Dynamic Trust-Based Resource Allocation Mechanism for Secure Edge Computing

Huiqun Yu[1,2(✉)], Qifeng Tang[1], Zhiqing Shao[1(✉)], Yiming Yue[1], Guisheng Fan[1(✉)], and Liqiong Chen[3]

[1] Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai, China
{yhq,zshao,gsfan}@ecust.edu.cn
[2] Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai, China
[3] Shanghai Institute of Technology, Shanghai 201418, China

Abstract. Edge computing plays an important role in processing and storing data. By offloading tasks to the edge server, mobile users can access necessary computing resources on demand. However, security of edge computing service is still a major concern. This paper proposes an edge computing resource allocation mechanism based on dynamic trust. First, security problems due to lack of reliability in the resource allocation process are solved based on the trust mechanism. This mechanism considers the resource allocation process between the mobile user node and the edge server as a transaction, according to the trading behavior in the transaction process of server to give its corresponding trust. Second, a trust mechanism is used for dynamic credit granting. Mobile users with similar behaviors form a group, where a representative is elected to trade resources and bundle information into a block and attach it on the chain. At the same time, the delay problem is added as a constraint to the trust calculation. Finally, the simulation experiment shows that the mechanism improves security of edge computing.

Keywords: Edge computing · Security · Trust · Blockchain · Resource allocation

1 Introduction

Edge computing technology has emerged with the development of innovative edge devices, such as the Internet of Things and smart phones. In order to

This work was partially supported by National Natural Science Foundation of China No.62276097, Shanghai Municipal Natural Science Foundation under Grant No.21ZR1416300, Capacity building project of local universities Science and Technology Commission of Shanghai Municipality No.22010504100.

H. Gao et al. (Eds.): CollaborateCom 2022, LNICST 461, pp. 111–132, 2022.
https://doi.org/10.1007/978-3-031-24386-8_7

improve the efficiency of computing resources and optimize performance indicators, edge computing resource allocation and task scheduling have received widespread attention [1]. At the same time, edge computing also faces many security problems, its nodes are exposed at the edge of the network [2], computing power and storage capacity are limited, which make equipment and network resources favor by attackers easily [3].

The attackers pose threats to different infrastructures in edge computing network architectures, such as user devices, server nodes, and network resources. In order to improve the security of edge computing in resource allocation, a reasonable trust mechanism can be established to filter the infrastructure in the network. To construct the evaluation mechanism including resource trust, identity trust and behavior trust, it is necessary to integrate the historical information, the matching degree of resources to different requirements.

For the design of trust mechanism, the credibility mechanism included in blockchain technology has been relatively mature in the calculation of trust, so it can be introduced to edge computing. The blockchain technology adopts the method of distributed data storage, in essence, it can be seen as a decentralized database. We can think of blockchain as an intermediary responsible for resource allocation transactions, account management and currency exchange, which is jointly managed and maintained by users. Therefore, compared with the traditional centralized database, the consensus mechanism, encryption algorithm, smart contract and other technologies included in it make it have the characteristics of multi-party maintenance, immutability, openness and transparency [4], and data security and high availability are well guaranteed [5]. Therefore, with the help of its immutable property, it can be considered to store the relevant information in the transaction process of edge computing resources on the chain, which can be monitored and viewed by users, and is not easy to be changed.

This paper studies the security of edge computing resource allocation and proposes a dynamic trust-based edge computing resource allocation mechanism (DTERAM). This mechanism regards the resource application process between the edge server and the mobile user as a transaction, and dynamically grants credit based on the behavior of the edge server in the resource transaction process. At the same time, in order to reduce the number of interactions between the edge server and mobile users and the cost of mobile users to purchase resources, the DTERAM divides mobile users into groups and realizes resource sharing within the group. Mobile users apply for resources to edge servers on a group basis, preferentially select edge servers with a high degree of trust for transactions, and group members share the cost of purchasing resources. The DTERAM takes the security of resource transactions as the evaluation standard, realizes resource allocation between edge servers and mobile users, and improves the security of the transaction process.

The main contributions of this paper are as follows:

(1) A trust model is established to realize the resource transaction process between edge servers and mobile users, which improves the security of resource allocation. The server pricing process takes into account the relationship between

price and user needs, and uses a greedy algorithm to solve the trust model to improve the security in the process of resource transactions.

(2) An edge computing resource allocation mechanism based on dynamic trust of blockchain is proposed. Two aspects of historical trust degree and dynamic trust degree are considered. Adding trust in the evaluation of the server status is more conducive for mobile users to select edge servers with high security.

The remainder of this paper is organized as follows. In Sect. 2, related work is reviewed. Section 3 introduces the system model. Section 4 introduces the DTERAM resource allocation mechanism. Section 5 carries on the experimental results and related analysis and the conclusion is given in Sect. 6.

2 Related Work

In recent years, there has been a lot of research work on resource allocation in edge computing. Dong et al. [6] presented a task priority-oriented resource allocation method for mobile edge computing, and assigned corresponding priorities to tasks based on their average processing value to achieve the effect of reducing overall delay and energy consumption. Li et al. [7] proposed a joint resource allocation and task scheduling algorithm, which improved the peak load capacity of the edge and reduced user delay. Xue et al. [8] established a joint convex optimization goal based on computational offloading and task allocation, and used Lagrangian multiplier method to iterate update to get the optimal solution. Yang et al. [9] proposed a joint optimization scheme for task offloading and resource allocation in a 5G communication network based on edge computing, and transformed the problem of task offloading and resource allocation into a joint optimization problem of time delay and energy consumption. Alfakih T et al. [10] proposed a state-action-state-action (RL-SARSA) algorithm based on reinforcement learning to solve the resource management problem of edge servers. Liao et al. [11] proposed a resource allocation and task scheduling optimization scheme based on service emergency priority. Samrat Nath et al. [12] studied the dynamic caching, computing shunting, and resource allocation problems in the cache-assisted multiuser MEC system with random task arrival. Wang et al. [13] studied the problem of effectively allocating and adjusted edge resources in the case of high dynamics brought about by user mobility in edge computing.

The main focus of the above-mentioned research is on the algorithm optimization of the edge computing resource allocation process, which is continuously improved under the premise of considering the characteristics of delay and mobility, but the security issues are ignored. However, the Internet technology is becoming more and more perfect, security issues such as data leakage and personal information privacy appear to be particularly important. Therefore, improving the security in the process of resource allocation has become an urgent problem to be solved in related fields.

As the underlying technology of the Bitcoin system, blockchain technology has been more and more used in recent years due to its high security. With the development of blockchain technology research, there are more researches on the application of blockchain in the field of non-digital currency [14], such as in

applying it to edge computing to solve security problems. Ref. [15] proposed an edge computing distributed trusted authentication system based on blockchain technology. Xu et al. [16] aimed at the problem of lack of trust in sharing the data generated in edge computing among stakeholders, developed a blockchain-based big data sharing framework, and a new type of blockchain transaction including Express. Zhang et al. [17] aimed at the security problem of the consensus algorithm vulnerable to attacks in blockchain-based mobile edge computing, proposed a group signature scheme to verify the generated blocks of the blockchain and verify the identity of mobile users. Wu et al. [18] introduced an incentive mechanism and a decentralized accountability mechanism to establish a trust and reputation system for CEC stakeholders, and used smart contracts to verify correctness and automatically punish them in case of failure. Nabil EI Loini et al. [19] established a trusted orchestration management framework based on blockchain, which supports the identification, traceability and orchestration of all participants, and achieves complete tracking and verification of data. Huang et al. [20] used blockchain technology to improve the security of edge computing resource allocation, while taking into account the fairness cost (FDC) and node mobility (RDC). In the reputation based consensus mechanism (PoR) included in the D2D-ECN framework proposed in Ref. [21], the device with the highest reputation score is responsible for packaging the resource transactions and reputation records of the blockchain.

For the existing research on improving the security of edge computing with the help of blockchain technology, the trust degree is mainly based on identity verification, data storage and verification, but the behavior trust of participants is only an evaluation value based on historical information. Therefore, we consider a dynamic trust evaluation of participants behavior, and evaluate trust from two aspects: historical information and real-time transaction behavior.

3 System Model

First, an example of edge computing resource trading is given to describe the process of resource trading. The process consists of four parts: pricing, bidding, selection and negotiation, transaction and feedback. When a transaction occurs, the sequence of steps performed is as follows:

(1) The edge server sets the price of a unit resource with reference to the overall demand put forward by the mobile user, and the unit resource price is inversely proportional to the overall demand of the mobile user.

(2) After negotiation within the group, mobile users give their own bids based on their actual conditions.

(3) Mobile users select the target server based on the trust level, the resource pricing and the resource capacity of the edge server, then negotiate the final price with the target server, and the final price must be higher than the cost price of the resource. The negotiation process is divided into three types: 1) If the mobile user bid is not less than the selling price of the edge server, the final price is the mobile user bid; 2)If the mobile user bid is less than the selling price of the edge server and greater than the resource cost price, the final price is the

selling price; 3)If the mobile user's bid is less than the cost price of the resource, then the transaction failed.

(4) The mobile user group and the edge server conduct transactions at the final price determined in step 3. After the transaction is completed, the mobile user will give feedback on the quality of experience during the transaction.

The specific process is shown in Fig. 1. The mobile users are divided into different groups. The members in the group have the same preference for resources and the group is used as a unit to apply for resources. In order to improve the security in the resource allocation process, the mobile users select edge servers for transactions based on factors such as trust and price. The resource transactions involve a group of edge servers (sellers) and mobile user groups (buyers). The edge server sets its own selling price according to the overall needs of users. The user makes a choice with reference to the trust and selling price of the edge server. During the transaction, mobile users do not know each other's bids, and edge servers do not know each other's selling prices, and the information is stored on the blockchain. According to resource demand, trust and price, complete the mapping of edge server and mobile user group to realize resource service.

As shown in Fig. 1, we model the process of resource allocation between edge servers and mobile users as trust transactions, and design a high security feasible solution for j mobile user groups to allocate i edge server resources. The solution considers that the resources provided by different servers are heterogeneous, because the same type of resources provided by different servers are different due to factors such as trustworthiness, service quality, and price.

In the transaction process between the edge server and the mobile user, in order to obtain the maximum utility, the edge server acts as the seller and sets the resource selling price according to the resource cost and demand. As a buyer, in order to reduce costs and the number of interactions with edge servers, the mobile users form a group with the same hobbies, share the resources and distribute the cost evenly. Additionally, the representative selected by group conduct transactions with the edge server. In the selection process of the edge

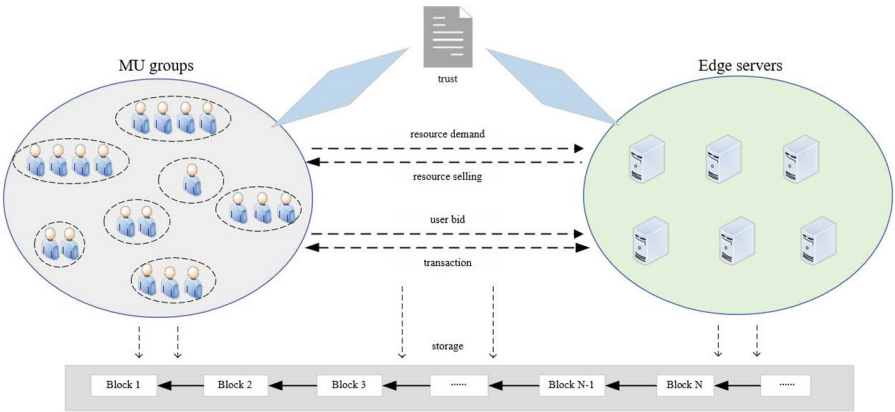

Fig. 1. Trading model of resource allocation mechanism based on dynamic trust

Table 1. Parameters of the system model

Symbol	Description
I	The set of edge servers
J	The set of mobile user groups
C_i	Capacity of edge server i
B_i	Bandwidth of edge server i
p_i	Unit resource final price of edge server i
s_i	Unit resource selling price of edge server i
c_i	Unit resource cost price of edge server i
val_j	Unit resource bid of mobile user group j
D_j	Total resource demand of mobile user group j
$m_{i,j}$	Resources demand by mobile user group j from edge server i
$trust_i$	Trust of edge server i
$\Delta_{i,j}$	State evaluation by mobile user group j for edge server i

server, although each edge server has its own resources, due to the difference in service quality, the user group will first evaluate the status value of the server through the trust level, resource capacity and resource price of the edge server, then select the server with the best status value for resource service. The main symbols involved in the transaction model are explained in Table 1.

Edge Servers: Edge server provides mobile users with the resources they request, the set of edge servers is denoted by $I = \{1, 2, \ldots, i, \ldots, m\}$, the capacity of edge server i is denoted by C_i and the bandwidth of edge server i is denoted by B_i. Different edge servers have different quality of service when providing resources to users. The trust degree of the edge servers is evaluated, the initial trust degree is set to 0.5, and the upper limit is set to 1. The edge servers are divided into three categories through trust changes: high-quality, low-quality, and malicious edge servers. Edge servers with a degree of trust between [0.5,1] provide high-quality services, with a large number of successful transactions, reasonable resource prices, and low transaction delays; Edge servers with a degree of trust between [0.2, 0.5) provide low-quality services, the number of successful resource transactions is moderate, resource prices are high, and the transaction process delay is relatively high; Those with a trust level of [0, 0.2) are malicious edge servers. When users apply for resources, they conduct malicious competition through measures such as price reduction, or tamper with the content of resources, which lead to a higher number of transaction failures.

Mobile User Groups: Mobile users with similar interests form a group, the set of mobile user groups is denoted by $J = \{1, 2, \ldots, j, \ldots, n\}$, the number of mobile users in mobile user group j at time t is denoted by $n_j(t)$. Most of the resources required by the members of the group are the same, so a representative from a user group can be selected to apply for resources from the edge server and conduct resource transactions. The total resource demand of mobile user group j is denoted by D_j. The resources obtained after the transaction is completed can be shared and exchanged within the group, which can improve their QoE. In addition, the members of the group equally share the costs required in

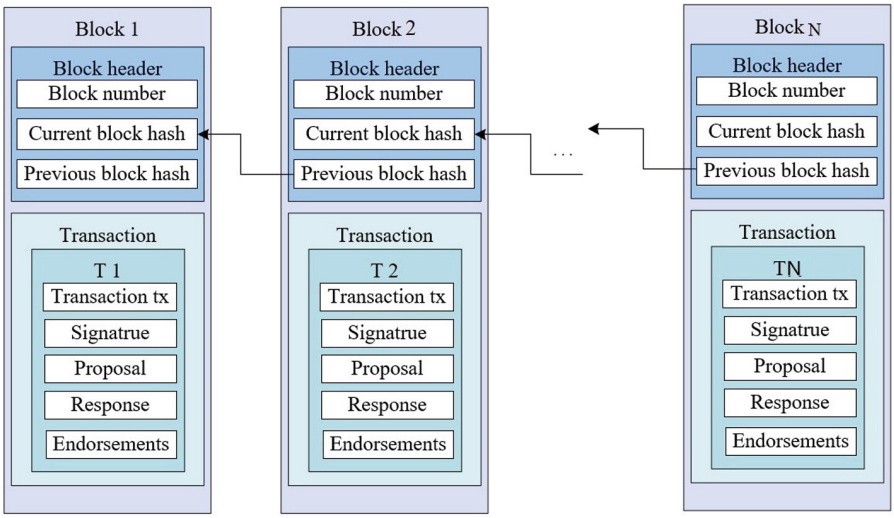

Fig. 2. Blockchain structure

the resource transaction process, which not only reduces the number of interactions between the user and the server, avoids repeated applications for the same resource, but also reduces the cost for users to obtain the required resources.

Blockchain: Blockchain can be divided into public chain and private chain. Public affairs can be verified by all independent participants, and private affairs need to be processed by authorized participants. In this paper, a public chain is used to record the resource transaction process between edge servers and mobile users. The block structure is shown in Fig. 2. Each block contains two parts: Block header and Transaction. The Block header realizes the connection between blocks through the included hash value, and Transaction is responsible for storing the relevant information of each transaction. Members of the same mobile user group can view and verify the information on the blockchain. Representatives selected by each group are responsible for packaging the relevant transaction records of each resource application and uploading them to the blockchain.

Smart Contract: The smart contract is a set of commitments defined in digital form, and an agreement that includes contract participants to implement these commitments. Smart contracts can be introduced in the transaction process, and information such as pricing, payment, storage, and delivery can be processed through smart contracts. As shown in Fig. 3, each smart contract is assigned a unique address, which can be triggered by sending a transaction. Different events are triggered by processes in the smart contract, and related transactions will be recorded on the blockchain in the order of timestamps. The use of smart contracts can enable entities to write transaction rules according to certain specifications of their own conditions, and achieve the purpose of maximizing utility through time and transaction prices. In addition, using smart contracts to execute transactions does not need to rely on trusted third parties (banks, Government, etc.).

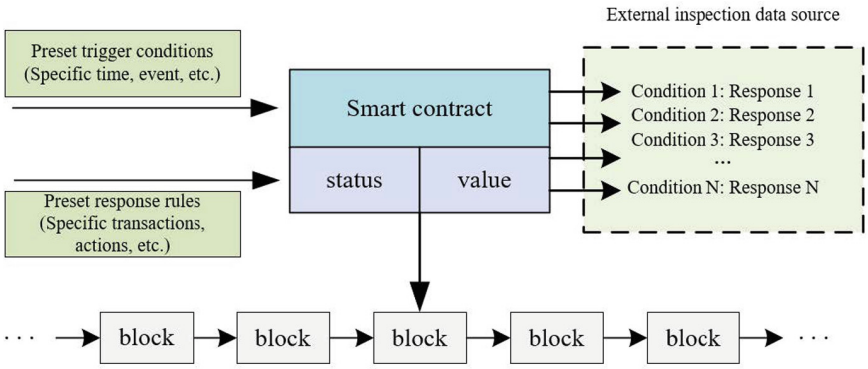

Fig. 3. Smart contract

The specific conditions and process of the smart contract are as follows:

Initialization: 1) Initial settings for resource transactions. The capacity C_i, cost price c_i, selling price p_i and trust $trust_i$ of edge server i; The resource demand D_j, resource bid val_j of mobile user group j; Transaction deployment time $dTime$. 2)The edge server i and the mobile user group j negotiate smart contract transaction rules.

Creation: After the edge server i and the mobile user group j agree on the smart contract transaction rules, use the create function to deploy smart contracts on the blockchain. The output of this function is the address of the smart contract on the blockchain, which is public to all edge servers and mobile users, so each entity can be selective Interact with the contract. In addition, in order to ensure the smooth progress of the smart contract, both the edge server i and the mobile user group j must put some deposits in their accounts into the smart contract to prevent malicious behavior. The smart contract will return the deposit after the transaction is over.

Transaction: If a smart contract is deployed on the blockchain, the transaction function is executed, resource transactions between edge server i and the mobile user group j will start after time $t > dTime$. The mobile user group j calculates status $\Delta_{i,j}$ based on the trust $trust_i$, the resource capacity C_i, and the unit resource selling price p_i of the edge server i, choose the server with the largest value of $\Delta_{i,j}$ to apply for resource transactions. The edge server i determines the resource selling price according to the total resource demand of mobile users in order to obtain the maximum utility u_i. In addition, smart contracts can supervise content delivery between mobile user group j and edge server i. If any party does not abide by the signed agreement, the function $Penally()$ will be called. Finally, if the smart contract reaches the service period, financial settlement is performed, and all assets owned by it are recovered.

Threat Model: We also consider the harm of untrusted edge servers and mobile user groups. First, the edge server maliciously participates in the competition of

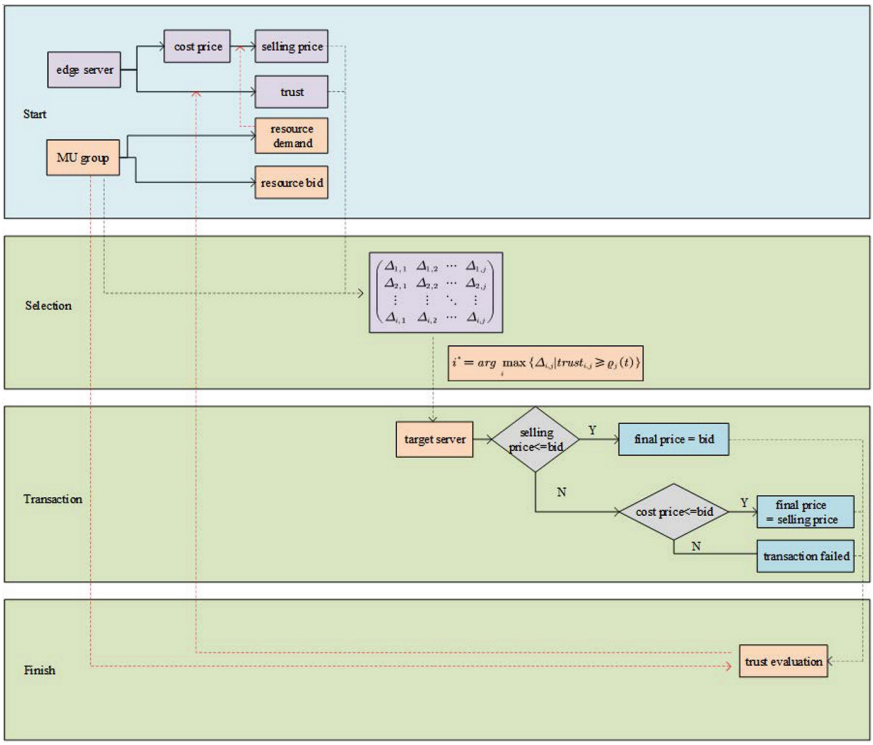

Fig. 4. Transaction process

resource transactions, such as malicious bidding that interferes with the transaction, resulting in a waste of time or resources. Second, attackers may use them to return malware or viruses to the requesting program to gain potential intent. Third, a malicious mobile user group may refuse the resource service of the edge server and thus refuse to pay. Similar to the existing blockchain-based applications, this paper uses a reputation mechanism to grant dynamic trust, and uses trust as an important reference condition in the resource transaction process to ensure security.

4 Dynamic Trust-Based Edge Computing Resource Allocation Mechanism

The specific process of the dynamic trust-based edge computing resource allocation mechanism is shown in Fig. 4. There are four stages including Start, Selection, Transaction, and Finish. Among them, the Start stage is mainly the pricing stage of the edge server, the pricing process is based on the total demand of mobile users. Selection, Transaction and Finish are three stages of the transaction process. In this three processes, mobile users will calculate state values based

on the trust and the pricing of the edge server, then select the edge server with the largest state value for transaction negotiation. The negotiation is mainly for the price of resources. The negotiation process is divided into three types, which corresponding to different results. Finally, the mobile user evaluates the edge server on the transaction quality, the transaction information and evaluation results will be recorded in the block.

4.1 The Resource Pricing of Edge Servers

Because the resource capacity of each edge server is limited, mobile users must pay the corresponding service fee when applying for resource services from the edge server. Therefore, after the transaction is completed, the utility u_i of the edge server i is denoted by:

$$u_i = p_i * D_j - c_i * D_j; i \in I, j \in J \tag{1}$$

where p_i is the final price of unit resource, c_i is the cost price of unit resource and D_j is the total resource demand of mobile user group j from edge server i.

Considering the relationship between the resource demand of mobile users and the resource pricing of the edge server, when the edge server sets the resource price, it needs to know the total demand for the resource of the mobile user. When the price is high, the demand will become lower. Conversely, when the price is low, the demand will become higher, that is, the user's demand and the price are inversely proportional. So in order to describe the relationship between price and demand, we use a linear function to describe it.

$$D_j = \begin{cases} C_i - \xi_i * s_i, s_i \leq \frac{C_i}{\xi_i} \\ 0, s_i > \frac{C_i}{\xi_i} \end{cases} \tag{2}$$

where C_i is the resource capacity of the edge server i, s_i is the selling price of unit resource and ξ_i is the price reference value of edge server i when pricing. The ξ_i is related to many factors, including the number of mobile users, the size of the resource applied for, and the popularity of the resource. Therefore, the calculation method of the ξ_i in reference [22] takes into account the relationship between the price of edge server resources and the demand of mobile users, and rewrites the utility of the edge server i as

$$u_i = (p_i - c_i) * (C_i - \xi_i * s_i); \forall i \in I \tag{3}$$

In this process, the malicious edge server can have two kinds of attack behaviors. The first type of behavior is that a malicious edge server deletes, modifies, or replaces the resource content applied by the mobile user to achieve some of its potential intentions. The second type of behavior is that the edge server has been destroyed, thereby injecting viruses or malware into mobile users requesting resources from themselves. If an attack is to be implemented here, mobile users need to select a malicious edge server to apply for resources. In the solution proposed in this paper, the choice of edge server is related to trust and

resource prices. Therefore, in order to attract more mobile users, each malicious edge server needs to obtain a higher degree of trust, and need to set a lower resource price, but this will not maximize the utility. At the same time, after a transaction is over, the feedback of mobile users will reduce the trust value of the malicious edge server, making it unable to have a higher trust value, then it will not continue to be selected. Therefore, our proposed scheme can avoid these two attack methods.

4.2 Mobile User Groups

In the process of resource transactions, the attack of malicious edge servers may cause mobile users to be unable to obtain the required resources for security. Therefore, in order to improve the security in the transaction process, we have added the concept of trust to enable mobile users to obtain reliable and trustworthy resource services. We assign a trust value to each edge server, and use the trust value to indicate the credibility of the edge server. The higher the value, the higher the credibility of the server and the safer and more reliable the resource services provided.

According to the interactive behavior and result of the resource transaction process with the edge server, the mobile user can evaluate the service quality of the edge server. If the user is satisfied with the service, they can send a high-level feedback, and the user can achieve dynamic credit to the edge server based on the real-time feedback of each service quality, update its trust value in time, and ensure high security at any time.

According to the processing method in reference [23], the DTERAM mechanism divides the entire process into a series of epoch from the running time, each epoch completes a resource transaction and generates a block, which is divided into three parts in the process of calculating the trust degree of the edge server.

(1) The initial trust $trust_i(his)$ based on the historical records before the start of each epoch;

(2) The trust $trust_i(t-1)$ of the last transaction at the current transaction moment t;

(3) The delay time *Latency* obtained by the calculation method of delay in reference [24].

First, based on the logistic regression model, the calculation method of the initial trust degree of the edge server is given.

$$trust_i(his) = \frac{1}{1 + e^{-\alpha(\sum_{x=0}^{n-1} \nu_x - \gamma \times \sum_{x=0}^{n-1} \varphi_x)}} \tag{4}$$

where $trust_i(his)$ is the initial trust given to the edge server i based on the previous behavior of i at the beginning of the current transaction, n is the current nth transaction, α is the total number of transactions that the server

has participated in, ν_x is whether edge server i is trading normally during the xth transaction, normally is 1 and otherwise is 0. And φ_x is whether edge server i is trading maliciously during the xth transaction, maliciously is 1 and otherwise is 0, γ is the penalty weight for malicious transactions performed by the edge server, which can be set by the user. The greater the weight, the greater the penalty for malicious transactions. At the same time, the initial trust level is specified as $trust_i(0) = \frac{1}{1+e^{-\alpha(0-0)}}$=0.5.

The logistic regression model has a rapid increase in the trust value during the logarithmic growth period. It is not reasonable to judge the trust value purely based on the model. Therefore, this paper balances the trust of current transaction based on the historical and the trust of the last transaction. At the same time, considering the delay of edge computing, the formula for calculating the trust of the transaction is finally obtained.

$$trust_i(t) = \beta \times trust_i(his) + (1-\beta)trust_i(t-1) + \frac{\lambda}{Latancy} \tag{5}$$

where $trust_i(t)$ is the trust of the edge server i in the tth epoch resource transaction. Here, parameter β issued to modify the rate of increase of trust to avoid the centralization of trust in the initial stage caused by excessive growth. The initial value of β is 1, because at the beginning it is not known whether the edge server will be prone to malicious transactions. Parameter λ is to weight the delay, the delay and the trust have an inverse relationship. The smaller the delay, the greater the trust degree value. Conversely, the greater the delay, the lower the trust degree value.

The change of parameter β is determined by the cumulative trust deviation $\nu_t * ttrust_i$, and the specific relationship is

$$\beta = threshold + c \times \frac{\delta_t * trust_i}{1 + \nu_t * trust_i} \tag{6}$$

Initially, $\nu_0 * trust_i = 0$, parameter c can be defined by the user to control the weight of the reaction to the recent behavior of the edge server. *threshold* is a threshold set to prevent β transition saturation from tending to 1, the initial value is set to 0.25, $\delta_t * trust_i$ is the trust degree deviation, the calculation method is,

$$\delta_t * trust_i = |trust_i(t-1) - trust_i(his)| \tag{7}$$

At the tth epoch transaction, the trust degree deviation of edge server i is equal to the difference between the current initial trust degree and the absolute value of the trust degree in the $t-1$th epoch transaction, therefore, the calculation method of the cumulative trust deviation $\nu_t * trust_i$ in the tth epoch transaction is

$$\nu_t * trust_i = c \times \delta_t * trust_i + (1-c)\nu_{t-1} trust_i \tag{8}$$

The larger the value of the parameter c, it means that the weight of the recent trust deviation given by the mobile user is more important than the previous cumulative trust deviation weight.

Latency is the delay time, which is inversely related to the trust. Here the delay time is divided into four parts, namely the bidding time of mobile user group bid^j, the bidding time of edge server $charge^i$ and the time of negotiate $Cond_i^j$.

$$Latency = \{bid^j, charge^i, Cond_i^j\} \tag{9}$$

After obtaining the trust of the edge server, the mobile user group will choose the edge server with the best trust according to their needs. There are two criteria for mobile users to choose the best edge server: 1)The optimal edge server selected should have a high degree of trust and be able to provide safe, reliable, and high-quality resource services; 2)The resource price of the optimal edge server should be low and the capacity should be large. Therefore, each mobile user group will establish a trust threshold to judge whether the edge server is trustworthy. The trust threshold is calculated as:

$$\nu_{i,j}(t) = \omega_{tr} trust_{i,j}^{max}(t) + \alpha \times log(1 + \frac{n_j(t)}{n_j^{max}(t)} \tag{10}$$

$trust_{i,j}^{max}(t)$ is the maximum trust of the mobile user group j to the edge server i from the initial time to the current time t, $n_j(t)$ represents the number of users in mobile user group j at time t, $n_j^{max}(t)$ represents the maximum number of users in the mobile user group in time[0,t], α is a weighting parameter, and ω_{tr} is a threshold adjustment parameter.

Each mobile user group calculates the resource status of the edge server, and then selects an optimal server for resource transactions. We define the resource status of the server as the ratio between trust level, resource capacity and resource price. Then for the mobile user group j, the resource status of the edge server i is

$$\Delta_{i,j} = \frac{\eta * trusti, j + \mu * C_i}{s_i} \tag{11}$$

where η and μ are the weighted parameters of trust level and resource capacity respectively, according to the resource status of each edge server, the mobile user group j selects the best edge server for transactions.

$$i^* = argmax_i\{\Delta_{i,j} | trust_{i,j} \geq \zeta(t)\} \tag{12}$$

After the mobile user group j selects the edge server i^* corresponding to the maximum state value $\Delta_{i,j}$, the two will negotiate the resource price of the transaction. The negotiation process is:

$$p_i = \begin{cases} val_j, val_j > s_j \\ s_i, c_i \leq val_j \leq s_i \\ fail; val_j < c_i \end{cases} \tag{13}$$

On the other hand, trust can be used as a reward for edge servers to provide high-quality services, and it is also a manifestation of edge server reputation. In order to maintain the number of edge servers participating in the resource transaction process, trust consumption is introduced. In addition, based on the characteristics of the Logistics regression model, the trust level of the edge server is limited by the upper limit. Here, the reference to trust consumption is to ensure the participation of edge servers. If few edge servers participate in the transaction process, the resource allocation mechanism based on dynamic trust is of little significance. As long as the edge server participates in the bidding and selection of resource transactions, regardless of whether it is selected by the mobile user in the end, there will be no trust consumption. On the contrary, if the edge server does not participate, then its trust will be consumed. The calculation formula for the trust consumption of the edge server i is

$$trust_i(his) = \begin{cases} \frac{1}{1+e^{-\alpha(\sum_{x=0}^{n-1} \nu_x - \gamma \times \sum_{x=0}^{n-1} \varphi_x)}}, \Delta_B = 0 \\ trust_i(t) \times e^{-D \times \Delta_B}, otherwise \end{cases} \tag{14}$$

where Δ_B represents the block interval, that is the interval between the last participating transaction and the current participating transaction (starting from 0), the calculation method is $\Delta_B = B_{cur}$ - B_{pre}. If two transactions are consecutive, then $\Delta_B = 0$, at this time, the edge server participates in the calculation and transactions with the current trust level, and the trust consumption function will not be executed, which greatly ensures that the edge server actively participates in resource transactions. The value of D will be dynamically adjusted according to the transaction quality, and the final resource transaction quality will be maintained at a stable level. The increase in transaction difficulty will make transactions require more trust weighting. High-quality servers will choose not to participate in the transaction temporarily, in order to find that the difficulty is reduced, and the opportunities will increase to participate in the transaction, but when the participation of the edge server is too low, the probability of the malicious edge server's success becomes higher. At the same time, the increase in difficulty will increase the trust consumption of edge servers that do not participate in transactions, which will help increase the participation of edge servers.

4.3 DTERAM Algorithm Implementation

The Algorithm 1 is the implementation process of the proposed DTERAM, DTERAM is mainly composed of two main parts, the user's choice of edge server (SelectEdgeServer) and the utility calculation of the edge server (Edgeserversutility). The DTERAM algorithm takes edge servers I, resource cost price c_i, resource selling price s_i, initial trust $trust_i(his)$, delay time T_i, mobile user group J, resource demand D_j and resource bid val_j as input. In each round of transactions, the mobile user group j will read the trust level of the edge server i^*, then calculate the service status $\delta_{i,j}$ of i, next sort the edge server status values in descending order, and select the server i^* with the highest status value, and then conduct price negotiation to get the transaction price p_i^*. After completing the transaction with price p_i^*, the mobile user group j evaluates the server i^*, and the server i^* calculates its own utility.

The DTERAM algorithm mainly uses blockchain-related technologies to improve the security of edge computing in the process of resource allocation. However, while using blockchain technology, the process of generating blocks and put the block on the chain will consume a part of the time. Therefore, the mechanism needs to be optimized in terms of time performance. The next step can be to reduce time consumption and improve the performance.

5 Experimental Results and Analysis

In this section, we evaluate the proposed method through simulation experiments. First, we introduce the relevant settings of the simulation experiment, and then analyze the results of the experiment.

5.1 The Setup of Simulation Experiment

First, 5 mobile user groups and 10 edge servers are deployed in the network. The number of users in each mobile user group is randomly determined between [5,10], the resource demand of resources is randomly determined between [1,10] Mb, and the resource capacity of each edge server is randomly determined between [10,50] Mb. The initial trust level of each edge server is set to 0.5, and the edge servers are preliminarily divided into three types: high quality, low quality and malicious. The proportions of the three types are 0.4, 0.3 and 0.3 respectively. Other parameter configurations are: $\zeta(i,j) = 0.4$, γ=2, $\alpha = 1$, $threshold = 0.25$, $\eta = 0.3$, $\mu = 0.4$.

Algorithm 1. Dynamic Trust-Based Edge Computing Resource Allocation Mechanism

Require:

The edge servers I,the resource cost price c_i, the resource selling price s_i, the initial trust $trust_i(his)$, the delay time T_i, the mobile user group J, the resource demand D_j and the resource bid val_j.

Ensure:

The set of redundant service DWS_f and the set of active execution service for tasks in user request U_f;

Initial: $t = 0$, $trust_i(his) = 0.5$, $p_i = 0$.
for $t = 1 \longrightarrow T$
procedure SelectEdgeServer
    get $trust_i(his)$ of each edge servers by using blockchain
    calculate $trust_i(t)$ by Eq.(5)
    for $i = 1 \longrightarrow I$
        for $j = 1 \longrightarrow J$
            $\frac{\eta * trusti,j + \mu * C_i}{s_i}$
            if $trust_{i,j} \geq \eta(t)$
                $i^* \longleftarrow max\Delta_{i,j}$
            end if
        end for
    end for
    for $j = 1 \longrightarrow J$
        if $val_j > s_i^*$
            $p_i^* \longleftarrow val_j$
            Transaction
        else if $val_j < s_i^*$ and $val_j > c_i^*$
            $p_i^* \longleftarrow s_j$
            Transaction
        else if $val_j < c_i^*$
            Transaction failure
        end if
    end for
    Each social group updates its current trust for Edge Server by Eq.(4)
    return updated $trust_i(t)$
end procedure
procedure Edgeserversutility
    for $i = 1 \longrightarrow I$
        $\mu_{i,j} \longleftarrow (p_i - c_i) \times D_j$ or 0
        $\mu_i \longleftarrow \sum_{j=1}^{J} \mu_{i,j}$
    end for
end procedure
$t \longleftarrow t + 1$
end for

5.2 The Analysis of Simulation Experiment Results

Figure 5 illustrates the relationship between the edge server unit resource price and the number of transactions. It can be seen from the figure that the unit resource price of a highquality edge server increases with the increase in the number of transactions, and eventually stabilizes. The unit resource prices of low-quality edge servers and malicious edge servers both increase at the beginning, but will gradually decrease in the future and eventually stabilize. It can be understood that the initial trust of all edge servers is the same at the beginning, the mobile user group will prefer lower-priced servers when choosing, so the resource prices of low-quality and malicious edge servers that have lower-priced will increase. However, as the transaction progresses, trust is an important basis for the selection of mobile user groups, low-quality edge servers and malicious edge servers will be exposed, users gradually turn to high-quality edge servers, which will cause the resource prices of high-quality edge servers to gradually rise to achieve greater utility. When the trust tends to stabilize, the price tends to stabilize accordingly and this time the utility of the edge server reaches its maximum value. At the same time, lowquality and malicious edge servers can only participate in the competition by lowering resource prices due to the decline in trust. However, since trust is an important basis for selection, the effect is not great. Figure 6 is the relationship between the resource demand and resource price of the mobile user group when the edge servers are 10, 20, 30, and 40 respectively. It can be seen that when the number of edge servers is different, as the mobile Fig. 6. unit resource price-resource requirements. user groups demand for resources increases, the price gradually increases. At the same time, when the resource demand of mobile users is the same, the fewer the number of edge servers, the higher the resource price. It can be understood that when the demand for mobile users increases, edge servers will increase resource prices to obtain greater profits. In addition, when there are more edge servers, the edge servers will participate in the competition by reducing prices to attract more mobile users to conduct resource transactions. Therefore, the more edge servers there are, the lower the resource price will be.

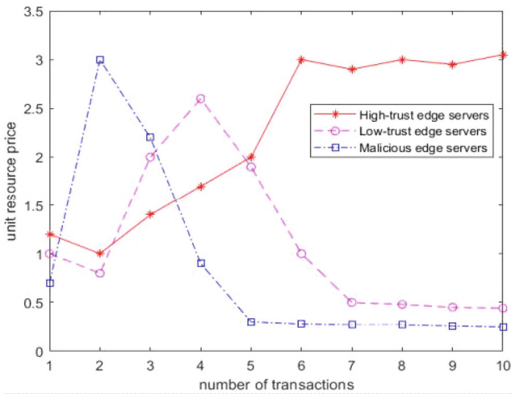

Fig. 5. Unit resource price-number of transactions

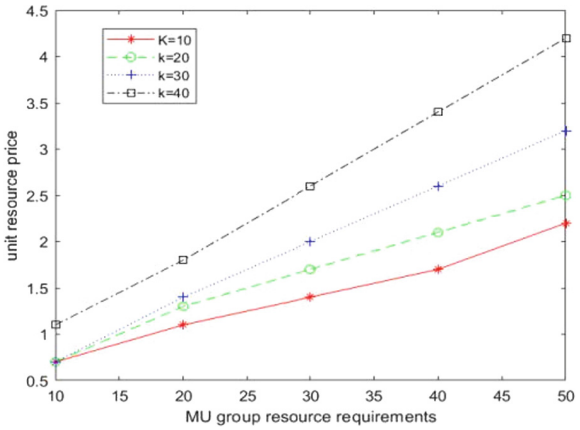

Fig. 6. Unit resource price-resource requirements

Figure 7 shows the relationship between the average trust of edge servers and the number of transactions. This paper calculates the average values of the trust levels of highquality, low-quality, and malicious edge servers respectively. The initial value of trust is 0.5, therefore, the initial values of the average trust levels of the three types of servers are all 0.5. It can be seen from the figure that the average trust of highquality edge servers increases with the increase in the number of transactions, and then gradually stabilizes. Conversely, the trust of low-quality and malicious edge servers will decrease over time. It can be understood that high-quality edge servers provide high-quality resource services. Due to high-quality services, mobile users' trust evaluation of the server during the transaction process will also increase, and as the number of transactions increases, the server's average trust level will stabilize. On the contrary, low-quality and malicious edge servers provide low-quality services that will cause mobile users to lower their trustworthiness, which leads to their average trustworthiness gradually decreasing as the number of transactions increases.

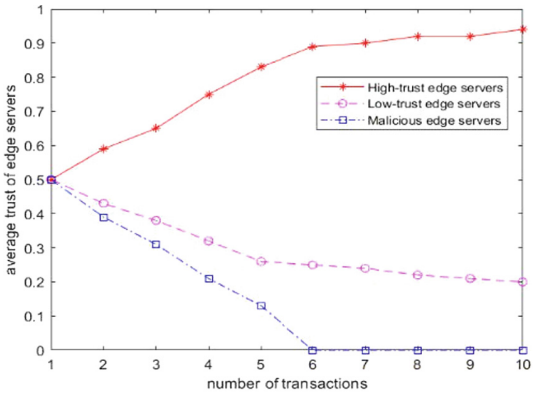

Fig. 7. The average trust-number of transactions

Figure 8 shows the relationship between the overall trust of high-quality edge servers and the number of transactions when the c are 0.1, 0.5, and 0.9 respectively. Since the high-quality ratio among the 10 edge servers is 0.4, the initial overall trust level is 2, and the upper limit of trust level is 4. It can be seen from the figure that the c value is different, the corresponding trust rate growth rate is also different, but in the end it will be close to the upper limit. The larger the value of c, it means that the trust growth rate in recent transactions accounts for a larger proportion of the overall trust growth rate, that is, the real-time trust changes reflected in the transaction process have a greater impact on the trust calculation. After the transaction, the trust of the edge server will be maintained at a stable level.

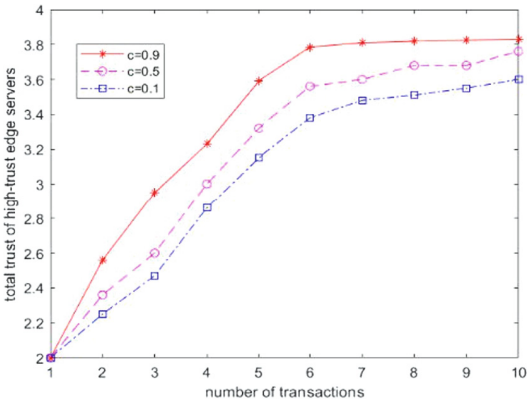

Fig. 8. Total trust-number of transactions

Figure 9 shows the decline process of the overall trust of edge servers. This process assumes that all edge servers remain offline when their trust reaches their peak, that is, if they do not participate in resource transactions, then the entire transaction will no longer be safe. It can be seen from the figure that if all edge servers do not participate in resource transactions, the trust level of the first few transactions remains basically stable, but after the fifth transaction, the trust level has dropped significantly, and the decline process is non-linear. After the seventeenth transaction, it gradually tends to zero.

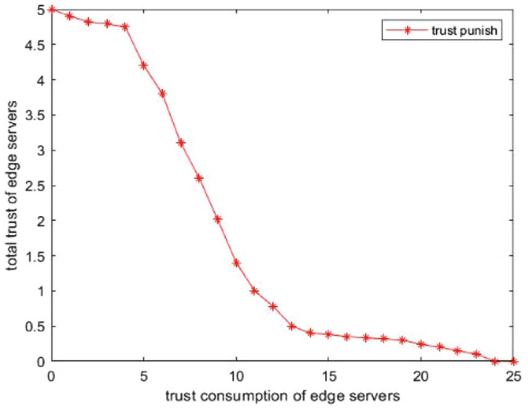

Fig. 9. Trust punish

6 Conclusion

In view of the malicious competition and vulnerability of edge computing in the process of resource allocation, an edge computing resource allocation mechanism based on dynamic trust of blockchain is proposed in this study, regarding trust as an important basis for selection, at the same time all transaction information is stored on the blockchain to avoid attacks such as malicious tampering of information. It is verified through simulation experiments that as the number of transactions increases, the trust of high-quality edge servers will gradually increase to a stable state, while the trust of low-quality and malicious edge servers will gradually decrease as the number of transactions increases. Mobile users choose high-quality edge nodes for resource services, and transaction information is stored on the blockchain, which greatly improves security.

This paper addresses the security of edge computing in the process of resource allocation, taking into account the security of mobile users when making choices and the security of information storage, using smart contract technology to process the transaction process. However, the use of smart contracts is still on the surface. Later, we will consider in-depth study of smart contract technology and borrow smart contract technology to better improve the security of edge computing resource allocation.

References

1. Wang, L., Wu, C., Fan, W.: Summary of edge computing resource allocation and task scheduling optimization. J. Syst. Simul. **33**(3), 509–520 (2020)
2. Zhuang, X., Yang, B., Wang, X., et al.: Research on mobile edge computing security. Telecommun. Eng. Technol. Standard. **31**(12), 38–43 (2018)
3. Chen, L., Tang, H., You, W., Bai, Y.: Research on mobile edge computing security defense. J. Netw. Inf. Secur. **7**(1), 130–142 (2021)

4. Guo, S., Wang, R., Zhang, F.: Overview of the principles and applications of blockchain technology. Comput. Sci. **48**(2), 271–281 (2021)
5. Zhang, L., Liu, B., Zhang, R., et al.: Overview of blockchain technology. Comput. Eng. **45**(5), 1–12 (2019)
6. Dong, S., Wu, J., Li, H., et al.: Resource allocation method of mobile edge computing for priority tasks. Comput. Eng. **46**(3), 18–23 (2020)
7. Li, J., Zhang, Y., Pang, L., Ding, W., Sun, G., Yu, H.: Resource allocation and task scheduling methods in mobile edge computing. J. Chongqing Univ. Technol. (Nat. Sci.) **440**(11), 164–171 (2020)
8. Xue, J., An, Y.: New task offloading and resource allocation strategy based on edge computing. Comput. Eng. Sci. **42**(6), 959–965 (2020)
9. Yang, S.: A joint optimization scheme for task offloading and resource allocation based on edge computing in 5G communication networks. Comput. Commun. **160**, 759–768 (2020)
10. Alfakih, T., Hassan, M., Gumaei, A., et al.: Task offloading and resource allocation for mobile edge computing by deep reinforcement learning based on SARSA. IEEE Access **8**, 54074–54084 (2020)
11. Liao, J., Xian, W., et al.: Resource allocation and task scheduling scheme in priority-based hierarchical edge computing system. In: 2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES) (2020)
12. Nath, S., Wu, J.: Dynamic computation offloading and resource allocation for multi-user mobile edge computing. In: GLOBECOM 2020–2020 IEEE Global Communications Conference. IEEE (2020)
13. Wang, L., Jiao, L., Li, J., et al.: MOERA: mobility-agnostic online resource allocation for edge computing. IEEE Trans. Mob. Comput. **18**(8), 1843–1856 (2019)
14. Shen, S., Mao, Y., Li, L.: A general application scheme of blockchain for non-digital currency. J. Nanjing Univ. Posts Telecommun. Nat. Sci. Ed. **39**(1), 1–11 (2019)
15. Hu, X., Guo, S., Guo, S., et al.: Blockchain meets edge computing: a distributed and trusted authentication system. IEEE Trans. Industr. Inf. **16**(3), 1972–1983 (2020)
16. Xu, C., Wang, K., Li, P., et al.: Making big data open in edges: a resource-efficient blockchain-based approach. IEEE Trans. Parallel Distrib. Syst. **30**(4), 870–882 (2019)
17. Zhang, S., Lee, J.: A group signature and authentication scheme for blockchain-based mobile-edge computing. IEEE Internet Things J. **7**(5), 4557–4565 (2019)
18. Wu, B., Xu, K., Li, Q., et al.: Toward blockchain-powered trusted collaborative services for edge-centric networks. IEEE Netw. **34**(2), 30–36 (2020)
19. Ioini, N., Pahl, C.: Trustworthy orchestration of container based edge computing using permissioned blockchain. In: 2018 Fifth International Conference on Internet of Things: Systems, Management and Security, pp. 147–154 (2018)
20. Huang, Y., Zhang, J., Duan, J., et al.: Resource allocation and consensus on edge blockchain in pervasive edge computing environments. In: 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). IEEE, pp. 1476–1486 (2019)
21. Qiao, G., Leng, S., Chai, H., et al.: Blockchain empowered resource trading in mobile edge computing and networks. In: ICC 2019–2019 IEEE International Conference on Communications (ICC). IEEE (2019)
22. Xu, Q., Su, Z., Yang, Q.: Blockchain-based trustworthy edge caching scheme for mobile cyber-physical system. IEEE Internet Things J. **7**(2), 1098–1110 (2020)

23. Huang, J., Xia, X., et al.: Trust degree certification mechanism based on dynamic authorization. J. Softw. **30**(9), 2593–2607 (2019)
24. Song, J., Gu, T., Ge, Y., et al.: Smart contract-based computing ResourcesTrading in edge computing. In: 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications (2020)