Song *et al. J Wireless Com Network* (2021) 2021:105
https://doi.org/10.1186/s13638-021-01986-4

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

An access control model for the Internet of Things based on zero-knowledge token and blockchain

Lihua Song, Xinran Ju*, Zongke Zhu and Mengchen Li

*Correspondence:
744636023@qq.com
College of Information,
North China University
of Technology,
Beijing 100043, China

Abstract

Information security has become a hot topic in Internet of Things (IoT), and traditional centralized access control models are faced with threats such as single point failure, internal attack, and central leak. In this paper, we propose a model to improve the access control security of the IoT, which is based on zero-knowledge proof and smart contract technology in the blockchain. Firstly, we deploy attribute information of access control in the blockchain, which relieves the pressure and credibility problem brought by the third-party information concentration. Secondly, encrypted access control token is used to gain the access permission of the resources, which makes the user's identity invisible and effectively avoids attribute ownership exposure problem. Besides, the use of smart contracts solves the problem of low computing efficiency of IoT devices and the waste of blockchain computing power resources. Finally, a prototype of IoT access control system based on blockchain and zero-knowledge proof technology is implemented. The test analysis results show that the model achieves effective attribute privacy protection, compared with the Attribute-Based Access Control model of the same security level, the access efficiency increases linearly with the increase of access scale.

Keywords: Internet of Things, Blockchain, Access control, Zero-knowledge proof

1 Introduction

With the development of IoT devices, more and more important information is generated, including personal or corporate privacy information. Lack of trust in privacy will lead to a decline in user recognition [1], and the low computing power of traditional IoT devices makes them more vulnerable to attacks compared with Internet devices. For example, in the past two years, there have been frequent candid incidents in some hotels and hostels, such as Taitang, Airbnb to Westin Hotel and Crowne Plaza hotel, which makes people who value privacy unbearable. In 2019, the Ring, a home surveillance camera owned by Amazon, was exposed as a security breach. Hackers could monitor users' homes, and the Ring would also expose their WiFi passwords. In June 2018, a 14-year-old hacker took control of a server after using a malware called Silex to trick up to 4000 insecure IoT devices. It could be seen that IoT

devices leave an opportunity for attackers due to the lack of secure access control measures and their security is seriously threatened. IoT access control security has increasingly become a focus of research.

As a key technology in the field of information security, access control technology plays an important role in resisting the malicious access of attackers. However, the disadvantage of the traditional access control model lies in the need for a central entity for information management. The problem of this approach is that the central entity is not completely trusted and there is a risk of disclosure. Besides, a single central entity is vulnerable, if the central entity is breached, it will also cause incalculable losses to users, so a decentralized access control model is needed to solve this problem.

The emergence of blockchain technology has effectively solved this problem. The blockchain consensus mechanism can be used to create a trusted distributed architecture, which can realize the registration, management, authentication, and authorization of IoT devices in an untrusted environment without relying on a third party, thus solving the hidden dangers of information security and single point of failure brought by the traditional centralized access control model. Most of the existing access control models combining blockchain with the IoT have solved the problem of untrusted of the central entity of the IoT and effectively dealt with the problem of unauthorized access. However, there are still many shortcomings in the security of authentication. Based on blockchain technology, digital currency coins as an example, the user identity is a part of the deal after a hash encryption blockchain address rather than a true identity, the currency was initially thought to be anonymous, but it turns out that its privacy is not high, because all transaction information is publicly on the blockchain, through the analysis of trade value and the date in the chart, books, coins may present address associated with the identity of the real world [2]. Some people engaged in privacy research have developed several powerful libraries of heuristic tools that allow attackers to link different bitcoins transactions to an ordinary user, and in many cases, to the user's real identity [3]. This has led to the design of some new cryptocurrencies whose primary focus is user privacy, such as Zcash. The basic principle of Zcash is zero-knowledge succinct non-interactive arguments of knowledge (zk-SNARKs), also known as zero-knowledge proof (ZKP) technology. A zero-knowledge proof is a problem proposed by S. Gesdwasser, S. Marks, and C. Ackoff in the early 1980s, which refers to the fact that the verifier makes the verifier believe that a certain deduction is correct without providing any valuable information. If zero-knowledge proof is applied to information verification, it will bring qualitative changes to many existing theories. Kouicem, Djamel Eddine et al. analyzed the security problems of the IoT and proposed that zero-knowledge proof is one of the most powerful solutions to protect the privacy of the IoT [4].

The main contributions of this paper can be summarized as follows: The emergence of network attacks targeting a single node makes the traditional centralized network structure not fully trusted. According to our model, the security hidden danger of a single node will not be able to threaten the information stored on the network.We use Ethereum smart contract on the basis of the Attrition-based Access Control Model, and make use of the high computing power characteristics of smart contract for access control.The Groth16 algorithm in zero-knowledge proof has strong security and high efficiency due to its non-inverse derivation and small amount of required data. We design a

special token for access control, which can not only improve the efficiency of access, but also realize the concealment of private information.

The following main work of this paper is as follows: The second section analyzes the current security problems of the IoT and the solutions proposed for these problems, and makes improvements on these methods; The third section introduces the BZBAC model and related concepts. In the fourth section, the implementation method of the BZBAC model is introduced. In the fifth section, the performance of the model is analyzed and tested. Our work is summarized at the end of the paper.

2 Related work

In recent years, the blockchain has been preliminarily applied and practiced in IoT access security. Ouaddah A. et al. introduced FairAccess as a new distributed privacy protection access control framework in the Scene of the IoT, combining access control model and cryptocurrency blockchain mechanism for the first time [5]. Ying M. takes advantage of the non-tamper-proof feature of the bitcoins blockchain to record access rights and other information on the chain, and proposes an IoT access control model based on the bitcoins platform [6].

After people found that blockchain is suitable for the Internet of Things, there are a lot of studies and improvements on the architecture of the Internet of Things based on blockchain. Wang G. et al. proposed a blockchain-based IIoT architecture to support immutable and verifiable services, and layered blockchain storage structure to solve the storage problem [7]. For the security of the Internet of Things, Xu R.et al. proposed a blockchain-enabled decentralized capability-based AC [8]. To further leverage the superiority of combining blockchain and crowdsourcing, Zhu et al. proposed an innovative hybrid blockchain crowdsourcing platform, named zkCrowd [9]. Yuanyu et al. proposed a distributed trusted access control framework composed of multiple access control contracts, a judge contract, and a registration contract to implement the IoT system [10]. Xian-li et al. designed secure storage and authorized access model of private information by combining the IPFS protocol with blockchain [11]. Yuan et al. applied blockchain technology to intelligent transportation and proposed a seven-layer conceptual model [12]. In the face of the concerns about the privacy and confidentiality of the collected information brought about by the intrusion of sensors and communication devices, Pinno et al. proposed an architecture based on blockchain for IoT access authorization [13].

However, although the use of blockchain can solve many defects and security problems of traditional Internet of Things architecture, people often ignore the security risks of blockchain itself. Some attacks may not be applicable to traditional network structures, but the impact on blockchain networks should not be underestimated. For example, pool-hopping attack, which attacks cost-efficient and straightforward, easily poses a threat to concentrated mining. Shi H. et al. proposed a hopping-proof pooled mining with fee-free in Blockchain, and applied the zero-determinant theory to design a novel pooled mining which offers an incentive mechanism for motivating non-memorial and memorial evolutionary miners not to switch in pools strategically [14]. A well-known attack in Blockchain is the forking attack, where divergent blockchains are produced for inserting some new features to facilitate security breaches. To take precautions, Wang S.

et al. employed the large deviation theory to study the vulnerability of blockchain networks incurred by intentional forks from a micro point of view, boosting forward-looking and strategic planning mechanisms for resisting the forking attack [15]. In Bitcoin's incentive system that supports open mining pools, block withholding attacks incur huge security threats. Hu Q. et al. take advantage of the Zero-Determinant strategy to analyze the block withholding attack between any two pools, where the Zero-Determinant adopter has the unilateral control on the expected payoffs of its opponent and itself [16].

It can be seen that the untamable and open and transparent characteristics of blockchain make it perfectly combined with the access control of the IoT. But because of this, people cannot write private information into the blockchain, which greatly limits its scalability. Therefore, many people also proposed to achieve blockchain anonymous access or other ways to protect private information. Because the server can clear data from a device, such as Zhou et al. designed a decentralized outsourcing computing scheme, the server can calculate the encrypted data from the data owner according to the request of the data owner, to detect the dishonest server while protecting the data privacy, reduce the risk of leak sensitive information [17]; Hardjono et al. proposed a ChainAchor system to provide an anonymous but verifiable identity for entities on the blockchain [18]; Some people also raised doubts about the anonymous network communication. Henry R. et al. showed through research that the general anonymous communication system like Tor could not solve the communication privacy problem [19], which brought new attention to the privacy security of the blockchain. Cai et al. proposed a novel mechanism for data uploading in smart cyber-physical systems, which considers both energy conservation and privacy preservation. The mechanism preserves privacy by concealing abnormal behaviors of participants, while still achieves an energy-efficient scheme for data uploading by introducing an acceptable number of extra contents [20].

From the perspective of zero-knowledge proof, some other people provide solutions for the security of access control. Khandavilli et al. proposed A security framework based on identity-based encryption, using zero-knowledge proof encryption to provide authentication and information security [21]. Yang introduced zk-SNARKs into the existing identity declaration model and designed methods for secret transfer of privacy attributes and authentication of attribute ownership to protect identity privacy [22].

At present, although there have been a lot of studies on the implementation of IoT access control by using blockchain and the exploration of its anonymous access, there are still the following problems in many designs: (1) The way of changing permissions is complex and it is difficult to achieve fine-grained control; (2) Huge amount of computing, and the blockchain's computing power is not fully utilized, resulting in large consumption of resources; (3) Due to the low computing power of IoT devices, most strategies are difficult to be promoted in practice; (4) Many anonymous access methods are difficult to integrate with ethereum smart contracts.

3 Methods

3.1 System model

To solve the above problems, this paper proposes a BZBAC (Blockchain and Zero-knowledge Token-Based Access Control) model. This model: (1) mainly manages access control through fine-grained attribute information, (2) utilizes ethereum smart contracts

for policy management, and designs zero-knowledge access tokens to improve access efficiency and reduce the computational pressure and time cost of the blockchain, (3) uses IoT gateway proxy devices to enhance the applicability of policies, and (4) uses the idea of off-chain computation and on-chain proof to further reduce the computational pressure on the chain and reduce the difficulty of implementing anonymous access.

According to the above description, the structure diagram of the model is shown in Fig. 1. The resource owner publishes the unique identifier and object property information of the device on the blockchain as the basis of fine-grained access control authority decision, stores it through a smart contract, and records the address of the property owner. When a user needs to authenticate the device's attribute information to access a private resource, the identity can be retrieved by invoking the smart contract authentication identifier to indicate ownership of the attribute. Then the smart contract verifies the attribute information to determine whether the user has the right to access the resource.

The token is introduced in the BZBAC model as an alternative to accessing subject information. When the attribute information is registered, the registration point will mark the subject identifier for the attribute information of the subject, and prove the ownership of the attribute through the digital signature generated by zero-knowledge proof. After the attribute information is written to the blockchain, each time the

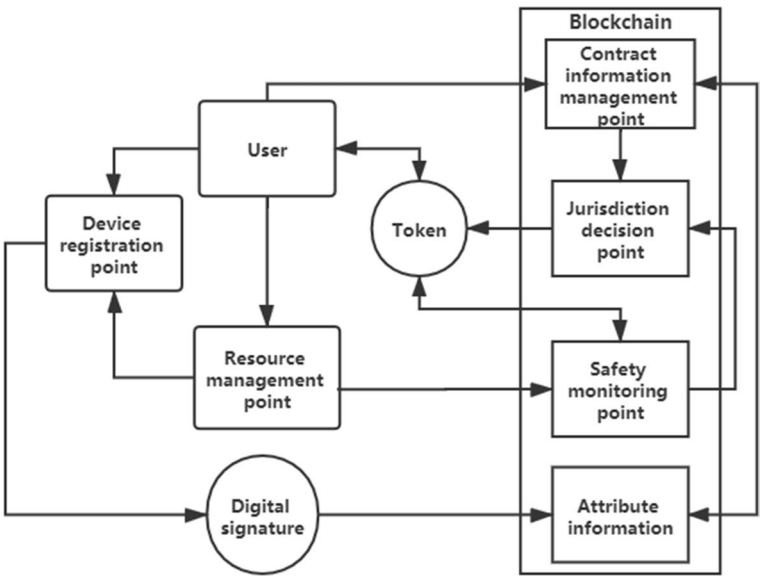

Fig. 1 Access control model. The resource owner publishes the unique identifier and object property information of the device on the blockchain as the basis of fine-grained access control authority decision, stores it through a smart contract, and records the address of the property owner. When a user needs to authenticate the device's attribute information to access a private resource, the identity can be retrieved by invoking the smart contract authentication identifier to indicate ownership of the attribute. Then the smart contract verifies the attribute information to determine whether the user has the right to access the resource. When the attribute information is registered, the registration point will mark the subject identifier for the attribute information of the subject, and prove the ownership of the attribute through the digital signature generated by zero-knowledge proof. After the attribute information is written to the blockchain, each time the attribute information is called, the source of the attribute is confirmed by verifying the signature. The result of the permission ruling will return the subject a ZKToken encrypted by zero-knowledge proof in place of the subject's identity information for legitimate access to the resource, which records the subject's access rights and valid time

attribute information is called, the source of the attribute is confirmed by verifying the signature. The result of the permission ruling will return the subject a ZKToken (zero-knowledge token) encrypted by zero-knowledge proof in place of the subject's identity information for legitimate access to the resource, which records the subject's access rights and valid time.

Visitors due to using the zero-knowledge proof encrypted signature instead of blockchain address release properties, using ZKToken instead of visitor's identifier for access to resources, the subject of identity information and address blockchain link is not visible to other subjects, The blockchain address of the subject cannot be traced through the information on the blockchain, more can't get access to the main body of real information.

3.2 Entity

Entities are collections of devices that are actually involved in access control in the access control model, connected together via Ethernet and blockchain. The main entities used in the model are as follows:

A) Access device: The access device refers to the subject that initiates the access request during the resource access and has the read permission of the blockchain. Use $Subject = \{s_1, s_2, \ldots, s_n\}$ to represent a collection of access devices.

B) Resource equipment: Resource equipment refers to the objects to be accessed during resource access, such as accessible programs and controllable hardware equipment. Use $Object = \{o_1, o_2, \ldots, o_n\}$ to represent a collection of resource devices.

C) Resource owner: the resource owner is the gateway agent, the entity that owns the above resources and administrative authority. Use $Resource = \{r_1, r_2, \ldots, r_n\}$ to represent a collection of Resource owner.

D) Permission: Permission is the set of entities to which the user operates on the resources, via $Permission = \{p_1, p_2, \ldots, p_n\}$ represents the permissions in the access control model, including the reading, writing, deleting of data and the operation of IoT devices.

E) Token: Token is a collection of entities granted by the access control strategy to the user, through $Token_{ij}(a,p,t_s,t_e) = \{token_{ij}(a,p,t_s,t_e) | i \in S(t_s),\ j \in O(t_s), p \in P\}$ denotes that collection. $token_{ij}(a,p,t_s,t_e)$ denotes that the user s_i who satisfies the attribute set a from t_s to t_e has the operation authority p for the resource body o_j [23].

F) ZKToken: Access token whose holder information is generated by zero-knowledge proof method and is represented as "zero-knowledge".

G) Attribute: Attribute is an abstraction of things in the IoT from the perspective of access control and the extracted properties related to access control. Use $Attribute = \{a_1, a_2, \ldots, a_n\}$ to represent a collection of the set of attributes.

H) UUID: During the registration process, the system generates a universally unique identifier for each device as its own identity.

I) Fine-grained access control: Fine-grained access control refers to access control that manages permissions on the data level by subdividing objects in the model. Compared with coarse-grained access control, Fine-grained access makes the granting of permissions more reasonable and flexible.

Attributes based fine-grained access control will call a variety of properties related to the resource to determine when the access request is received, and give corresponding permissions according to the decision result. Different from coarse-grained models such as role-based access control, fine-grained access control model can effectively reduce the problem of excessive authorization and make policy changes more freely.

J) Access policy: The access policy is created by the resource owner and published on the blockchain through transactions. The access subject's authorization to initiate relevant operations on the resource is determined by the corresponding access policy.

K) subject attribute: The subject attribute refers to the attribute inherent in the resource access subject and related to access control. The collection of subject attributes is represented by an *SA*. The set of attributes for a subject si is represented by s_i. $a_j = \{a_j | a_j s_i, SA\}$.

L) Resource attribute: The attribute of a resource refers to the attribute inherent in the object being accessed and related to access control. The collection of resource properties is represented by *OA*. The attribute set of oi for a resource is represented by o_i. $a_j = \{a_j | a_j o_i, OA\}$.

M) AccessLog: AccessLog is the historical access behavior and related information record of users in the system. $AccessLog(s,o,t) = \{al_1, al_2, \ldots, al_n\}$ represents a collection of session histories. The unchangeable nature of the content on the blockchain ensures that every access is accurately recorded.

3.3 Equipment registration

Access subjects and resource devices both need to be registered on the blockchain through the device registration point for broadcast, and written into the blockchain using the blockchain consensus mechanism. The device registration point itself does not have the ability to store data.

The purpose of subject registration is to obtain the attributes and UUID assigned through the device registration points and to upload relevant information to the Smart Contract. subjects and resource owners can get the attributes owned by users by calling the smart contract *GetAttribute(s)* function.

Resource registration is similar to subject registration in that its purpose is to obtain UUID and attributes allocated by device registration points and upload access policies to smart contracts.

3.4 Access control implementation

Most IoT devices do not have the computing power to execute authority decisions. Therefore, IoT gateways with certain computing power and the ability to interact with authority decision points are not only owned as resources in the BZBAC model, but also as access control enforcement points in the access process.

When the subject wants to access a resource device, it cannot communicate directly with the terminal device resource, but indirectly interacts with the access control enforcement point. The subject sends an access control request to the access control

enforcement point, and the access control enforcement point interacts with the access control decision point. After the permission is passed, a token is generated, and then the access control enforcement point interacts with the device resource, and the access result and token are returned to the subject.

The access subject can also directly access the decision point of permission and reduce the time cost of access control by applying tokens in advance.

3.5 Jurisdiction ruling

Permission decision points can receive access control requests from access control enforcement points and access subjects. The authority award point first invokes the smart contract to obtain the relevant information of the subject and resources and checks the security monitoring module to determine whether the subject is in the state of punishment. It determines that the authority award will be made by invoking the policy contract. If the adjudication is passed, a token will be generated and the result of the award and token will be returned to the information sender.

3.6 Safety monitoring

In the access control model, a security monitoring module is set up, which can better supervise user behavior and punish malicious behavior. The access object has an access log, and the blockchain will record the user's access process behavior and will be punished when the user has an improper or malicious operation. When security exceptions such as users' unauthorized behaviors or malicious collusion among users are detected, the system will take certain Punishment measures [24] and record the Punishment information through the *Punishment(s)* function.

3.7 Zero-knowledge proof

Zero-knowledge Proof is when a person makes the verifier believe an argument, but gets nothing of value other than the information that the argument is correct. The three characteristics of zero-knowledge proof are correctness, completeness, and zero-knowledge. Correctness: If the argument is not certain to be true, then the verifier has difficulty believing the argument; Completeness: If the argument is correct, then the verifier has an absolute reason to trust the argument; Zero-knowledge: The verifier cannot acquire any additional knowledge.

At present, zk-SNARKs are mainly Groth16[25], Sonic[26], and Marlin[27], etc. Given the small amount of Groth16 proof data and the advantages of high running speed, the zero-knowledge proof method in this paper is mainly implemented based on Groth16. The specific implementation process will not be explained here, only the main steps will be listed.

The process of zero-knowledge proof is as follows:

Randomly generate $\alpha, \beta, \gamma, \delta, x$ on the cyclic subgroup *Fr* of the elliptic curve, and generate set τ, σ.It should be noted that α, β, γ *and* δ represent polynomials, and x is a random number. Among them:

$$\tau = (\alpha, \beta, \gamma, \delta, x), \tag{1}$$

$$\sigma = ([\sigma_1]_1, [\sigma_2]_2), \tag{2}$$

$$\sigma_1 = \left(\alpha, \beta, \delta\{x^i\}_{i=0}^{n-1}, \left\{\frac{\beta u_i(x) + \alpha v_i(x) + w_i(x)}{\gamma}\right\}_{i=0}^{l}, \left\{\frac{\beta u_i(x) + \alpha v_i(x) + w_i(x)}{\delta}\right\}_{i=l+1}^{m}, \left\{\frac{x^i t(x)}{\delta}\right\}_{i=0}^{n-2}\right), \tag{3}$$

$$\sigma_2 = \left(\beta, \gamma, \delta\left\{x^i\right\}_{i=0}^{n-1}\right). \tag{4}$$

The process of generating the proof is to randomly select two parameters r and s, calculate $\pi = \Pi\sigma = ([A]_1, [C]_1, [B]_2)$. Among them:

$$A = \alpha + \sum_{i=0}^{m} \alpha_i u_i(x) + r\delta, \tag{5}$$

$$B = \beta + \sum_{i=0}^{m} \alpha_i u_i(x) + s\delta, \tag{6}$$

$$C = \frac{\sum_{i=l+1}^{m} a_i(\beta u_i(x) + \alpha v_i(x) + w_i(x)) + h(x)t(x)}{\delta} + As + rB - rs\delta \tag{7}$$

In the above formula, the generated results are the three points on the elliptic curve (Point B needs to be calculated twice, and $H(x)$ is calculated by the QAP equation), and then the three points are handed over to the verifier for verification. The verification formula is:

$$A \cdot B = \alpha \cdot \beta + \frac{\sum_{i=0}^{l} a_i(\beta u_i(x) + \alpha v_i(x) + w_i(x))}{\gamma} \cdot \gamma + C \cdot \delta. \tag{8}$$

Therefore, the perfect zero-knowledge of the algorithm can be verified by the following formula:

$$C = \frac{AB - \alpha\beta - \sum_{i=0}^{l} a_i(\beta u_i(x) + \alpha v_i(x) + w_i(x))}{\delta}. \tag{9}$$

In the formula used in the above groth16 algorithm, we do not need to entangle the specific values represented by each parameter (because many parameters are randomly generated), only need to prove that in the case of providing τ and public key (a_0,a_1,...,a_l), a verifiable proof π can be obtained through calculation even if the private key information (a_{l+1},a_{l+2},...,a_m) is not known.

3.8 ZoKrates

Ethereum runs computations on all nodes of the network, resulting in high costs, limits in complexity, and low privacy. In 2018 Jacob Eberhardt and Stefan Tai, two doctoral students at the Polytechnic University of Berlin, Germany, proposed a framework for

off-chain computing/on-chain validation and provided a tool for the entire framework on ethereum.

ZoKrates supports the outer chain processing model described above by using zk-SNARKs as a proof system, which defines a domain-specific language that allows developers to easily specify outer chain calculations at a high level of abstraction. This allows them to specify demonstrable computations without having to understand the low-level programming abstractions that justify the system. To do this, ZoKrates include a compiler that converts domain-specific code into demonstrable constraint systems. To facilitate the implementation of on-chain validation, ZoKrates supports export validation of smart contracts, verifying proofs generated off-chain, thus confirming the correctness of out-of-chain calculations [28].

In this model, ZoKrates implements encryption validation in a black-box manner.

4 Implementation of the model

4.1 Network topology structure

Network topology structure is shown in Fig. 2.

The addresses of UUIDA and UUIDAgents accessing the device and gateway agents are stored in the local PC database, managed by the Web module configuration. After

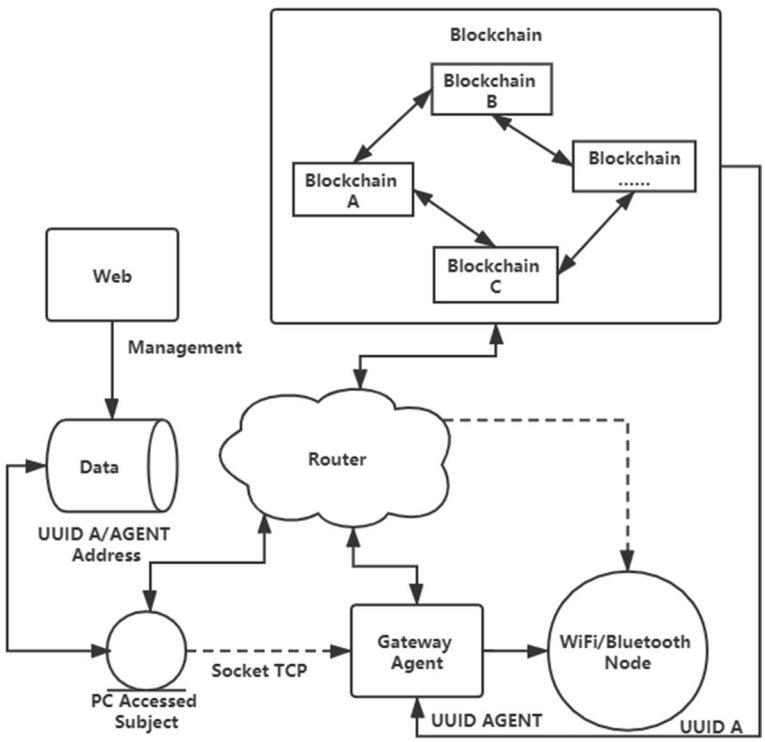

Fig. 2 Network topology structure. The addresses of UUIDA and UUIDAgents accessing the device and gateway agents are stored in the local PC database, managed by the Web module configuration. After the gateway device is started, the Web module begins to listen on the port. When PC performs a read UUIDA operation, it needs to search the IP address of the reading device from the database and establish a socket connection to this address. After receiving the packet, the gateway agent sends the data to the router, which forwards the data to the corresponding node based on the contents of the routing table to invoke the smart contract on the blockchain. The blockchain node starts with an RPC address that is connected to the gateway through web3J's interface to perform the same operations as the client

the gateway device is started, the Web module begins to listen on the port. When PC performs a read UUIDA operation, it needs to search the IP address of the reading device from the database and establish a socket connection to this address. After receiving the packet, the gateway agent sends the data to the router, which forwards the data to the corresponding node based on the contents of the routing table to invoke the smart contract on the blockchain. The blockchain node starts with an RPC address that is connected to the gateway through web3J's interface to perform the same operations as the client.

Among them, the access subject device, the IoT gateway agent device, and the blockchain are not directly communicated, but connected in a LAN through Ethernet. Object resources are connected in different ways depending on the type of device (such as WiFi and Bluetooth, etc.).

4.2 Effectiveness principle of ZKToken

ZKToken is a collection of entities generated by an access control policy through zero-knowledge proof. It has the feature of "zero knowledge", the user can be anonymous in the state to verify the identity authority. The workflow of ZKToken is as follows:

A) An access subject initiates an access request.
B) The agent receives the request and verifies the attribute information of the subject and object through the blockchain.
C) After successful verification, ZKToken is generated through the calculation under the blockchain and sent to the access subject.
D) The access subject receives the ZKToken which can be stored in the access device, and the ZKToken needs to be submitted when requesting resources.
E) The agent receives the request to verify the ZKToken carried in the subject request. By using the original secret key and ciphertext to conduct the same signature operation, and then comparing the generated signature with the signature carried by ZKToken, inconsistency indicates that the original text has been modified, verification fails and error information is returned. If the authentication is successful, the requested resource is returned.
F) Where a validity period is set for ZKToken on the blockchain, and ZKToken and validity period are verified on each request. By decrypting the ZKToken ciphertext, authorization time and validity can be obtained, and whether the ZKToken expires or not can be judged by comparing this with the current time.

The algorithm for generating ZKToken is as follows:

Algorithm 1 Request access permission algorithm

Input : (s,o,r,o.p)

Output : ZKToken

```
1.  RequestAccess (s,o,r,o.p)
2.  if s=msg.sender∧r≠NULL∧o≠NULL∧r=Agent(o) then
3.      If s≠Punishment(s) then
4.      s.a←GetAttributes(s)； ra,pa←s.a；
5.      o.a←GetAttributes(o);
6.          get Decision(s);
7.          If s.a is permitted by o.p then
8.              If o.p is available in {ra,pa} then
9.              Decision(s.a, o.a,s.p)=true then
10.               emit request event;
11.               ZKToken←ZKP(s.a,o.a,s.p);
12.               return token;
13.         else return Access denied;
14.         end if
15.     else return In the punishment;
16.     end if
17. else return Fail verification;
18. end if
```

In Algorithm 1, after the subject initiates the access information quad *(s,o,r,o.p)* to BZBAC, it first judges whether the application information is valid or not. After the judgment passes, the device is checked to see if it is in the penalty phase and the relevant attribute information is retrieved from the blockchain. A decision contract is then invoked to make a permission decision using attribute information, and a ZKToken is returned and passed to the access subject.

For subjects that have obtained the ZKToken, steps 4 to 14 of access will be replaced with verification of the authenticity and validity of the ZKToken, and a Boolean value will be output for the basis of access permission.

4.3 Access process

The access flow is shown in Fig. 3.

The function of each module in the model and implementation process of access control is as follows:

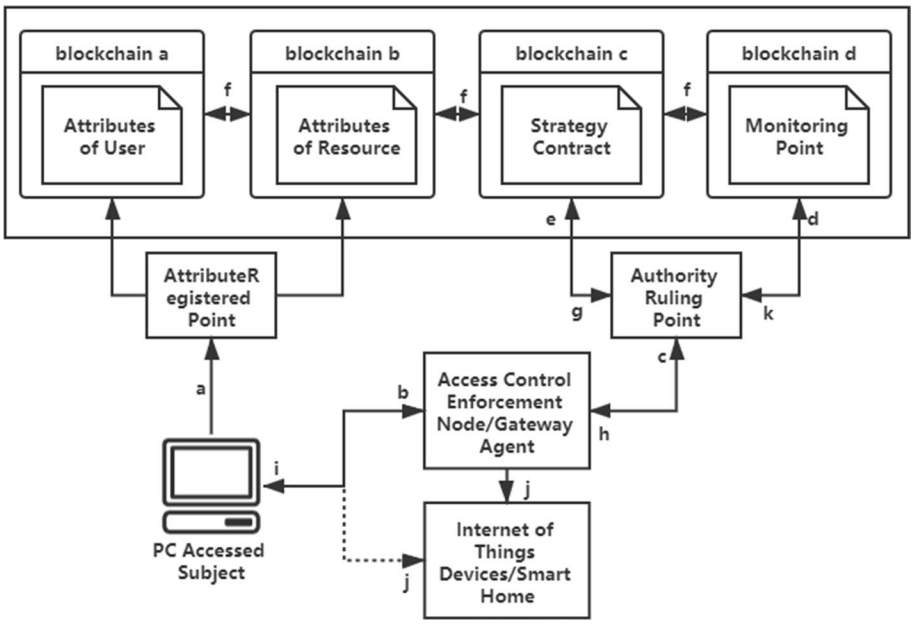

Fig. 3 Access control flow. The function of each module in the model and implementation process of access control is as follows:device registration, submit access information, application for access decision, security monitoring, call contract judgement, inter-contract invocation, generate the ruling result, return the verdict, the token is delivered to the access subject, the subject accesses the resource device through the token, access behavior exception

A) Device registration: The accessing subject registers the attribute information instead of the subject information through the digital signature generated by the zero-knowledge proof, and is written into the blockchain node together with the object resource attribute information to be accessed.

B) Submit access information: When the subject tries to access the resource of the object, it will first issue an access application to the proxy device of the object. The agent determines whether the subject has an access rights token. If there is a token, it goes to step J; otherwise, goes to step c.

C) Application for access decision: After receiving the application, the access decision point visits the security monitoring node to check whether the access information contains the device in the disciplinary time. If the device is in the disciplinary time, it will directly return the error message and refuse the subject to access; If not, a valid message is returned and the following access behavior is recorded.

D) Security monitoring: Monitoring nodes conduct security monitoring on the visit behavior and record the behavior information into the blockchain. At the same time, the monitor node records the valid time of the token.

E) Call contract judgement: Authority decision point obtains authority decision contract, public policy contract, and exclusive policy contract created by the object from contract information management point. The authority decision contract obtains device attributes from the property information management point according to the subject and object device identification; the policy contract makes attribution-based judgment according to the present method; the authority decision is made according to the judgment results of the public policy contract and the exclusive policy contract.

F) Inter-contract invocation: The contract accesses the attribute information by looking up the blockchain address of the attribute information record and also realizes the inter-contract invocation in the same way.

G) Generate the ruling result: Return the ruling result to the permission ruling node after determining the access attribute information of the subject and object according to the contract call. If the decision is not approved, it is not approved and an error message is returned; If the decision passes, an encrypted access token is returned via ZoKrates.

H) Return the verdict: Return the verdict to the gateway agent and record it. If the failed message is passed multiple times, the device will be temporarily denied access.

I) The token is delivered to the access subject: The generated token is delivered to the subject device. The user can hold a token or token stored in the access device, and for the access rights passed, the user can access the resource by validating the token, without re-registering. However, if the user wants to use additional permissions on the same resource device, he needs to re-register to generate a new token to override the token's original information.

J) The subject accesses the resource device through the token: After the user verifies the token, the resource can be legally used within the permission until the token is invalid due to an exception.

K) Access behavior exception: When the subject has abnormal access behavior, the monitoring node will return exception information and generate a new policy to overwrite the original token information so that the token will be invalid. It is necessary to reapply for the next access. The node then handles the access subject based on the exception. Exceptions can be divided into two types: 1) Illegal behaviors occur in user access, such as malicious attacks on resources or unauthorized access attempts. In this case, the node will take corresponding disciplinary measures against the user, such as account closure; 2) If the token exceeds the valid access time, the authorization of the token will be frozen, but the user will be not punished. Instead, a message will be sent back to remind the user to re-register.

5 Experiment

5.1 Time cost

The generation time of zero-knowledge proof trust setting for this experiment is 4.48 s, which is the average of 10 tests. The time of generating zk-SNARKs proof is determined by the set code logic and calculation amount as well as some external environment and other factors, so the setting of this experiment may not apply to all environments. In the calculation of larger circuits, the generation time of zero-knowledge proof trust setting can last minutes or even hours. For different experiments, the processing effect of proprietary code logic is optimal.

The access time cost is shown in Figs. 4 and 5.

It can be seen that at the beginning of the experiment, due to the need for the consensus time of blockchain [29] and the generation time of zero-knowledge proof, the running time of BZBAC is much longer than that of Attribute-Based Access Control (ABAC). However, with the increase in the number of tests, ABAC has been in a state

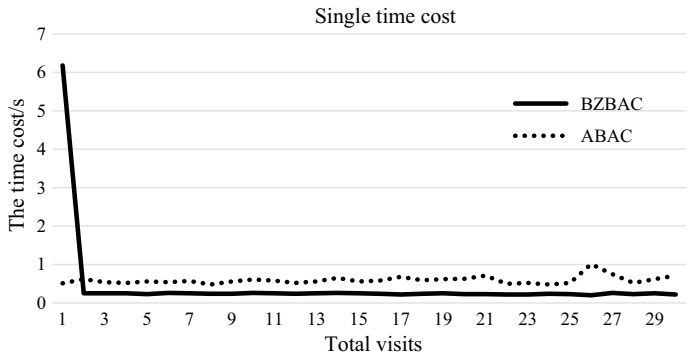

Fig. 4 Single time cost

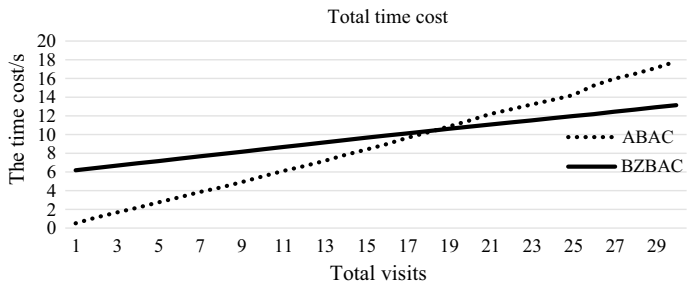

Fig. 5 Total time cost. The experimental results: It can be seen that at the beginning of the experiment, due to the need for the consensus time of blockchain and the generation time of zero-knowledge proof, the running time of BZBAC is much longer than that of Attribute-Based Access Control. However, with the increase in the number of tests, ABAC has been in a state of linear growth. Since BZBAC has completed the generation of ZKToken, subsequent access time is only spent on the verification of ZKToken, so the growth of running time is relatively flat. Although ABAC has a lower cumulative time consumption than BZBAC for previous visits, it gradually has a higher time consumption than BZBAC after the 19th of this access. Thus, the BZBAC model can improve access efficiency and reduce time cost in the case of multiple accesses. If only a few visits are made, the time cost of this model is still relatively high due to the need for trust setting. However, considering reality, users will often visit the same device multiple times rather than just several times. Also, compared with the time required for each visit, users are more inclined to prepare for the subsequent quick access by setting the first time for a longer time. Therefore, in general, the accumulated time cost of BZBAC is reduced and the user's experience effect is improved

of linear growth. Since BZBAC has completed the generation of ZKToken, subsequent access time is only spent on the verification of ZKToken, so the growth of running time is relatively flat. Although ABAC has a lower cumulative time consumption than BZBAC for previous visits, it gradually has a higher time consumption than BZBAC after the 19th of this access. Thus, the BZBAC model can improve access efficiency and reduce time cost in the case of multiple accesses. If only a few visits are made, the time cost of this model is still relatively high due to the need for trust setting. However, considering reality, users will often visit the same device multiple times rather than just several times. Also, compared with the time required for each visit, users are more inclined to prepare for the subsequent quick access by setting the first time for a longer time. Therefore, in general, the accumulated time cost of BZBAC is reduced and the user's experience effect is improved.

Since the decision time is related to the amount of node information processed, in the context of the IoT, the decision time of both the ABAC model and BZBAC model will be synchronously improved, and the time influence of the trusted setting will be greatly reduced. Therefore, in an IoT environment, BZBAC's advantages in time cost are even more obvious.

5.2 Security

Relevant proofs of zero-knowledge proof used in this experiment have been verified in the literature [25]. Therefore, its completeness and zero-knowledge is not explained anymore, and safety analysis is only carried out in combination with the overall model.

Zero-knowledge proof is used to verify the user's authority and has the following characteristics:

A) Correctness: For the equation $P(x) = V$, if $\times 1 \neq \times 2$, $P(\times 1) \neq P(\times 2)$, which means you can't get a correct V using the wrong x.
B) Completeness: For the equation $P(x) = V$, if $\times 1 = \times 2$, $P(\times 1) \equiv P(\times 2)$, which means if you use the right x, you're going to get the right V.
C) Zero-knowledge: For the equation $P(x) = V$, $P^{-1}(V) \nRightarrow x$, which means you can't deduce x from V.

We evaluated the model's performance in the face of common IoT attacks:

A side-channel attack is one of the most difficult attack methods to defend against at present. Its main attack method is to infer private data through malicious nodes in the network and to acquire device energy consumption. There are two ways to counter this attack: restricting access to side-channel information and protecting sensitive data from inference attacks [28]. However, at present, there is no feasible defense mechanism to restrict access to the uncontrollable side-channel, so sensitive data protection is relatively easy to implement at present. In this model, visitors use ZKToken for access, and it is difficult for attackers to distinguish the ownership of access rights. Even if the attribute information in the model is cracked by fetching the packets, it is impossible to know if the attributes are from the same visitor, reducing the risk of access channel exposure. Therefore, the model can improve the resistance of IoT devices to side-channel attacks.

A flood attack is a denial-of-service attack in which an attacker forces a server to shut down by sending a large number of malicious (fake or redundant) packets. Because of the distributed storage structure of blockchain, a flood attack is difficult to threaten the server by attacking a single node. In this model, visitors access resources through tokens, and the fact that zero-knowledge proves difficult to illegally crack greatly increases the cost of attackers. Also, because the visitor's access behavior is stored in the session history, the node that sent the malicious packet will be discovered first, thus freezing access to the token. Therefore, the model can withstand a flood attack.

5.3 Black-box testing

The test equipment has two desktop computers, three laptops, one IoT gateway development board, and two Wi-Fi development boards. The specifications of the computer equipment are shown in Table 1, and the physical diagram of the development board is

Table 1 Computer equipment specifications

Device name	CPU	The operating system	memory
Dell Inspiron 14-7460	IntelCorei7-7500U	Ubuntu-16.04.4	8 GB
Dell Inspiron 15-5543	IntelCorei5-5200U	Ubuntu-16.04.4	8 GB
Dell Inspiron 3650	IntelCorei7-6700	Ubuntu-16.04.4	8 GB
Dell D09M003	IntelCorei7-3770	Window7- × 64	8 GB
Dell D09M004	IntelCorei7-3770	Window7- × 64	16 GB

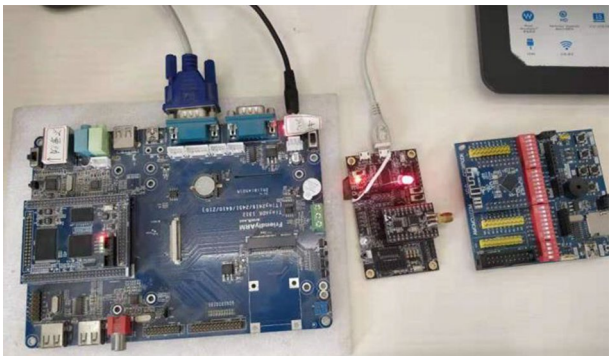

Fig. 6 Physical drawing of the development board. The test equipment has two desktop computers, three laptops, one IoT gateway development board, and two Wi-Fi development boards. The laptop, as the access host device, initiates the resource applies to the object while maintaining the private blockchain network together with the desktop. WiFi development board simulates IoT terminal devices and selects the IoT gateway as a proxy to implement access control. Install ethereum Geth client on two desktops, three laptops, and the IoT gateway development board to convert to ethereum node and join the same private blockchain by configuration. The IoT gateway development board compacts the smart contract for testing through the Remix integrated development environment, and the ethereum node is interactively tested with the Geth client through Web3J

shown in Fig. 6. The laptop, as the access host device, initiates the resource applies to the object while maintaining the private blockchain network together with the desktop. WiFi development board simulates IoT terminal devices and selects the IoT gateway as a proxy to implement access control.

Install ethereum Geth client on two desktops, three laptops, and the IoT gateway development board to convert to ethereum node and join the same private blockchain by configuration. The IoT gateway development board compacts the smart contract for testing through the Remix integrated development environment, and the ethereum node is interactively tested with the Geth client through Web3J. The generated access log information is shown in Fig. 7, and the access results are shown in Table 2.

As is shown in Table 2, when the user uses the light controller to read and write the lighting equipment, the user is allowed to read the state of the equipment and carry out the operation due to the authorization. When the user frequently uses permission to modify the state of the resource, the permission of ZKToken is revoked due to its improper behavior, and access is denied due to invalidated token authentication. When using a light monitor to read and write to a lighting device, the permission attribute in ZKToken does not contain a write operation, so only the permission to read the state of the device can be obtained. When the temperature and controller are

Fig. 7 Object access log. The object access log records all the access information about the device in order to provide some protection for the security of the device

Table 2 The simulation results

Access device	Resource equipment	Operation	Results	Return
Light controller	Lighting equipment	R/C	Y	State of the light
Light controller	Lighting equipment	R/C	N	Token failure
Light monitor	Lighting equipment	R/C	N	State of the light
Temperature controller	Lighting equipment	R/C	N	Attribute without permission

used to read and write the lighting equipment, the ruling will not pass because the attribute information does not match, so the ZKToken cannot be obtained.

Through simulation results and access logs, we verify the practical feasibility of the BZBAC model. The simulation results show that the resource device can be used when the user has access rights and the device does not have effective access to the resource at the penalty stage. If the user does not have the corresponding access right or the access right attribute does not match the device attribute, the resource device will not provide the relevant resources for the visitor to use. When the access device is in the penalty phase, the system will directly deny the user's access request and no longer issue the request to the resource. Thus, the BZBAC model can realize the access control function of the IoT.

6 Results and discussion

Through the generation and verification of zero-knowledge token, the Internet of Things access is anonymized and the access efficiency is improved. Experiments and analysis show that the model is suitable for access control in the Internet of Things environment.

In today's Internet of Things environment, attacks against devices of the Internet of Things are ubiquitous, and the security of the Internet of Things has always been a problem that needs to be solved. From the perspective of fine-grained access control and anonymous access based on attributes, this paper designs an access model, which provides ideas and methods to solve the problems of unauthorized access and identity exposure. However, this model still has some limitations. For example, when using some Internet of Things equipment resources that are not commonly used, the time cost of a single visit will be relatively high because it takes a long time to set the trust of zero-knowledge proof.

7 Conclusion

Aiming at the current access security problems of the IoT, this paper designed and implemented a BZBAC model based on blockchain technology, which is a secure access control model of the IoT using zero-knowledge proof and smart contract technology. By using a smart contract to distribute the attribute information and access policy, the single point of failure and credibility in the traditional access model are solved; by using zero-knowledge proof technology to realize anonymous access, the problem that sensitive information cannot be stored in a blockchain is solved; finally, by carrying on the simulation experiments, the safety of the model is verified, tested the feasibility of the model in the Internet environment, performance results show that the model can meet the security requirement of the Internet environment and security.

Abbreviations

IoT: Internet of Things; ZKP: Zero-knowledge proof; ABAC: Attribute-Based Access Control; zk-SNARKs: Zero-knowledge succinct non-interactive arguments of knowledge; BZBAC: Blockchain and Zero-knowledge Token-Based Access Control; ZKToken: Zero-knowledge token.

Acknowledgements

The authors gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Authors' contributions

ML studies the blockchain and attribution-based access control model and builds the access framework. ZZ has improved the access framework, fine-grained access attributes, and improved access efficiency by introducing Token. XJ further improves the model, realizes the anonymization of access, and carries out simulation test on the model. LS conceived the study and was involved in its design and coordination. All authors read and approve the final manuscript.

Funding

This work was supported by National Key R&D Program of China (2018YFB1800302),Beijing Natural Science Foundation (KZ201810009011, 4202020, 19L2021), Science and Technology Innovation Project of North China University of Technology, Research on Key Technologies for Secure Access and Reliable Access of Industrial IoT Nodes (No.18XN053).

Availability of data and materials

Not applicable: Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Competing interests

The authors declare that they have no competing interests.

Received: 16 December 2020 Accepted: 13 April 2021
Published online: 26 April 2021

References

1. A. Ouaddah, H. Mousannif, A.A. Elkalam, A.A. Ouahman, Access control in the internet of things: big challenges and new opportunities. Comput. Netw. **112**, 237–262 (2017)
2. Y. Zhang, L. Yu, L. Zhen, Z. Liu, D. Gu, *Z-Channel: Scalable and Efficient Scheme in Zerocash* (Springer, Cham, 2018)
3. R. Henry, A. Herzberg, A. Kate, Blockchain access privacy: challenges and directions. IEEE Secur. Priv. (2018)
4. D.E. Kouicem, A. Bouabdallah, H. Lakhlef, Internet of things security: a top-down survey. Comput. Netw. **141**, (2018)
5. A. Oua Dd Ah, A.A. Elkalam, A.A. Ouahman, *Towards a Novel Privacy-Preserving Access Control Model Based on Blockchain Technology in IoT* (Springer, 2017)
6. Y. Mei, S.O. Computer, Simplification model construction of internet access control based on block chain. J. Commun. Univ. China (2017)
7. G. Wang, Z.J. Shi, M. Nixon, S. Han, Chainsplitter: towards blockchain-based industrial iot architecture for supporting hierarchical storage. IEEE (2019)
8. R. Xu, Y. Chen, E. Blasch, G. Chen, BlendCAC: A blockchain-enabled decentralized capability-based access control for IoTs. (2018)
9. S. Zhu, Z. Cai, H. Hu, Y. Li, W. Li, zkCrowd: A Hybrid Blockchain-Based Crowdsourcing Platform. IEEE Trans. Ind. Inform. (2020)
10. Y. Zhang, K. Shoji, Y. Shen, X. Jiang, J. Wan, Smart contract-based access control for the internet of things. IEEE Internet Things J. (2018)

11. X.L. Fan, C.X. Fan, W.U. Yue-Xin, Realization of privacy protection of food supply chain based on blockchain and IPFS. J. Appl. Sci. (2019)
12. Y. Yuan, F.Y. Wang, Towards blockchain-based intelligent transportation systems. in *IEEE International Conference on Intelligent Transportation Systems* (IEEE, 2016)
13. O. Pinno, A. Gregio, L. Bona, ControlChain: Blockchain as a central enabler for access control authorizations in the IoT. in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference* (IEEE, 2018)
14. H. Shi, S. Wang, Q. Hu, X. Cheng, J. Yu, Fee-free pooled mining for countering pool-hopping attack in blockchain. in *IEEE Transactions on Dependable and Secure Computing* (2020)
15. S. Wang, C. Wang, Q. Hu, Corking by forking: Vulnerability analysis of blockchain. in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications* (IEEE, 2019), pp. 829–837
16. Q. Hu, S. Wang, X. Cheng, A game theoretic analysis on block withholding attacks using the zero-determinant strategy. in *Proceedings of the International Symposium on Quality of Service*. (2019), pp 1–10
17. L. Zhou, L. Wang, T. Ai, Y. Sun, BeeKeeper 2.0: confidential blockchain-enabled IoT system with fully homomorphic computation. Sensors **18**(11), 3785 (2018)
18. T. Hardjono, A. Pentland, Verifiable anonymous identities and access control in permissioned blockchains. (2019)
19. R. Henry, A. Herzberg, A. Kate, Blockchain access privacy: Challenges and directions. IEEE Secur. Priv. **16**(4), 38–45 (2018)
20. Z. Cai, X. Zheng, A private and efficient mechanism for data uploading in smart cyber-physical systems. IEEE Trans. Netw. Sci. Eng. **7**(2), 766–775 (2018)
21. A.P. Khandavilli, M. Rahman, S. Sampalli, A mobile role-based access control system using identity-based encryption with zero knowledge proof. in *2012 IEEE symposium on computational intelligence for security and defence applications* (IEEE, 2012), pp 1–7
22. X. Yang, W. Li, A zero-knowledge-proof-based digital identity management scheme in blockchain. Comput. Secur. **99**, 102050 (2020)
23. Y. Liu, G. Zhou, Key technologies and applications of internet of things. in *2012 Fifth International Conference on Intelligent Computation Technology and Automation* (IEEE, 2012), pp. 197–200
24. K. Christidis, M. Devetsikiotis, Blockchains and smart contracts for the internet of things. IEEE Access **4**, 2292–2303
25. J. Groth, On the size of pairing-based non-interactive arguments. in *Annual international conference on the theory and applications of cryptographic techniques* (Springer, Berlin, Heidelberg, 2016), pp. 305–326
26. M. Maller, S. Bowe, M. Kohlweiss, S. Meiklejohn, Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings. in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (2019), (pp. 2111–2128)
27. A. Chiesa, Y. Hu, M. Maller, P. Mishra, N. Vesely, N. Ward, Marlin: Preprocessing zksnarks with universal and updatable srs. in *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (Springer, Cham, 2020), pp. 738–768
28. J. Eberhardt, S. Tai, Zokrates-scalable privacy-preserving off-chain computations. in *2018 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)* (IEEE, 2018), pp. 1084–1091
29. Y. Hao, Y. Li, X. Dong, L. Fang, P. Chen, Performance analysis of consensus algorithm in private blockchain. in *2018 IEEE Intelligent Vehicles Symposium (IV)* (IEEE, 2018), pp. 280–285

Publisher's Note