Computers, Materials & Continua CMC, vol.65, no.3, pp.2441-2453, 2020

Task-Attribute-Based Access Control Scheme for IoT via Blockchain

Hao Chen[1], Wunan Wan[1, *], Jinyue Xia[2, *], Shibin Zhang[1], Jinquan Zhang[1], Xizi Peng[1] and Xingjie Fan[1]

Abstract: As a new form of network, the Internet of things (IoT) is becoming more widely used in people's lives. In this paper, related theoretical research and practical applications of the IoT are explored. The security of the IoT has become a hot research topic. Access controls are methods that control reasonable allocations of data and resources and ensure the security of the IoT. However, most access control systems do not dynamically assign users' rights. Additionally, with some access control systems, there is a risk of overstepping other user's authority, and there may exist a central authority that is a single point of failure. Therefore, to solve these problems, this paper proposes a Task-Attribute-Based Access Control scheme for the IoT via blockchain that combines the access control technologies of both the IoT and blockchain. This model, which merges the advantages of task-based access controls and attribute-based access controls, is perfectly integrated with blockchain technology. This model uses hash functions and digital signature algorithms to ensure the authenticity and integrity of the data, and it can dynamically allocate users' minimum privileges and thus perfectly solves the single point of failure problem. The model is implemented using a Geth client and solidity code, and the simulation results demonstrate the effectiveness of the model.

Keywords: Access control, task-attribute-based access control, blockchain, consortium blockchain, Internet of Things.

1 Introduction

Presently, the Internet of Things (IoT) is increasingly being used in people's daily lives. The IoT has extended the connection between the Internet and physical things employed on a regular basis. The IoT is increasingly being utilized in different types, numbers and scenarios, and it can be used in furniture, grids, cities, transportation, vehicles and many other places.

Although use of the IoT is accelerating and is making people's everyday lives more convenient due to the rapid development of new technologies, problems endemic to the

[1] Chengdu University of Information Technology, Chengdu, 610225, China.

[2] International Business Machines Corporation, New York, USA.

[*] Corresponding Author: Wunan Wan. Email: nan_wwn@cuit.edu.cn.

Received: 31 May 2020; Accepted: 22 July 2020.

IoT have gradually emerged [Ren, Zhu, Sharma Pradip et al. (2020); Yin and Wei (2018)]. One critical problem in the IoT is the access control of data and other resources. Access control is a key technology to manage the capabilities and scope of users' access to data and information through the identification of legitimate users. Access control facilitates the authorized use of key resources and prevents illegal users from using system resources and legal users from overusing system resources. Access control is the cornerstone of a network security system and can effectively guarantee data privacy. Due to the large number of devices in the IoT, traditional access control schemes, such as discretionary access controls using an ACL list, are not applicable because there is no way to configure an ACL list for all IoT devices. Mandatory access control systems, which have a central administrator, have a single point of failure [Ding, Cao, Li et al. (2019)]. Role-based access control systems do not assign privileges to devices dynamically, so malicious users can access or damage resources beyond their authority.

Task-attribute-based access control systems provide a dynamic, scalable and flexible access control scheme. These systems are based on the state of the task, the required conditions and the environment in which the task is performed. Other information is used to allocate privileges in real time, adhering to the principle of minimum privilege. The advantage of attribute-based access control is that it eliminates the need to create an ACL table, making attribute-based access controls applicable to a large number of IoT environments. Currently, most people know that blockchain provides distributed and decentralized solutions, which can mitigate the single point of failure problem. The combination of blockchain and the IoT to solve network problems is now a hot research topic that is applicable to many scenarios [Kim, Min and Kim (2019); Medhane, Sangaiah, Hossain et al. (2020); Ren, Liu, Ji et al. (2018)]. When transactions in a blockchain are recorded into a block, it is impossible to tamper with the transaction information. Therefore, blockchain can provide increased security for the IoT.

In this paper, our main research contributions are as follows:

1. We put forward an innovative idea of combining access controls based on tasks and attributes, with blockchain so that IoT users can only obtain the minimum privilege to perform requested tasks and thus ensure user security.
2. This model can help users of the IoT observe the operational status and conditions of the required tasks in real time and assign corresponding tasks and attributes to users of the IoT, which helps to make access requests faster and more efficient.
3. The model we proposed was simulated, and its safety and performance were analyzed, which proves the feasibility of the system model.

The structure of this paper is as follows: Section 2 summarizes related studies. Section 3 introduces the background information. The detailed design of our proposed scheme is presented in Section 4, and Section 5 outlines the security analysis. Section 6 introduces the implementation of our model and provides a performance analysis. Finally, the conclusions are given in Section 7.

2 Related works

Unauthorized access to IoT devices is a major risk to IoT networks. Access control in the

IoT is mainly performed by limiting the rights of subjects to access objects. Access control is one of the foundations of the overall security of IoT systems. Beltrand et al. [Beltran and Skarmeta (2019)] introduced different architectural models for access control and explained the different security implications of each model. The OAuth protocol provides a secure, open and easy standard for authorization of user resources [Hardt (2012)]. Oh et al. [Oh, Kim and Cho (2019)] proposed an interoperable IoT platform access control framework based on the OAuth protocol that uses an access token with global access capabilities on an entire IoT platform. According to Ouaddah et al. [Ouaddah, Mousannif and Ait Ouahman (2015)] and Gusmeroli et al. [Gusmeroli, Piccione and Rotondi (2012)], most access control models used in the IoT are well known. However, the access controls for the IoT have a single point of failure, which is why we are considering using blockchain technology to provide access control services. The use of this technology to manage the distribution, heterogeneity, scalability, fault tolerance, security and privacy of devices connected to the Internet of things is promising [Riabi, Ayed and Saidane (2019)].

Blockchain is a new application that was proposed by Nakamoto [Nakamoto (2008)]. Blockchain mainly uses distributed data storage, point-to-point transmissions, consensus mechanisms, encryption algorithms, timestamps and other technologies to implement its security [Couto da Silva (2019); Aljosha, Nicholas, Katharina et al. (2017)] and uses cryptography to guarantee the privacy of the block data [Wan, Chen, Chen et al. (2019)]. Blockchain is decentralized, so combining it with the IoT is a good way to solve the single point of failure problem and design the IoT to be decentralized and private [Conoscenti, Vetrò and De Martin (2016)]. FairAccess is a new framework for access control of the Internet of things based on blockchain technology proposed by Ouaddah et al. [Ouaddah, Abou Elkalam and Ait Ouahman (2016)]. FairAccess enables users to own and control data in the framework. The attribute-based access control scheme of IoT systems proposed by Ding et al. [Ding, Cao, Li et al. (2019)] avoids single points of failure and data tampering. Xu et al. [Xu, Chen, Blasch et al. (2018)] proposed BlendCAC, a token management strategy for access authorization using smart contracts. However, these blockchain access control schemes are prone to overreach because they cannot dynamically assign a user minimum privileges according to the states of the executing tasks. Therefore, we focus on task-based access control [Lu, Zhang and Sun (2008)], which can dynamically assign the user minimum privileges in real time, and study how task-based access control can combine with other access control systems [Liu (2010); Uddin and Islam (2019)]. In this study, we propose a new model based on these findings.

Vujičić et al. [Vujičić, Jagodić and Ranđić (2018)] studied and summarized the Bitcoin and Ethereum technologies and explained the differences and connections between them. Hammi et al. [Hammi, Hammi, Bellot et al. (2018); Riabi, Dhif, Ben Ayed et al. (2019)] specifically implemented their solution with an Ethereum smart contract, and we surmised that Ethereum was more suitable for the implementation of our proposed model.

3 Background

3.1 Consortium blockchain

Blockchains come in three forms: public blockchains, private blockchains, and

consortium blockchains [Shahzad and Crowcroft (2019)]. A public blockchain cannot be edited and is sporadic, but some device states in the IoT need to be changed, which makes a public blockchain unsuitable for the IoT. Although a private blockchain does not have this problem, the devices become independent when used in the IoT. Therefore, a private blockchain is also not suitable for the IoT. A consortium blockchain combines the characteristics of the other two blockchains. A consortium blockchain belongs to members of an alliance, unlike private blockchains. Moreover, as long as most of the consortium nodes in the consortium blockchain reach a consensus, the data in the blockchain can be modified. Therefore, the consortium blockchain makes up for the other shortcomings and makes it usable for the IoT.

3.2 Address in a blockchain

There are three properties of blockchains: public keys, private keys and addresses. The relationship between a private key and a public key is the same as in an asymmetric encryption algorithm. The private key is selected by the user him/herself, then the public key is calculated through the private key, and finally, the address is calculated through the public key. Public keys and addresses are public parameters. The address of a blockchain is a string of numbers and letters. Ethereum uses the elliptic curve signature algorithm secp256k1 to calculate the public key according to the user's private key; then, it uses the hash algorithm Keccak256 to obtain the hexadecimal string, remove the last 40 letters, and add 0x at the beginning to generate the address. This is also the way we generated addresses in our scenario.

3.3 Access control technology

Access control is used to restrict the access of the subject to the object to ensure the effective use and management of data resources within a legal framework. For example, users of some video repositories need to distinguish between members and nonmembers. Some videos in the repositories can only be watched by members, while nonmembers cannot watch these videos. This is a kind of role-based access control system. If an age limit is added, some users need to be sixteen years of age to watch specific videos, and even if a person is a member, if he or she is under the age of sixteen, he or she cannot watch these videos. This is a kind of attribute-based access control. However, these access controls do not use dynamic allocation of the user’s minimum privileges and already assigned privileges, as we observed in the task-based access control systems.

3.4 The principle of minimum privilege

The principle of minimum privilege means that a user cannot have more privileges than he or she needs to perform a task. This principle is not implemented in most access control scenarios. Role-based access control systems first assign users a role whose privileges are defined when the system is designed, so users may perform tasks with more permissions than they need. The task-attribute-based access control scheme assigns privileges dynamically, so it avoids this problem and implements the principle of minimum privilege.

4 The proposed scheme

4.1 System model

First, we introduce the system model, in which there are three roles: task attribute distributor (TAD) and IoT users and miners.

4.1.1 Task attribute distributor

TAD is a node in the consortium blockchain that manages IoT users in its own region. TADs verify the qualification of IoT users, assign corresponding tasks and attributes, generate corresponding transactions and store them in their own transaction pool. TADs wait for miners to take the transactions in the transaction pool and publish them on the blockchain. Each TAD has its own $TADID_i$.

4.1.2 IoT users

IoT users mainly share resources, perform tasks, and apply to TAD for corresponding attributes and tasks to gain access control. Each IoT user has his or her own UID_i.

4.1.3 Miner

Miners compete for the billing rights of the blocks, and the miner who competes for the billing rights takes transactions out of TAD's transaction pool in each region and publishes them on the blockchain.

The system selects the elliptic curve signature algorithm, secp256k1, and the hash algorithm, Keccak256, to generate the corresponding address. The basis point selected in the elliptic curve signature algorithm secp256k1 is G. TAD_i first randomly selects a number SK_{TAD_i} as its private key, so the corresponding public key is $PK_{TAD_i} = SK_{TAD_i} \times G$. The system uses the Keccak256 algorithm to calculate the hash value of $TADID_i \,||\, PK_{TAD_i}$ and converts it into a hexadecimal string, removes the last 40 letters, and adds 0x at the beginning to generate the address as a TAD. The IoT users also choose a random number SK_{U_i} as their private key, and the corresponding public key is $PK_{U_i} = SK_{U_i} \times G$. Then, the Keccak256 algorithm is used to calculate the hash value of $UID_i \,||\, PK_{U_i}$ and convert it into hexadecimal string; then, it removes the last 40 letters and adds 0x at the beginning to generate the address as Address.

4.2 System structure

The structure of the system is shown in Fig. 1.

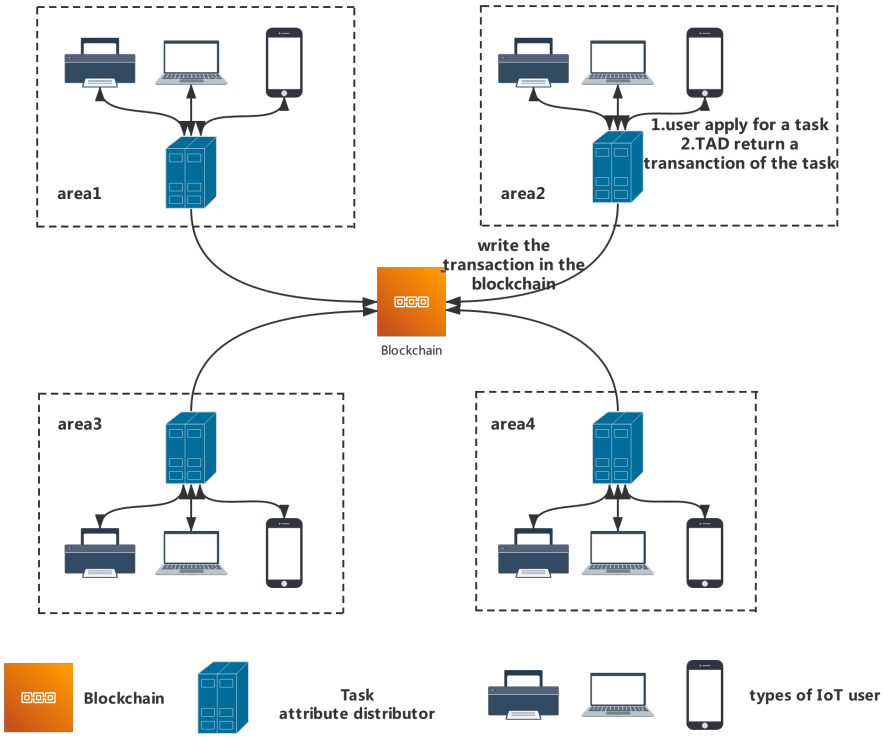

Figure 1: System structure of the proposed scheme

4.2.1 Task registration

Each TAD first writes information about the tasks in its region to its own transaction pool in the form of {TID, TC, TP}. TID represents the id of a task. TC represents the condition of a task. TC includes condition information, such as the task state, life cycle of the task, and required attributes of the task. TP represents the privilege of a task. TP represents the minimum privilege of the user who can perform the task. There are five states of a tasks: the ready state is denoted as RS, the active state is denoted as AS, the execution state is denoted as ES, the suspended state is denoted as SS, and the invalid state is denoted as IS. RS indicates that the status of the task conditions has not yet been met; AS indicates that the task conditions have been met and that it is ready to enter the execution status; ES indicates that the task is in the running process; SS indicates that the task is placed in this state due to unsatisfied conditions or other reasons and it can be restored to the execution state or become an invalid state due to the end of a life cycle or for other reasons; and IS indicates that a task has completed or is placed in this state for some other reason. The privileges that each state can assign to the user are varied.

4.2.2 Task request

The public key of each TAD is used to generate their addresses, and the private key is used to sign the transactions. TAD first verifies whether IoT users are eligible for tasks and attributes. If the validation is successful, a transaction is generated in the form of

$TAD \xrightarrow{(attr,tid)} Address$. attr represents the attribute, and tid represents the id of the task. TAD then combines this transaction with a timestamp and signs it with its private key in the form of $Sig_{SK_{TAD_i}}(TAD \xrightarrow{(attr,tid)} Address \| timestamp)$. TAD packages transactions, signatures, and timestamps into its own transaction pool.

4.2.3 Broadcasting transactions

When new blocks are generated by blockchain, miners who have obtained block accounting rights view the transactions in the transaction pool of the TAD in each region, which sorts all the transactions in the transaction pool according to tid and then packs those transactions and broadcast them to other consortium nodes so that they can reach a consensus.

4.2.4 Implementation of access control

The concrete implementation of the model is shown in Fig. 2 and is described as follows:

Alice first negotiated a session key K with Bob, which was generated by a symmetric encryption algorithm, and the communication between Alice and Bob was encrypted and decrypted by this session key.

①Alice checked the task returned from TAD to see if she can apply for the task, and if she can, she then requests the task to Bob and sends ID_A to Bob.

②After receiving ID_A, Bob selects a random number N and the access policy P and sends it to Alice.

③After receiving the random number N and the access policy P, Alice chooses a subset S that satisfies the access policy, signs N with her private key in the form of $Sig_{SK_{U_A}}(N)$, and then sends subset S, taskid, task attribute, public key PK_{U_A} and the signature $Sig_{SK_{U_A}}(N)$ to Bob.

④After Bob receives the data, he first calculates the address according to the received ID_A and PK_{U_A} with the elliptic curve signature algorithm secp256k1 and hash algorithm Keccak256 and then searches to determine whether there is such a transaction $TAD \xrightarrow{(attr,tid)} Address$

in the blockchain according to the received attribute and taskid. Because the transactions in the block are sorted by tid, by first looking for tid and then Address, the lookup is fast. If the transaction exists, it proves that Alice can apply for such a task. Then, it can be verified whether subset S meets the access policy P, and if so, the signature $Sig_{SK_{U_A}}$ can be verified with the received public key PK_{U_A}. If the verification is successful, then it is proved that it is Alice; Alice's task request is allowed.

⑤Bob searches the status, conditions, privileges, and so on of the task in the blockchain to assign Alice the minimum privileges that can perform the task. This transaction information is also sorted by tid, so the search is also very fast.

⑥Bob returns the task execution permission and the minimum privilege to execute the

task to Alice.

After searching the status and conditions of the task in the blockchain, Bob will check the privileges to execute the task according to them. For example, Alice is not assigned a privilege to execute a task when the task is in the ready state or invalid state. When the task is in the active state, it will check whether the condition is satisfied, such as whether the previously completed steps required by the task are completed, and reassign the privilege to Alice. If the task condition only satisfies the privilege to read a specific file, it will not assign more privileges to read other files. Alice performs tasks with this privilege. After the task enters the execution state, it will check whether the condition of the task status has changed in real time. If the task enters the invalid state at this time, all Alice's privileges for this task will be revoked immediately.

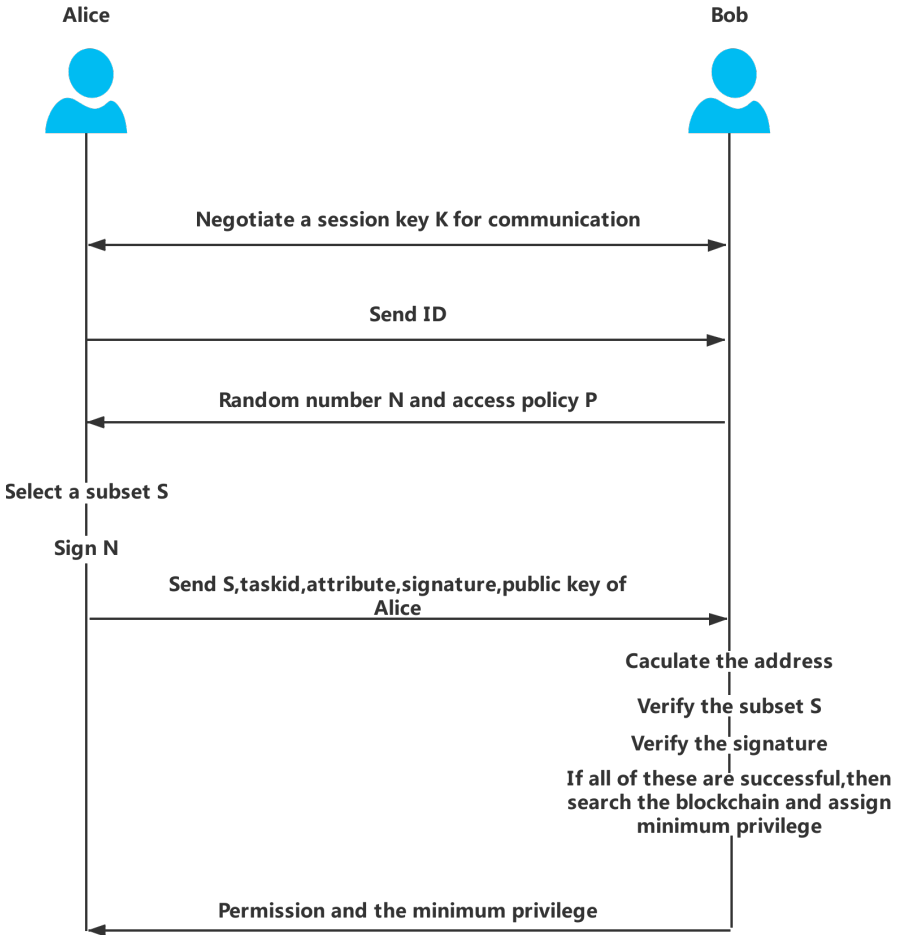

Figure 2: The implementation of access control between Alice and Bob

5 Security analysis

5.1 Ultra vires attack resistant

The scheme we have put forward can efficiently resist Ultra vires attacks. Traditional

access controls and attribute-based access controls and other systems first give the requester certain attributes or roles and other information, which contain certain privileges. These privileges are predefined. When the access control request passes, the requester can execute the privileges he or she has, which may include some privileges other than the requested resource. A malicious requester can attack the resource and other information using such privileges. However, the scheme proposed in this paper dynamically assigns the privileges of requesters according to the status of tasks, required conditions and other information. Requesters only have the minimum privileges to access resources. Therefore, malicious requesters have no additional privileges to take advantage of. Hence, our proposed scheme is Ultra vires attack resistant.

5.2 Man-in-the-middle attack resistant

There may be some malicious users in the traditional IoT, who can use a man-in-the-middle attack to obtain the data and resources they want. In our scheme, even malicious users cannot obtain any useful information by using a man-in-the-middle attack. For example, Alice sends a task request to Bob, and a malicious user Charlie intercepts the taskid sent by Alice to Bob, the corresponding attribute, the public key PK_{U_A} and the signature $Sig_{SK_{U_A}}(N)$ through a man-in-the-middle attack. Then, Charlie sends this information to Bob. Bob calculates $Keccak256(ID_C \| PK_{U_A})$. Bob then removes the last 40 letters, adds 0x to the beginning, and searches the Address in the blockchain. If the two addresses are not equal, Bob knows that Charlie is not Alice and refuses the request, as shown in Fig. 3. Hence, our proposed scheme is man-in-the-middle attack resistant.

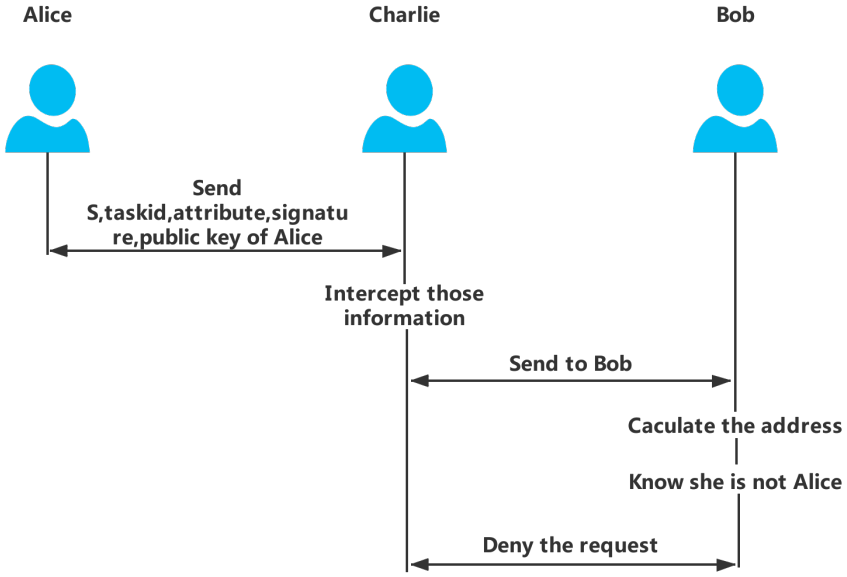

Figure 3: Man-in-the-middle attack Resistant

6 Implementation of our model and performance analysis

To demonstrate the feasibility of our proposed model, we implemented it on two PCs with running Ubuntu16.04.1. A Geth client was built on the computer to connect and interact with the Ethereum network. We used the Geth client to build a consortium blockchain on these PCs. We used the solidity language to write the smart contract code and successfully achieved access control in the model.

6.1 Time overhead

In our proposed model, when users make task requests, they can check whether they can apply according to the tasks returned by TAD, so users can judge their rights very quickly. In addition, all transactions are sorted according to taskid when publishing transactions. Users can first look up the address, task status, conditions and other information by taskid. In the traditional blockchain-based access control scheme, to find the permissions of the requester, it is necessary to compare the addresses one by one in the blockchain. Therefore, compared with the traditional scheme, the proposed scheme is more efficient and takes less time, as shown in Tab. 1.

Table 1: The comparison of Time Overhead

	Our model	Traditional scheme
Time Overhead	Find the results more quickly because of tid	Look up the address one by one

6.2 Space overhead

Due to the limited resource storage of devices in the IoT, it is necessary to analyze the spatial overhead of our proposed model. The main parameters in the model are the basis point G in the elliptic curve signature algorithm secp256k1, the public and private keys of TAD and IoT users, and the session key K negotiated between users. Since the public and private keys of the TAD are determined according to the number of regions in the IoT, the public and private keys of IoT users are determined according to the number of users in the IoT, and both the basis point G and the public and private keys are generated according to the fixed elliptic curve signature algorithm, secp256k1, and their sizes are acceptable for IoT devices. The session key K, between users, is determined according to the number of access control requests established between users, and the session key K will be destroyed when each access control request is over; so the session key K is also acceptable for this model.

6.3 Computation overhead

The main computational overhead of the proposed model is from the secp256k1 and Keccak256 algorithms. These two algorithms are also used in Ethereum. Because secp256k1 is constructed in a special nonrandom way, it can achieve particularly efficient calculations, so it is 30% faster than other curves algorithms. Keccak256 effectively reduces the amount of computing that Ethereum can perform on large smart contracts. Additionally, these two algorithms are used in Ethereum. The solidity code used in the

model can quickly implement these two algorithms, so the computational cost of the model is acceptable.

7 Conclusion

In this paper, we proposed a new task-attribute-based access control model combined with blockchain to improve the security of IoT systems. We explained that this model assigns user's privileges dynamically. It also solves the single point of failure problem because of its integration with blockchain. The system is decentralization, making it more secure than other systems. The security analysis performed in this paper also proves the enhanced security of our proposed model in IoT systems. We successfully implemented the model using Ethereum and smart contracts and proved the feasibility of the model. A performance analysis demonstrated that the model is acceptable for IoT devices.

Acknowledgment: The authors would like to thank the reviewers for their detailed reviews and constructive comments, which have helped improve the quality of this paper.

Funding Statement: This work was supported in part by the National Key Research and Development Project of China (No. 2017YFB0802302), the Science and Technology Support Project of Sichuan Province (No. 2016FZ0112, No. 2017GZ0314, No. 2018GZ0204), the Academic and Technical Leaders Training Funding Support Projects of Sichuan Province (No. 2016120080102643), the Application Foundation Project of Sichuan Province (No. 2017JY0168),the Science and Technology Project of Chengdu (No. 2017-RK00-00103-ZF, No. 2016-HM01-00217-SF).

Conflicts of Interest: We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

Aljosha, J.; Nicholas, S.; Katharina, K.; Edgar, W. (2017): Blocks and chains: introduction to bitcoin, cryptocurrencies, and their consensus mechanisms. *Morgan & Claypool*, vol. 9, no. 1, pp. 1-123.

Beltran, V.; Skarmeta, A. F. (2019): Overview of device access control in the IoT and its challenges. *IEEE Communications Magazine*, vol. 57, no. 1, pp. 154-160.

Couto da Silva, F. J.; Bro Damsgaard, S.; Mousing Sorensen, M. A.; Marty, F.; Altariqi, B. et al. (2019): Analysis of blockchain forking on an ethereum network. *European Wireless*, pp.1-6.

Conoscenti, M.; Vetrò, A.; De Martin, J. C. (2016): Blockchain for the Internet of Things: a systematic literature review. *IEEE/ACS 13th International Conference of Computer Systems and Applications*, Agadir, pp. 1-6.

Ding, S.; Cao, J.; Li, C.; Fan, K.; Li, H. (2019): A novel attribute-based access control scheme using blockchain for IoT. *IEEE Access*, vol. 7, pp. 38431-38441.

Gusmeroli, S.; Piccione, S.; Rotondi, D. (2012): IoT access control issues: a capability based approach. *Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Palermo, pp. 787-792.

Hardt, D. (2012): The oAuth 2.0 authorization framework, document RFC 6749. https://rfc-editor.org/rfc/rfc6749.txt.

Hammi, M. T.; Hammi, B.; Bellot, P.; Serhrouchni, A. (2018): Bubbles of trust: a decentralized blockchain-based authentication system for IoT. *Computers & Security*, vol 78, pp. 126-142.

Kim, D. Y.; Min, S. D.; Kim, S. (2019): A DPN (delegated proof of node) mechanism for secure data transmission in IoT services. *Computers, Materials & Continua*, vol. 60, no. 1, pp. 1-14.

Lu, Y.; Zhang, L.; Sun, J. (2008): Types for task-based access control in workflow systems. *IET Software*, vol. 2, no. 5, pp. 461-473.

Liu, S. (2010): Task-role-based access control model and its implementation. *2010 2nd International Conference on Education Technology and Computer*, Shanghai, vol. 3, pp. V3-293-V3-296.

Medhane, D. V.; Sangaiah, A. K.; Hossain, M. S.; Muhammad, G.; Wang, J. (2020): Blockchain-enabled distributed security framework for next generation IoT: an edge-cloud and software defined network integrated approach. *IEEE Internet of Things Journal*, pp. 1, https://doi.org/10.1109/JIOT.2020.2977196.

Nakamoto, S. (2008): Bitcoin: A peer-to-peer electronic cash system tech. Rep.

Oh, S. R.; Kim, Y. G.; Cho, S. (2019): An interoperable access control framework for diverse IoT platforms based on oAuth and role. *Sensors*, vol. 19, no. 8, pp. 1884-1884.

Ouaddah, A.; Mousannif, H.; Ait Ouahman, A. (2015): Access control models in IoT: the road ahead. *IEEE/ACS 12th International Conference of Computer Systems and Applications*, Marrakech, pp. 1-2.

Ouaddah, A.; Abou Elkalam, A.; Ait Ouahman, A. (2016): FairAccess: a new blockchain-based access control framework for the Internet of Things. *Security and Communication Networks*, vol. 9, no. 18, pp. 5943-5964.

Ren, Y. J.; Zhu, F.; Sharma Pradip, K.; Wang, T.; Wang, J. et al. (2020): Data query mechanism based on hash computing power of blockchain in Internet of Things. *Sensors*, vol. 20, no. 1, pp. 207-207. doi: 10.3390/s20010207.

Ren, Y. J.; Liu, Y. P.; Ji, S.; Sangaiah, A. K.; Wang, J. (2018): Incentive mechanism of data storage based on blockchain for wireless sensor networks. *Mobile Information Systems*, vol. 2018, pp. 6874158:1-6874158:10.

Riabi, I.; Ayed, H. K. B.; Saidane, L. A. (2019): A survey on blockchain based access control for Internet of Things. *15th International Wireless Communications & Mobile Computing Conference*, pp. 502-507, Tangier, Morocco.

Riabi, I.; Dhif, Y.; Ben Ayed, H. K.; Zaatouri, K. (2019): A blockchain based access control for IoT. *15th International Wireless Communications & Mobile Computing Conference*, pp. 2086-2091, Tangier, Morocco,

Shahzad, B.; Crowcroft, J. (2019): Trustworthy electronic voting using adjusted blockchain technology. *IEEE Access*, vol. 7, pp. 24477-24488.

Uddin, M.; Islam, S. (2019): A dynamic access control model using authorizing workflow and task-role based access control. *IEEE Access*, vol. 7, pp. 166676-166689.

Vujičić, D.; Jagodić, D.; Ranđić, S. (2018): Blockchain technology, bitcoin, and ethereum: a brief overview. *17th International Symposium Infoteh-jahorina*, pp. 1-6. East Sarajevo.

Wan, W.; Chen, H.; Chen, J.; Zhang, S.; (2019): Side channel security analysis for blockchain elliptic curve cryptography algorithm. *Journal of Applied Sciences*, vol. 37, no. 2, pp. 203-212.

Xu, R.; Chen, Y.; Blasch, E.; Chen, G. (2018): BlendCAC: a blockchain-enabled decentralized capability-based access control for IoTs. *IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 1027-1034, Halifax, NS, Canada.

Yin, B.; Wei, X. T. (2018): Communication-efficient data aggregation tree construction for complex queries in IoT applications. *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3352-3363.