Personal and Ubiquitous Computing (2021) 25:247–254
https://doi.org/10.1007/s00779-019-01339-5

ORIGINAL ARTICLE

MBS: Multilevel Blockchain System for IoT

Bacem Mbarek[1] · Nafaâ Jabeur[2] · Tomás Pitner[1] · Ansar-Ul-Haque Yasar[3]

Received: 19 August 2019 / Accepted: 18 October 2019 / Published online: 13 November 2019

Abstract
Despite of their increasing popularity, Internet of Things (IoT) platforms are still suffering from major security problems, particularly during communications between the IoT devices. Indeed, these devices are commonly prone to malicious attacks and require prior mutual authentication to guarantee the confidentiality and security of data being shared as well as a proper network operation. In order to deal with this issue, several researchers have proposed the use of the emergent blockchain paradigm, which has emerged as a key technology that will transform the way in which information will be shared. As this paradigm is facing scalability and flexibility problems, the use of multi-agents systems has been adopted in recent works. However, current solutions are using agents for the rudimentary role of data collection only. In this paper, we propose to secure the IoT platform with a multilevel blockchain system (MBS) where the speed and flexibility of blockchain transactions are enforced by mobile agents which are migrating throughout the IoT network. The simulations of our solution through the Hyperledger Fabric are showing relevant results in terms of response time and energy consumption.

Keywords Blockchain · Internet of Things · Security · Scalability

1 Introduction

Blockchain has recently emerged as a promising technology for sharing and protecting vulnerable data in distributed and decentralized systems [1]. It is basically using public-key cryptography to sign transactions between parties. Transactions are then stored in a distributed ledger. The ledger consists of cryptographically linked blocks of data (i.e., transactions), which are almost impossible to change or remove. Thanks to their enforced security, blockchain technologies (BCT) are attracting an increasing interests from the industrial, academic, and governmental sectors. They have been already used in several application fields, such as banking, asset management, smart appliances, and healthcare. Within the context of Internet of Things (IoT), BCT are being used to address various challenges, including decentralization, anonymity, and trust [2–4]. However, the performance of the proposed solutions still needs to be improved, particularly since blockchain is fundamentally computationally expensive and involves high bandwidth overhead and delays and its related technologies are still immature. As the IoT devices are becoming capable of acting and reacting to events in their environments, we argue that blockchain solutions could benefit from the collaboration capabilities of these devices to enhance the processes of authentication and coordination of data collection as well as optimizing the use of the available resources.

Collaboration in the IoT filed is basically allowing the IoT devices to achieve tasks exceeding their individual capabilities. Because of the commonly limited processing, storage, and communication capabilities of these devices as well as the dynamic, open, and unpredictable environments in which they operate, implementing effective collaboration mechanisms remains a challenging concern. Potential solutions could be found/inspired from the extensive studies

✉ Bacem Mbarek
bacem.mbarek1@gmail.com

Nafaâ Jabeur
nafaa.jabeur@gutech.edu.om

Tomás Pitner
tomp@fi.muni.cz

Ansar-Ul-Haque Yasar
ansar.yasar@uhasselt.be

[1] Faculty of Informatics, Masaryk University, Brno, Czech Republic

[2] German University of Technology in Oman (GUtech), Athaibah, Sultanate of Oman

[3] Transportation Research Institute Hasselt University, Hasselt, Belgium

on enabling collaboration between distributed autonomous entities done in the context of multi-agent systems (MAS). MAS have, indeed, proven flexibility, autonomy, and intelligence to solve complex problems within highly dynamic, constrained, and uncertain environments [5, 6]. In this regard, proposed approaches have focused on endowing distributed software agents with intelligence to autonomously and proactively make decisions based on current activities, contextual information, and the envisioned applications. These agents have been successfully used, for example, to exchange sensitive data [7], plan critical missions (e.g., [8, 9]), optimize energy distributions (e.g., [10, 11]), and improve healthcare services (e.g., [12, 13]).

Being motivated by this success, recent works (e.g., [14–17]) have highlighted the relevance of integrating MAS and BCT. On the one hand, BCT could solve the security limitations of MAS, thanks to their capabilities of dealing with decentralization, anonymity, and trust issues [18]. On the other hand, MAS can highly improve the flexibility and scalability of blockchain solutions [19], which are not yet capable of efficiently fulfilling the requests of high volumes of concurrent access to blockchain platforms [18]. Mobile agents can also authenticate, optimize, and coordinate the data collection process for blockchain activities. In spite of the inspiring ideas presented in the works combining the use of MAS and BCT, we argue that additional efforts are still needed to allow for a seamless integration of both paradigms. Indeed, in the particular field of IoT applications, the proposed MAS architectures have limited configurability, scalability, and efficiency. In addition, the current use of mobile agents is limited to the rudimentary role of migrating to IoT devices to bring back data to the blockchain platform. The coordination of actions between the different levels of the IoT network is also not being supported correctly. We further consider that our proposed solution is application agnostic and well suited for various IoT applications. That means any use case is feasible where the blockchain collects sensor data.

We, therefore, propose in this paper a new secure, private, and lightweight multilevel blockchain system (MBS) for IoT applications. Our solution is based on the use of mobile agents to reduce blockchain processing and communication overheads. To this end, the agents collaboratively execute blockchain functions at different levels of the IoT network before transferring the hashed data blocks they create to the blockchain platform. Compared to the existing solutions, our contributions could be summarized as follows: (1) a new multilevel architecture combining the three paradigms of IoT, blockchain, and multi-agent systems; (2) a new solution enabling the improvement of the blockchain scalability via the use of mobile agents that migrate throughout the IoT network and perform blockchain activities at different hierarchical levels of this network; and (3) a new multilevel solution based on mobile agents to reduce energy consumption and communication overheads in blockchain-IoT applications.

In the remainder of this paper, Section 2 outlines the existing works that have addressed the use of BCT within the IoT context. It also outlines the use of MAS along with BCT in order to improve IoT applications. Section 3 describes our proposed solution. It particularly focuses on our proposed authentication scheme as well as on our use of BCT and mobile agents toward an improved authentication process. Section 4 provides a performance evaluation of our solution. Section 5 summarizes the paper and highlights our future works.

2 Related work

Because of the commonly restricted resources and the heterogeneity of their devices, IoT networks are facing serious security problems [20]. These problems are particularly significant due to the increasing amounts of data which are being collected and shared between the IoT devices [21]. In order to overcome these problems, recent works (e.g., [22, 23]) have highlighted the relevance of using the emergent BCT along with the IoT paradigm. In this regard, Jesus et al. [24] have presented a survey on how to use blockchain to secure the IoT network and its related activities. Golomb et al. [25] have proposed the solution CIoTA where blockchain is used jointly with extensible Markov model (EMM) to discover abnormalities in IoT applications. Dorri et al. [21] have investigated the breaches in existing security and privacy methods and proposed a lightweight and scalable blockchain (LSB) solution to reduce bandwidth and computation costs while improving IoT security and privacy. Huckle et al. [26] have investigated the contributions of blockchain for monetizing IoT applications. Huh et al. [27] have demonstrated how blockchain can be used to store IoT devices' data with smart contracts. Shafagh et al. [28] have proposed a new framework where blockchain is used to support the distributed storage of data in IoT applications. Danzi et al. [29] have proposed various mechanisms/protocols that enable the synchronization of IoT endpoints to the blockchain. To this end, the authors have used several security levels and communication requirements. They have also analyzed the traffic and the power consumption created by the synchronization protocols via numerical simulations. Extended reviews of using

blockchain to store IoT data (with a focus on open problems of anonymity, integrity, and adaptability) as well as to architect blockchain-based IoT applications are available in [30] and [31] respectively.

In spite of their relevance, the proposed solutions are suffering from the problems of scalability, delays, and bandwidth overhead inherited from the blockchain paradigm. As demonstrated by Danzi et al. [29], these problems could get worse due to high communication exchanges between the IoT devices. Furthermore, reaching consensus, for example on adding a new block to the ledger, can be extremely complex in a distributed network (like the IoT) where peers may have unaligned interests [32]. To overcome such issues, recent works (e.g., [33, 34]) have supported the idea of integrating multi-agent systems (MAS) with BCT. In this context, Ferrer [35] has demonstrated that by integrating cryptographic algorithms with peer-to-peer (P2P) networks, software agents can autonomously be able to find consensus on particular state of affairs. Calvaresi et al. [18] have highlighted that MAS combined with BCT can provide the necessary means to make the operations of distributed entities more autonomous, flexible, secure, and profitable. Kapitonov et al. [19] have presented a solution to organize communications involving software agents in an IoT network. The solution uses the decentralized Hyperledger blockchain technology as well as smart contracts. In this paper, we argue that the potential of software agents, and particularly mobile agents, is still untapped. We also argue that the right support from agents will definitely leverage the performance of any combined blockchain-IoT solution.

3 Multilevel blockchain system (MBS)

3.1 Network model

In order to support the exponential growth of IoT, we are proposing a blockchain-based system to enforce the privacy and security matters related to collecting, sharing, and managing vulnerable data. To reduce blockchain overheads while addressing its scalability problems, we are deploying mobile agents that migrate throughout the IoT network to collect relevant data, create hashed data blocks, reduce response time delays, and solve synchronization and scalability problems. We are summarizing our solution into a multilevel blockchain system (MBS) that allows smart devices to securely send their data to the central IoT processing platform (also blockchain platform). The levels of the proposed solution are (see Fig. 1): (1) micro-level (IoT device level); (2) meso-level blockchain (cluster heads level of the IoT network); and (3) macro-level blockchain (platform level or also the highest hierarchical level of the IoT network).

3.2 Overview of our layered design

The implemented platform consists of four parts as shown in Fig. 2: IoT device, ordering service, endorsing peers, and committing peers.

- IoT device: These devices are collecting and transmitting data to private blockchain distributed ledgers.
- Endorsing peers: During the commissioning and configuration of the blockchain network, the developers

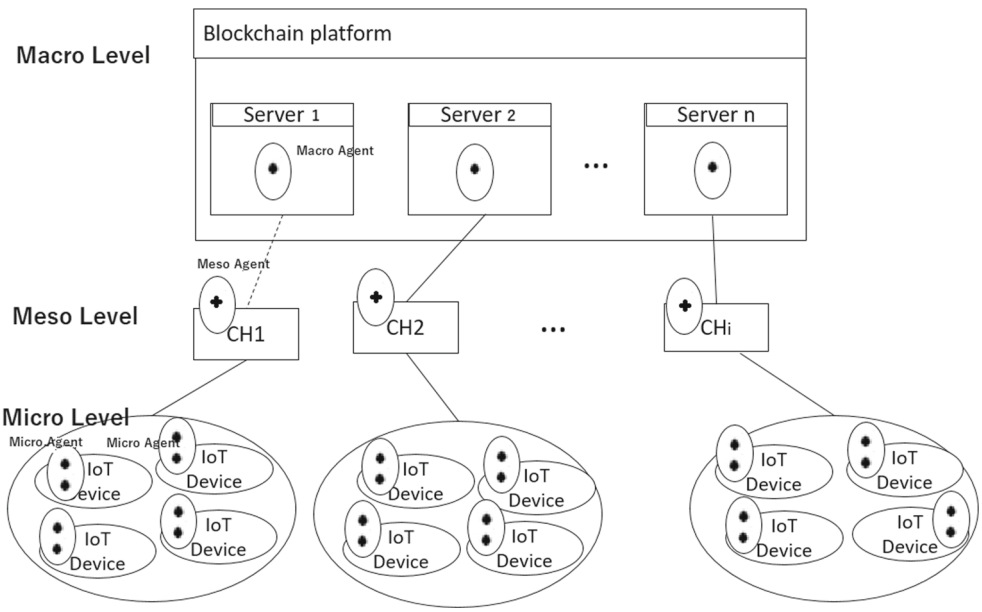

Fig. 1 Hierarchical architecture for data management

should select a number of IoT devices defined as the endorsing peers.
- Ordering service: Creates the block of transactions, and sends it to all the peers. The ordering service collects transactions for a channel into proposed blocks for distribution to peers. Blocks are delivered on a channel basis. The ordering service accepts endorsed transactions, orders them into a block, and delivers the blocks to the committing peers. The main responsibility of ordering service is to receive transactions from the IoT devices and fit into a block.
- Committing peers: A committing peers are a predefined number of IoT devices. Usually endorsing peers are also committing, but a peer can be only committing and not endorsing. Committing peers (including endorsing peers) run validation and update their copy of the blockchain and world state. Each peer receiving the block, now in the role of a committing peer, appends the whole block to its blockchain copy. Committing peers are responsible for adding blocks of transactions to the shared ledger and updating the world state. They may hold smart contracts, but it is not a requirement.

3.2.1 Multiagent system model

Our solution includes three types of agents: (1) macro agents, (2) meso agents, and (3) micro agents. The role of these agents and their locations are depicted on Fig. 2 and explained in what follows.

Macro agent: A macro agent is located on the highest level of the IoT, which is also called the blockchain platform (BPL). This agent is responsible of authenticating the IoT clusters and their cluster heads. It is also responsible of creating the related meso agents and collecting from them the encrypted blocks of data of interest. The macro agent will then update the blockchain accordingly and store related data for subsequent uses within the context of the envisioned application.

Meso agents: We assume here that the IoT network is following a specific algorithm to create clusters and elect their heads (this algorithm is out of the scope of this paper). Once authenticated at the macro level of the system, a given cluster will receive its specific meso agent. This agent will collect and aggregate the data of interest from the micro agents assigned to the IoT devices of its cluster. It will then create and encrypt the data block corresponding to its cluster. The meso agent will then migrate to the macro level to deliver the data to the macro agent.

Micro agents: The micro-level of the IoT hierarchy is composed of IoT devices. Once an IoT device is authenticated as member of a given cluster, its related micro agent will be created by the meso agent of the

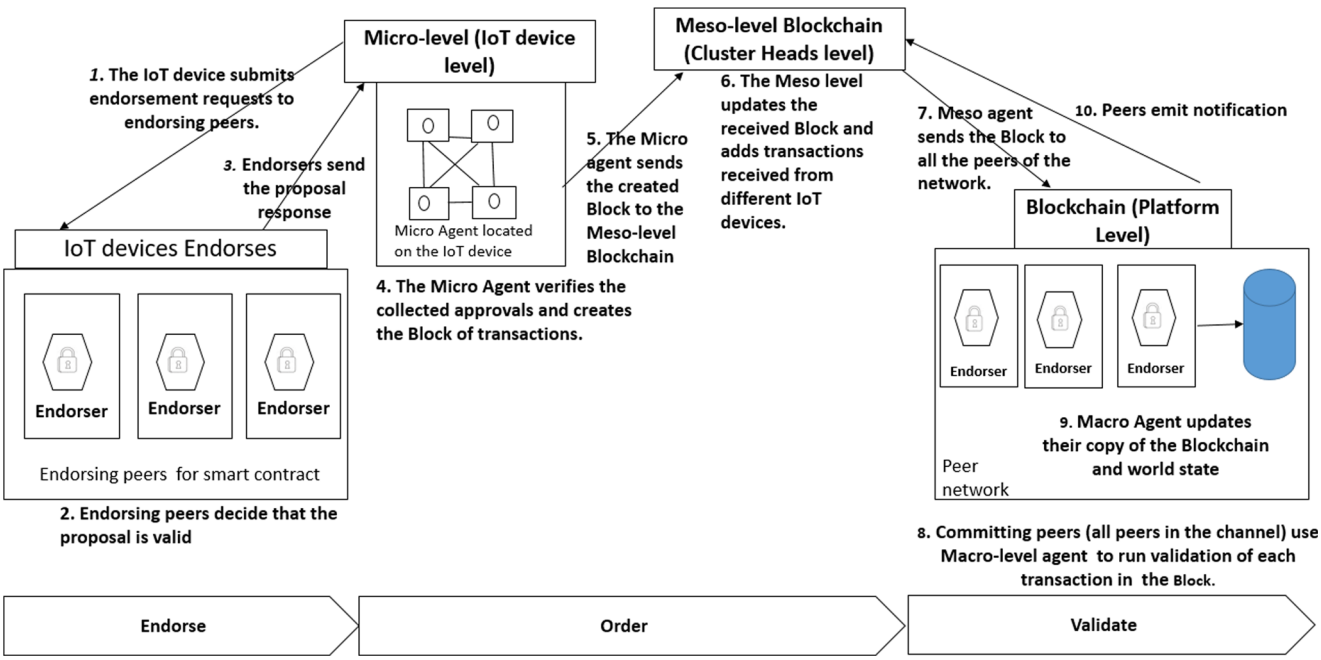

Fig. 2 Order-execute architecture to store data in MBS blockchain platform

cluster. The micro agent will migrate to the IoT device and collect the data of interest periodically or on demand. The agent will then encrypt the data and migrate back to the meso level (i.e., the head of its cluster).

3.2.2 Order-execute architecture in MBS blockchain platform

Figure 2 shows the order-execute architecture to data storage in the MBS blockchain platform. This architecture can be summarized as follows.

1. The IoT device submits endorsement requests to endorsing peers.
2. Endorsing peers approve or reject the status after checking for consistency (checking the smart contract).
3. Endorsing peers send the proposal response to the IoT device applicant. Each execution captures the set of read and written data (also called the RW set), which is flowing in the Hyperledger Fabric. Moreover, the endorsement system chaincode (ESCC) signs the proposal response on the endorsing peer . The RW sets are signed by each endorser. Every set will include the number of the version of each record.
4. Afterwards, the ordering service located in the micro agent verifies the collected endorsements and creates the block of transactions.
5. The micro agent sends the created block to the meso-level blockchain (cluster heads level).
6. The meso level updates the received block and adds transactions received from different IoT devices.
7. Then, the meso agent sends the block to all the peers of the network to be appended to their blockchain copies.
8. The peers check the transaction's validity (verification of the smart contract).
9. If the validity is fulfilled, finally, the macro level (blockchain platform) adds the checked block into the data base of the blockchain platform.
10. Peers emit notification.

3.2.3 Smart contracts for monetizing IoT data

The concept of smart contracts can be used to automate negotiations between service providers and users along with required monetary transactions without a trusted intermediary. A smart contract is fundamentally a code that validates a negotiation and immediately brings a contract into effect, without the involvement of any intermediaries [36]. The smart contract code resides on a blockchain as multiple functions with unique addresses that can be called by any user of the blockchain. All entities interacting with a blockchain (including users and Things) must own at least one public-private key pair. First of all, the sender encrypts a smart contract with the public key. The receiver then receives the encrypted smart contract and decrypts it with the private key.

4 Performance evaluation

In order to implement our multilevel blockchain system (MBS), we used Hyperledger Fabric [37], which is an open source permissioned blockchain introduced by IBM. We run our simulations on a large-scale network of 1000 nodes. We assumed that all nodes have a fixed position throughout the period of simulations, with the settings provided in Table 1. The results have been obtained by a confidence interval of 95%. We compared our MBS solution with the solution proposed in [19], which focuses on organizing a communication system between agents (representing smart things or robots) in a P2P network using the decentralized Hyperledger blockchain technology and smart contracts to reduce energy consumption. This choice was motivated by the fact that both solutions are integrating the use of MAS and blockchain at several levels of the IoT hierarchy. The setup of our experiments is as follows: (1) nodes run on Fabric version v1.1.0-preview2 instrumented for performance evaluation through local logging; (2) all nodes are 2.0 GHz 16-vCPU VMs running Ubuntu with 8 GB of RAM and SSDs as local disks; (3) there are 1000 peers in total, 100 IoT devices endorsers; (4) signatures use the default SHA-1 scheme; and (5) we adopt 2 MB as data block sizes. During our experiments, we analyzed the size mint and spend transactions. In particular, the 2-MB data blocks contained 473 mint or 670 spend transactions. In other words, the average transaction size is 3.06 kB for spend and 4.33 kB for mint. In general, transactions in Fabric are large because they carry certificate information.

We have implemented a micro agent in each IoT device as a Hyperledger Fabric ordering service to create blocks. Micro agents' actions are executed when an IoT device received and verified a smart contract request. Each node in the system runs Ubuntu with 8GB of memory. We have also implemented a meso agent in each cluster head which can

Table 1 Simulation parameters

Parameter	Value
Endorsing peers	100 peers
Cluster head	50
Block size	2MB
Run times	50 times
Simulation time	100 s
Number of nodes	1000

collect the received blocks from different IoT devices. We install on the blockchain platform a macro agent to validate the chain of blocks received from the cluster head and to update the blockchain platform.

4.1 Response time

Figure 3 outlines our assessment of response time for both solutions MBS and BCE. The figure clearly demonstrates that our solution outperforms the solution proposed by [19]. For instance, for the same size of network consisting in 1200 nodes, the response time of BCE is around 9 ms, compared to about 5 ms in MBS. This could be explained by the fact that BCE uses a single blockchain level installed in a remote server platform. In this configuration, decisions about the necessary actions to perform (e.g., data hashing, data encryption) will be delayed as many mobile agents will not be able to concurrently access the blockchain platform to insert their data. In our solution, mobile agents at different levels of the IoT network have already the rights and the duty to perform blockchain actions ahead before migrating to the remote blockchain platform. Once there, they will not be constrained to wait until blockchain actions are executed and relevant actions are taken accordingly.

4.2 Energy consumption

Figure 4 gives the average energy consumption in Joules which is needed to store the data in the blockchain platform. Compared to the BCE where each data transaction is processed and sent individually, our MBS solution consumes less energy since our meso agents are collecting

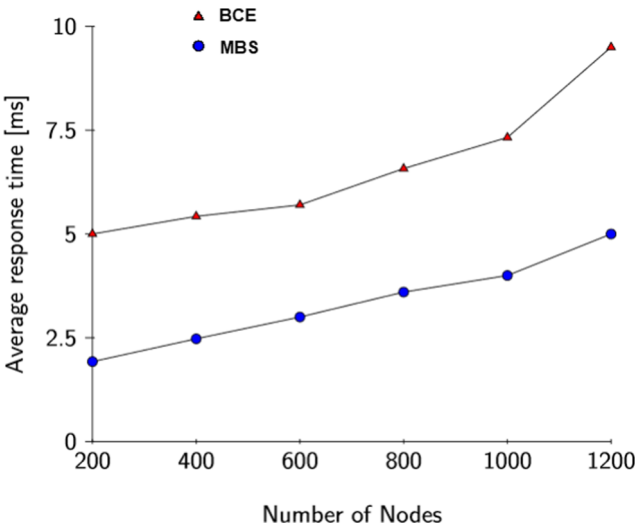

Fig. 3 Average response time to store data in blockchain platform

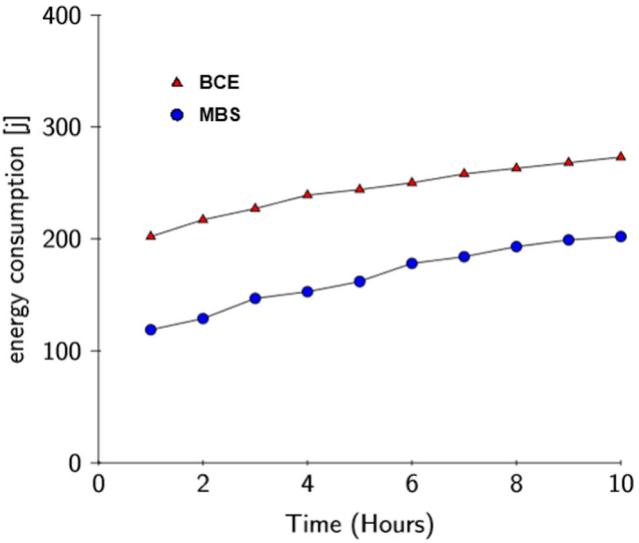

Fig. 4 Energy consumption

and aggregating blocks of data to reduce communication overheads. For example, as could be seen on Fig. 4, at time of 8 hours, BCE uses about 250 (j) whereas MBS consumes only 190 (j).

4.3 Use case: Internet of bikes

The IoB (Internet of Bikes) is one of the widely spreading examples of distributed applications [38], urging for highly secure, independent, and distributed platform, which blockchain is capable of supporting. Moreover, blockchain needs to be established in IoB applications in order to truly leverage the benefits provided by its distributed ledger to find solutions to problems in bike sharing such as bike deployment, redistribution, parking, and maintenance. Due to the nature of the blockchain, all bikes information in the edge can be tracked in a synchronized and distributed blocks.

In order to prevent such threats, we are proposing to send encrypted bike information based on blockchain. In addition, for guaranteeing confidentiality, our proposed security scheme allows for encrypting and generating illegible messages. It has been proven (e.g., [39]) that these messages are only accessible by authenticated parties. For additional confidentiality, smart contract is used alongside with cryptography. Therefore, we are proposing a blockchain-based system to enforce the privacy and security matters related to collecting, sharing, and managing vulnerable data. Moreover, we encounter the blockchain overhead communication issues by deploying agent that collects data, creates encrypted data, creates blocks, and

solves synchronization and scalability problems in IoB environment.

5 Conclusion

We proposed in this paper a multilevel blockchain system (MBS) to enhance current data security and privacy in IoT applications while improving response time and energy consumption. Our solution is based on the use of mobile agents that are capable of migrating between the different levels of the IoT network and the blockchain platform and execute the necessary blockchain functions as well as hashing, encryption, aggregation, and decryption functions when and where needed. Simulations of our solutions are showing satisfactory results. These results are motivating us to focus our future works on implementing our solution in a real IoT case study.

References

1. Back A, Corallo M, Dashjr L, Friedenbach M, Maxwell G, Miller A, Poelstra A, Timón J, Wuille P Enabling blockchain innovations with pegged sidechains, http://www.opensciencereview.com/papers/123/enablingblockchain-innovations-with-pegged-sidechains
2. Li Z, Kang J, Yu R, Ye D, Deng Q, Zhang Y (2018) Consortium blockchain for secure energy trading in industrial internet of things, IEEE Transactions on Industrial Informatics
3. Signorini M, Pontecorvi M, Kanoun W, Di Pietro R (2018) Bad: blockchain anomaly detection. arXiv:1807.03833
4. Mbarek B, Jabeur N et al (2018) Ecass: an encryption compression aggregation security scheme for secure data transmission in ambient assisted living systems. Personal and Ubiquitous Computing
5. Wray K, Thompson B (2014) An application of multiagent learning in highly dynamic environments. In: AAAI workshop on multiagent interaction without prior coordination
6. Kong Y, Zhang M, Ye D (2017) A belief propagation-based method for task allocation in open and dynamic cloud environments, Knowledge-Based Systems
7. Schatten M, Ševa J., Tomičić I. (2016) A roadmap for scalable agent organizations in the internet of everything, Journal of Systems and Software
8. Wong D, Paciorek N, Moore D (1999) Java-based mobile agents. Commun ACM 42(3):92–ff
9. Caripe W, Cybenko G, Moizumi K, Gray R (1998) Network awareness and mobile agent systems. IEEE Commun Mag 36(7):44–49
10. Tong L, Zhao Q, Adireddy S (2003) Sensor networks with mobile agents. In: Military communications conference, 2003. MILCOM'03. 2003 IEEE, vol 1. IEEE, pp 688–693
11. Dong M, Ota K, Yang LT, Chang S, Zhu H, Zhou Z (2014) Mobile agent-based energy-aware and user-centric data collection in wireless sensor networks. Comput Netw 74:58–70
12. Su C-J, Wu C-Y (2011) Jade implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring. Appl Soft Comput 11(1):315–325
13. Chan V, Ray P, Parameswaran N (2008) Mobile e-health monitoring: an agent-based approach, IET communications
14. Kvaternik K, Laszka A, Walker M, Schmidt D, Sturm M, Dubey A et al (2017) Privacy-preserving platform for transactive energy systems. arXiv:1709.09597
15. Qayumi K (2015) Multi-agent based intelligence generation from very large datasets. In: IEEE international conference
16. Norta A, Othman AB, Taveter K (2015) Conflict-resolution lifecycles for governed decentralized autonomous organization collaboration. In: Proceedings of the 2015 2nd international conference on electronic governance and open society: Challenges in Eurasia. ACM, pp 244–257
17. Ponomarev S, Voronkov A (2017) Multi-agent systems and decentralized artificial superintelligence. arXiv:1702.08529
18. Calvaresi D, Dubovitskaya A, Calbimonte JP, Taveter K, Schumacher M (2018) Multi-agent systems and blockchain: results from a systematic literature review. In: International conference on practical applications of agents and multi-agent systems. Springer
19. Kapitonov A, Lonshakov S, Krupenkin A, Berman I (2017) Blockchain-based protocol of autonomous business activity for multi-agent systems consisting of uavs. In: Research education and development of unmanned aerial systems (RED-UAS)
20. Jing Q, Vasilakos AV, Wan J, Lu J, Qiu D (2014) Security of the internet of things: perspectives and challenges. Wireless Netw 20(6):2481–2501
21. Dorri A, Kanhere SS, Jurdak R, Gauravaram P (2017) Lsb: A lightweight scalable blockchain for iot security and privacy. arXiv:1712.02969
22. Kshetri N (2017) Can blockchain strengthen the internet of things? IT Professional 19(4):68–72
23. Christidis K, Devetsikiotis M (2016) Blockchains and smart contracts for the internet of things. IEEE Access v4:2292–2303
24. Jesus EF, Chicarino VR, de Albuquerque CV, Rocha A. A. d. A. (2018) A survey of how to use blockchain to secure internet of things and the stalker attack, Security and Communication Networks
25. Golomb T, Mirsky Y, Elovici Y (2018) Ciota:, Collaborative iot anomaly detection via blockchain. arXiv:1803.03807
26. Huckle S, Bhattacharya R, White M, Beloff N (2016) Internet of things, blockchain and shared economy applications. Procedia Comput Sci 98:461–466
27. Huh S, Cho S, Kim S (2017) Managing iot devices using blockchain platform. In: 2017 19th international conference on advanced communication technology (ICACT). IEEE, pp 464–467
28. Shafagh H, Burkhalter L, Hithnawi A, Duquennoy S (2017) Towards blockchain-based auditable storage and sharing of iot data, In: Proceedings of the 2017 on cloud computing security workshop. ACM, pp 45–50
29. Danzi P, Kalor AE, Stefanovic C, Popovski P (2018) Analysis of the communication traffic for blockchain synchronization of iot devices. In: 2018 IEEE international conference on communications (ICC). IEEE, pp 1–7
30. Conoscenti M, Vetro A, De Martin JC (2016) Blockchain for the internet of things: a systematic literature review. In: 2016 IEEE/ACS 13th international conference of computer systems and applications (AICCSA). IEEE, pp 1–6
31. Ramachandran GS, Radhakrishnan R, Krishnamachari B (2018) Towards a decentralized data marketplace for smart cities. In: invited paper at the 1st international workshop on BLockchain enabled sustainable smart cities (BLESS 2018), Kansas City, MO, USA, held in conjunction with the 4th IEEE Annual International Smart Cities Conference (ISC2)

32. Shermin V (2017) Disrupting governance with blockchains and smart contracts. Strateg Chang 26(5):499–509
33. Kvaternik K, Laszka A, Walker M, Schmidt D, Sturm M, Dubey A et al (2017) Privacy-preserving platform for transactive energy systems. arXiv:1709.09597
34. Ponomarev S, Voronkov A (2017) Multi-agent systems and decentralized artificial superintelligence. arXiv:1702.08529
35. Ferrer EC (2018) The blockchain: a new framework for robotic swarm systems. In: Proceedings of the future technologies conference. Springer, pp 1037–1058
36. Di Pierro M (2017) What is the blockchain? Comput Sci Eng 19(5):92–95
37. Hyperledger fabric. https://github.com/hyperledger/fabric
38. Behrendt F (2016) Why cycling matters for smart cities. internet of bicycles for intelligent transport. Journal of Transport Geography 56:157–164
39. Hamid N, Yahya A, Ahmad RB, Al-Qershi OM (2012) Image steganography techniques: an overview. Int J Comput Sci Secur (IJCSS) 6(3):168–187

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com