchain with Novel Privacy Risk Control, *Journal of medical systems* 40 (2016) 218-231.

[22] A. Abdullahi, I. Brown, F. El-Moussa, Privacy in the age of mobility and smart devices in smart homes, *In Privacy, Security, Risk and Trust* (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom). IEEE, 819-826.

[23] V. Sivaraman, H. Habibi Gharakheili, A. Vishwanath, R. Boreli, O. Mehani, Network-level security and privacy control for smart home IoT devices, In Wireless and Mobile Computing, Networking and Communications (WiMob), *IEEE 11th International Conference on. IEEE* (2015) 163-167

[24] SH. Hashemi, F. Faghri, P. Rausch, RH. Campbell, World of empowered IoT users, *In 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)* 4 (2016) 13-24, IEEE. http://dx.doi.org/10.1109/IoTDI.2015.39/

[25] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, W. Shi, On security analysis of proof-of-elapsed-time (poet), *InInternational Symposium on Stabilization, Safety, and Security of Distributed Systems* (2017) 282-297. Springer, Cham.

[26] A. Chakravorty, T. Wlodarczyk, C. Rong, Privacy preserving data analytics for smart homes, *IEEE Security and Privacy Workshops* (2013) 23-27. http://dx.doi.org/10.1109/SPW.2013.22/

[27] S. Huh, S. Cho, S. Kim, Managing IoT devices using blockchain platform, *19th international conference on advanced communication technology (ICACT)* (2017) 464-467, http://dx.doi.org/10.23919/ICACT.2017.7890132/

[28] B. Lee, JH. Lee, Blockchain-based secure firmware update for embedded devices in an Internet of Things environment, *The Journal of Supercomputing* 73 (2017) 1152-1167.

[29] SH. Hashemi, F. Faghri, P. Rausch, RH. Campbell, World of empowered IoT users, *IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)* (2016) 13-24, http://dx.doi.org/10.1109/IoTDI.2015.39/

[30] O. Alphand, M. Amoretti, T. Claeys, S. Dall'Asta, A. Duda, G. Ferrari, F. Rousseau, B. Tourancheau, L. Veltri, F. Zanichelli, IoT Chain: A blockchain security architecture for the Internet of Things, *IEEE wireless communications and networking conference (WCNC)* (2018) 1-6, http://dx.doi.org/10.1109/WCNC.2018.8377385/

[31] AR. Chowdhury, T. Chatterjee, S. DasBi, LOCHA: a light-weight one-way cryptographic hash algorithm for wireless sensor network, *Procedia Computer Science* (2014) 497-504, http://dx.doi.org/10.1016/j.procs.2014.05.453/

[32] A. Huszti, A homomorphic encryption-based secure electronic voting scheme, *Publ. Math. Debrecen* (2011) 479-496.

[33] C. Cooja, http://anrg.usc.edu/contiki/index.php/CoojaSimulator/.

[34] Omnet++, https://omnetpp.org//, [Online; accessed July-2019].

Rezvan Mahmoudie She received Master degree in Software Engineering from Islamic Azad University, Qazvin Branch in 2010. She is currently a PhD student Software Engineering in Islamic Azad University, Science and Research Branch. Her main interest areas include IoT security.

Dr. Saeed Parsa is an Associate Professor in the Software Department of the School of Computer Engineering and the Director of the Reverse Engineering Research Laboratory (formerly parallel processing laboratory) at Iran University of Science and Technology. His research interests are in software testing, automated software engineering, compilers, and reverse software engineering. At present, the projects of their master and doctoral students are on the topics of automatic production of test data, statistical location of hidden errors, fuzzy test and automatic repair of programs. According to him, the use of compiler knowledge and reverse engineering techniques in key testing and quality assurance of software has a key role.

Dr Amir Masoud Rahmani received his BS in computer engineering from Amir Kabir University, Tehran, in 1996, the MS in computer engineering from Sharif University of Technology, Tehran, in 1998, and the PhD degree in computer engineering from IAU University, Tehran, in 2005. Currently, he is a Professor in the Department of Computer Engineering. He is the author/co-author of more than 350 publications in technical journals and conferences. His research interests are in distributed systems, the Internet of things, and evolutionary computing.