Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Efficient and secure privacy protection scheme and consensus mechanism in MEC enabled e-commerce consortium blockchain

Guangshun Li[1], Haoyang Wu[1], Junhua Wu[1*] and Zhenqiang Li[1]

Abstract

The application of blockchain technology to the field of e-commerce has solved many dilemmas, such as low transparency of transactions, hidden risks of data security and high payment costs. Mobile edge computing(MEC) can provide computational power for blockchain, and can meet the demand for high real-time and low latency in e-commerce transaction systems. However, there are still some constraints in the MEC enabled e-commerce consortium blockchain, such as the leakage of user privacy information, low security of consensus algorithm and other security issues. In this paper, we propose a secure transaction model suitable for MEC enabled e-commerce consortium blockchain, aiming to ensure the efficiency of system transaction processing while improving the security of users' privacy information and transaction data. The model adopts the lightweight Paillier encryption algorithm to protect the security of user privacy information and transaction data to prevent the leakage of user privacy information, and optimizes the security of leader election phase of Raft consensus algorithm by introducing the shamir secret sharing protocol to improve the anti-Byzantine failure capabilities of Raft consensus algorithm. The effectiveness of the scheme proposed in this paper is demonstrated by experimental simulations.

Keywords Consortium blockchain, Mobile edge computing, Consensus algorithm, E-commerce, Privacy protection

Introduction

The continuous advance of computer network technology has made network applications such as e-commerce possible and provided an crucial foundation and impetus for its development [1]. E-commerce allows people to shop beyond time and space and is increasingly becoming an integral part of their daily life [2]. As e-commerce provides people with great convenience, it is growing considerablely rapidly, which also indicates that e-commerce has great development potential. Yet, with the rapid development of e-commerce applications, drawbacks such as limited transaction transparency, payment disputes, chargebacks, fraud have steadily surfaced, limiting the further development of e-commerce [3].

Blockchain has the characteristics of decentralization [4], non-tampering and transparency, which can cope with many problems in the field of e-commerce [5]. Compared with public blockchain, consortium blockchain has greater applicability, higher flexibility and more efficient speed of processing transactions, so it is more suitable for e-commerce scenarios [6].

In recent years, researches on combining e-commerce and blockchain have been in full swing. Blockchain can be applied to optimise multiple aspects of e-commerce. Blockchain can enable the traceability of goods in the cross-border e-commerce supply chain [7], achieve a safe and reliable reputation score for merchants of e-commerce [8], facilitate the implementation of financing for e-commerce retailers with limited funds [9], and so on. However, with the rapid

*Correspondence:
Junhua Wu
shdwjh@163.com
[1] School of Computer Science, Qufu Normal University, Rizhao, China

development of e-commerce consortium blockchain as well as the sharp rise in the number of users and e-commerce transactions, the performance of the existing consortium blockchain platform is insufficient to meet the demand of efficient and secure e-commerce transactions [10]. E-commerce consortium blockchains are sensitive to transaction latency and have high real-time requirements, so a technology is needed to provide fast computing power and reduce transaction load for e-commerce consortium blockchains.

MEC is a product of the continuous development of the cloud computing technology, and compared to the centralised deployment mode of the traditional cloud computing, it solves the problems of long communication time and large aggregated traffic, and provides better support for the low-latency and high real-time business [11]. Deploying the e-commerce consortium blockchain application on MEC can provide computing power and save resource overheads. And the membership identity management mechanism of the consortium blockchain guarantees the legitimacy of the identity of the edge computing devices in the system, and the consensus mechanism achieves the consistency of the synchronisation of the transaction data, so as to achieve a safe and efficient MEC Enabled e-commerce consortium blockchain transaction.

However, the existing consensus mechanisms also fail to meet the security and efficiency needs of MEC enabled e-commerce consortium blockchain. The core consensus algorithm will affect the performance of the blockchain [12]. The Practical Byzantine Fault Tolerance (PBFT) consensus algorithm is commonly employed by Consortium blockchain as its core consensus algorithm, which can ensure certain system security; however, the PBFT algorithm has performance shortcomings, namely, the communication complexity is excessively high and the scalability is relatively low [13]. Furthermore, it has high latency in the case of network instability. In addition to PBFT algorithm, Raft consensus algorithm is also one of the consensus algorithms commonly utilized in consortium blockchain. Raft consensus algorithm is a simplified and optimized version of Paxos algorithm [14]. Since Paxos is more complex and difficult to understand and implement, Raft algorithm was born and has widely used in consortium blockchain and private blockchain. In terms of performance and scalability, the Raft algorithm has advantages over PBFT algorithm and is therefore more widely used in scenarios requiring high throughput and performance demands [15]. However, in many applications, such as e-commerce trading platforms and other consortium blockchain applications, security requirements are also high, so further optimization of Raft consensus to enhance its security is needed.

Furthermore, with the extensive application of the consortium blockchain and the introduction of MEC devices, transaction data and user privacy are vulnerable to security risks, which is the most important concern of e-commerce trading platforms [16]. With growing concerns about user privacy and data security, a variety of regulations and policies have been proposed to limit access to data [17]. Even if the consortium blockchain has the node access control mechanism, user privacy and transaction data of the blockchian system still have the risk of being leaked [18]. For example, if the transaction privacy information in the e-commerce system is not protected, it may result in issues such as disclosure of users' real identitities, leakage of transaction amount, and exposure of merchants' trade secrets [19]. As a result, such privacy and security issues can seriously harm users' interests and diminish their motivation to join e-commerce consortium blockchain, which has a negative impact on the widespread implementation of e-commerce consortium blockchain [20]. Precisely for this reason, privacy protection has become the focal point of our study.

To address the above issues, we propose a homomorphic encryption-based e-commerce transaction model deployed on MECs and improved consensus algorithms to ensure user privacy and security while improving system throughput and scalability, and the high level architecture of our model is shown in Fig. 1. The working principle of our model is to enhance user transaction and privacy security using homomorphic encryption and improve the Raft algorithm to increase its ability to resist malicious nodes, which is described in detail in "Background" section. Our main contributions in this work are as followings:

1. This work proposes a transaction model based on the lightweight Paillier encryption algorithm to encrypt the user's sensitive information and source transaction data, e.g. the balance of the user's account, the information of the merchandise, etc. The privacy of users and the transaction data of MEC enabled e-commerce consoritum blockchain are protected through the transaction model.
2. We optimize the leader election phase of the Raft consensus algorithm and propose a novel consensus algorithm TD-Raft. In the election phase of TD-Raft, leader nodes are elected through both shamir threshold secret sharing scheme and voting to prevent Byzantine nodes from cooperating to elect Byzantine node into leader node, thus improving the anti-Byzantine failure ability of Raft consensus algorithm.

The remainder of this paper is organized in the following order. "Related work" section presents the related

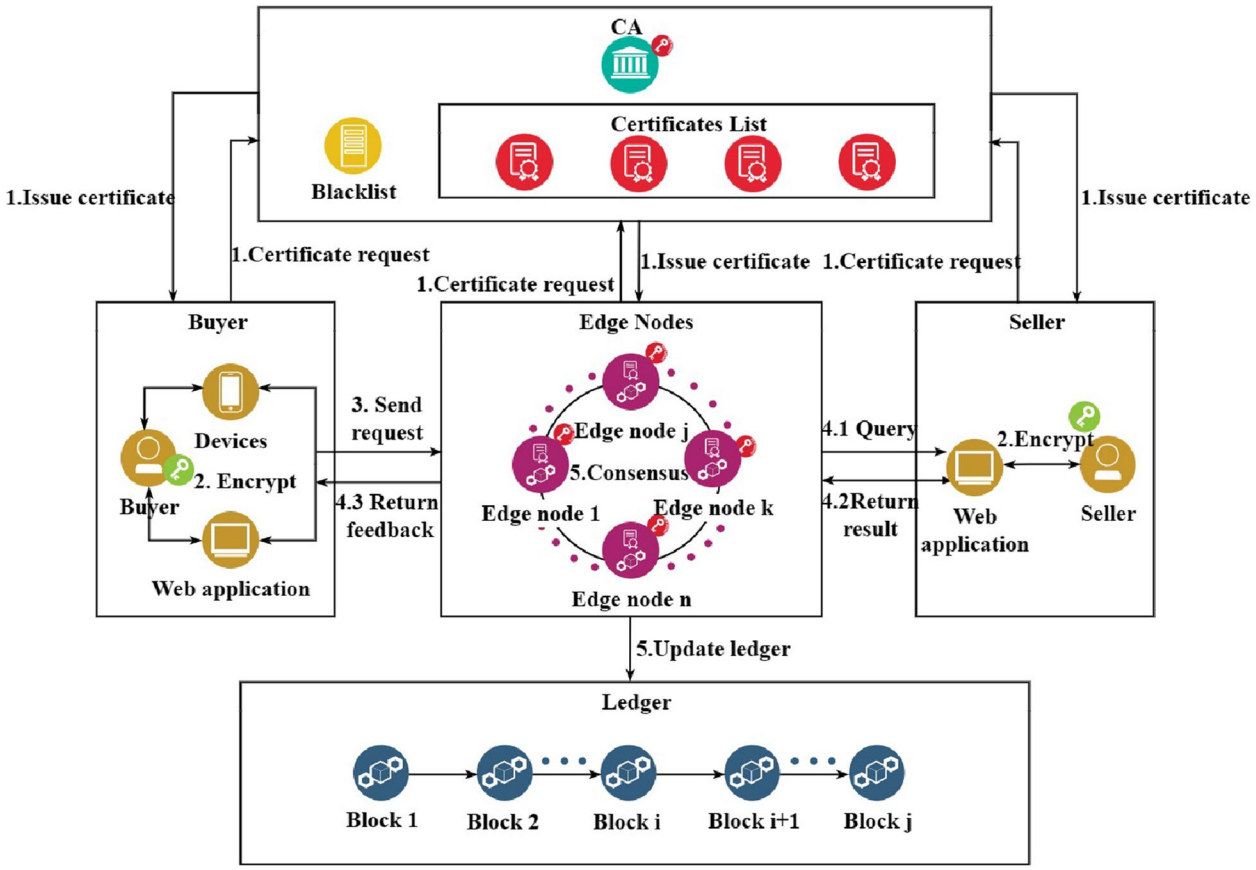

Fig. 1 High level architecture of e-commerce transaction model

work. "Background" section describes the research background, including the threat model and the design goals. The details of the model design and the associated algorithms are depicted in "Model design" section. In addition, we provide the evaluation results of the simulation in "Security analysis" section. And we disscuss the results and weaknesses of our study in "Evaluations" section. Finally, we conclude the work and point out the future work in "Discussion" section.

Related work

With the extensive development of consortium blockchain, the performance constraints are gradually highlighted in a large number of transactions. Especially when the e-commerce consortium blockchain is in the peak transaction period, the demand for high performance and low latency is more prominent. And MEC technology can provide efficient data processing capability for blockchain to achieve high efficiency and real-time processing transactions, so the introduction of MEC technology in the e-commerce consortium blockchain has become an inevitable trend.

MEC technology are widely used in areas such as Internet of vehicle, video QoS optimisation, etc., as they are able to provide services and cloud computing capabilities close to the user, creating a high-performance and low-latency service environment. To reduce the negative impact of uneven or discrete distribution of edge servers in multimedia IoT systems, Xu et al. [21] proposed a traffic flow prediction driven resource reservation method and used a deep spatio-temporal residual network to predict future traffic and estimate the amount of multimedia services offloaded to edge servers to determine the offload destinations. In order to reduce the total overhead of the Telematics system, Yang et al. [22] utilised an optimised Fuzzy C-means algorithm to cluster the vehicles and other edge devices, and proposed a Deep Q network based task offloading algorithm to obtain an optimal task offloading scheme. In order to determine optimal offloading decisions in highly dynamic and heterogeneous edge cloud environments, Xu et al. [23] proposed a dynamic offloading strategy based on game theory combined with convolutional neural network partition for vehicular edge networks to use resources more efficiently and reduce latency. Qi et al. [24] utilised the cybertwin as a

centric controller and took advantages of crowdsourcing technology to attract mobile users to follow the specified path and share the network resources with other users to guarantee communication performance between edge devices. Most of the above researches on MEC technology are mostly focused on the field of Telematics, which enhances the communication performance of vehicular networks. However, the MEC technology is rarely applied in the field of e-commerce, whereas it is able to provide high real-time transaction processing capability, which is well suited to play a powerful role in e-commerce, thus improving the efficiency of e-commerce transactions.

Another mainstream method to enhance the performance of consortium blockchain is currently to improve the consensus algorithms. Since Raft consensus algorithm is better to understand, easier to implement, and as efficient as Paxos consensus algorithm, it is easier to apply in practical systems and is widely used in permissioned blockchains. Scholars have made numerous positive efforts to improve the security of Raft consensus algorithms in order to accommodate more scenarios. Wang et al. [25] proposed an improved Raft algorithm "hhRaft", which optimizes the Raft consensus process by introducing a new monitoring role. The monitoring node supervises the candidate nodes and compares the computation results of transactions, which improves transaction throughput and Byzantine resistance, reduces consensus latency, and is suitable to use in in a high real-time, highly adversarial environment. Inspired by the original Raft algorithm and the Practical Byzantine Fault Tolerance algorithm, Copeland et al. [26]proposed a Byzantine fault-tolerant Raft consensus algorithm. The algorithm maintains the security, fault tolerance, and activity of the Raft algorithm in the event of a Byzantine failure, while also pursuing simplicity and comprehensibility of the original Raft consensus. Fu et al. [27] optimized the Raft consensus algorithm for the Hyperledger Fabric platform in terms of log replication and leader election to address the performance degradation problem caused by the blockchain backup mechanism, which aims to reduce the communication complexity and the election time. To optimize the Byzantine fault tolerance of the Raft algorithm, Tian et al. [28] proposed a Byzantine fault-tolerant algorithm B-Raft incorporating the Schnorr signature mechanism, which combines the signature mechanism with the Raft algorithm to provide Byzantine fault tolerance. Most of the aforementioned studies have focused on the improvement of the Raft consensus algorithm on enhancing the security of the algorithm, with little consideration given to the efficiency of the consensus algorithm. Since the e-commerce trading platform needs to ensure the security of transactions and the speed of processing transactions at the same time, it is necessary to consider a consensus algorithm that is more suitable for a safe and high-speed transaction environment.

In addition, privacy protection is also a key issue in the e-commerce consortium blockchain, and the security of users' private information as well as transaction data are of vital importance. To ensure consumer privacy security, Azad et al. [29]proposed PrivBox, a decentralized reputation system for privacy protection, which achieves the goal of privacy protection in e-commerce systems by using a homomorphic cryptosystem and non-interactive zero-knowledge proofs to compute the reputation of a retailer or service provider using users' feedback. Ghadamyari et al. [30] proposed a novel privacy-preserving approach for statistical analysis of health data in distributed blockchain networks using the Paillier homomorphic encryption algorithm to improve the security level of the data while minimizing the amount of unnecessary information exposed to third parties, providing researchers with accurate analysis results and protecting patient privacy. In order to protect the privacy and security of personal data in the Internet of Medical Things, Zhou et al. [31] proposed a unique clustering-based approach for participant selection using social context data, creating different groups of edge participants and performing group-specific federated learning. In order to achieve data privacy protection in IoV, Ma et al. [32] proposed a blockchain-based secure data sharing scheme for IoV, which enables reliable sharing of IoV data through smart contracts and processes the sensitive part of the data whith homomorphic encryption and zero-knowledge proof. To cope with the challenges posed by the access of a large number of heterogeneous edge devices for the security management and data privacy protection of edge computing architectures, Xu et al. [33] proposed a lightweight edge computing data privacy protection scheme based on blockchain and homomorphic encryption, and designed a blockchain data encryption transmission scheme to guarantee data transmission between edge nodes, which supports verification of the legitimacy and correctness of transactions in the form of ciphertext. Zhou et al. [34] designed three-layer federated reinforcement learning framework with an end-edge cloud structure with an edge cloud structure and designed a dual reinforcement learning scheme to facilitate lightweight training and real-time processing of models in high-speed mobile networks. Zhou et al. [35] proposed a peer-to-peer based privacy-aware asynchronous federated learning framework for enabling secure and resilient decentralised model training of modern mobile robotic systems in 5G and beyond networks to safeguard the privacy security and performance of robotic systems. Mishra et al. [36] introduced

a blockchain-based methodology to employ a certificateless public auditing model against malicious and procrastinating auditors with efficient user revocation to address security concerns. However, most of the privacy-preserving methods are inefficient and cannot meet the high real-time and low-latency requirements of e-commerce consortium blockchains. In e-commerce transactions, there is a need for a lightweight privacy-preserving approach that simultaneously meets the high performance and security requirements.

In this paper, the proposed blockchain-based e-commerce transaction system model is able to guarantee the security of transaction data while guaranteeing the privacy of consumers and merchants and improves the security of consensus through an improved Raft consensus algorithm to achieve a low latency and high security transaction system.

Background

Threat model

We assume that the malicious nodes originate from the nodes of various organizations in the e-commerce trading platform. Due to the transparency of transactions and the openness of the blockchain, the transaction information on the blockchain can be easily obtained. The attack behaviors of malicious nodes in blockchain systems include leaking transaction and privacy information and interfering with the consensus process, which can pose a threat to the efficiency and security of the whole system. Our model aims to avoid the leakage of users' transaction data and private information, as well as to eliminate the threats to the consensus process posed by malicious nodes.

Since data information on the blockchain is public, malicious nodes can easily get access to users' transaction data and steal users' private information, such as leaking buyers' account information, source data of transactions or other private information data such as merchants' product price and inventory during e-commerce transaction process, which may result in problems such as privacy leakage, malicious competition among different merchants or transaction tracking.

The behaviors of malicious nodes in the consensus process are as follows. The malicious follower nodes ignore messages sent by other nodes, do not respond to requests sent by leader nodes, send wrong messages to leader nodes and tamper with or forge messages sent by other nodes; the malicious candidate node increases the number of terms by maliciously making its term number larger than the term number of other nodes, in order to be successfully elected as the leader node after recieving the votes of majority nodes in the leader election process.

Design goals

Our goal is to design an efficient and secure blockchain-based scalable e-commerce transaction model that can quickly process transaction requests sent by clients and synchronize transaction information to the ledger without compromising transaction information and user privacy. Our model aims to satisfy three objectives, namely, high throughput, security and scability of the blockchian system.

Throughput: Throughput is one of the important criteria to measure the performance of a system. Without considering the influence of other factors such as network, the fundamental purpose of our model is to ensure high throughput of blockchain system. To reduce transaction delays in e-commerce scenarios, for example, it is necessary to respond to transaction requests from users in a timely manner and process transactions promptly. Therefore, the system adopts the Raft consensus algorithm, which significantly reduces the communication overhead between nodes compared to the traditional PBFT consensus algorithm and reduces the system latency, thus ensuring high throughput of the system.

Security: The security objective implies that our proposed model needs to guarantee not only the security of users' transaction data and privacy in the blockchain system, but also the security of consensus. From the start, the private information of buyers and sellers should be hidden and the confidentiality of transaction data should be guaranteed to prevent the possibility of leakage. Furthermore, the security goal requires the consensus algorithm to resist malicious attack behaviors in the e-commerce transaction environment, which is reflected in reducing the possibility of malicious nodes becoming the leader node of the consensus, so as to ensure the smoothness of the consensus process.

Scalability: Scalability has become an important limiting factor for the implementation and development of the blockchain system, so the model we proposed should ensure the scalability of the blockchain system. As the number of nodes in the system grows, the transaction communication between nodes of our improved consensus algorithm is linearly increasing, preserving the high efficiency of Raft consensus algorithm, thus ensuring the scalability of the system.

Model design

In this section, we propose a homomorphic encryption-based e-commerce transaction system and introduce the improved Raft consensus. First, we explain the general framework and transaction flow of the e-commerce transaction model in "Homomorphic encryption-based e-commerce transaction system" section, and detail the

improved Raft consensus process in "Improved Raft consensus algorithm TD-Raft" section.

Homomorphic encryption-based e-commerce transaction system

This section focuses on the homomorphic encryption-based e-commerce transaction system, which encrypts and protects the transaction data of the blockchain through the improved paillier encryption algorithm, ensuring the privacy of users and the security of transaction data. The components of our proposed model include CA, buyer nodes, seller nodes, and other nodes. The flow of e-commerce transactions based on lightweight homomorphic encryption is shown below.

Step1. System initialization. First, CA will first verify whether each node applying to join the consortium blockchain is qualified, and each node of the system must register with CA and then join the blockchain. Each node obtains its own digital certificate from CA, which includes the node's public-private key pair (pk_i, sk_i), as shown in step 1 of Fig. 1. In addition, CA uses the improved paillier encryption algorithm to generate the homomorphic secret key of the consortium blockchain system.

1) First, CA selects two large prime numbers p and q and ensures that $p \equiv q = 3\,(mod\,4)$ and $gcd(p-1, q-1) = 2, \delta = (p-1)(q-1)/2$.
2) Then, CA computes $w = pq$, $g = w + 1$ and selects a random number $a \in Z_n^*$, and $h = -a^2\ mod\ n$.
3) As a result, the homomorphic secret key has been generated. The homomorphic public key (w, h) is sent to all nodes of the blockchain system, and the homomorphic private key δ is kept by CA.

Step2. Buyers and sellers uses homomorphic public key to encrypt the private data. Firstly, buyer uses homomorphic public key to encrypt its account balance b, account balance after purchasing goods b', purchased goods c and its quantity num respectively. And the seller uses the homomorphic public key to encrypt the price of the item v, the inventory of the item r, and the inventory of the item r' after transaction, as shown in step 2 of Fig. 1.

Step3. Buyer initiates a transaction request. Before the buyer initiates a transaction request, it first uses its private key to sign the transaction request, then uses the public key of the leader node to encrypt it, and sends the encrypted message request to the leader node of the consensus through the SDK, as shown in step 3 of Fig. 1.

Step4. Leader node verifies the legitimacy of the transaction. After the leader node receives the transaction request message, it needs to verify the transaction request first, that is, it needs to verify whether the buyer of the transaction has enough transaction balance to buy these commodity and whether the seller has enough commodity in stock.

1) After the Leader node receives the transaction request from the buyer, it first decrypts the transaction request using its own private key. And then the leader node uses the buyer node's public key to verify authenticity of the message, and finally gets $Enp(c)$.
2) Then, the leader node sends a request message to the seller node to query the commodity c, as shown in step 4.1 of Fig. 1. The seller node sends the transaction message of the corresponding commodity c to the leader node, as shown in step 4.2 of Fig. 1. And the leader node decrypts it by its own private key to verify whether the signature of the seller node is valid, and then gets the detail information of the transaction information.
3) The leader node verifies whether $En_p(b) = En_p(b') \times En_p(sum)$ holds, where $En_p(sum) = En_p(v) \times En_p(v) \times En_p(v) \times \ldots$, thus verifying whether the account balance of the buyer node before purchasing the commodity is the sum of the price of the goods purchased and the account balance after purchasing the commodity, i.e., verifying whether the account balance of the buyer node is sufficient to purchase the commodity; similarly, the leader node verifies whether $En_p(r) = En_p(r') \times En_p(num)$ holds by verifying whether the commodity inventory of the seller node is the sum of the number of commodity bought by the buyer node and the commodity inventory after the purchase of commodity, i.e., verify whether the commodity inventory of the seller node is sufficient for the number of commodity bought by the buyer node.
4) If the leader node validates the account balance of the buyer node and the inventory of the seller node successfully, the transaction is validated legally and the leader node will broadcast the transaction to the other follower consensus nodes; otherwise, whenever the inventory of the seller node is insufficient or the account balance of the buyer node is not adequate, the transaction is judged to be illegal and the leader node sends a message to the buyer and seller nodes that the validation has failed, as shown in step 4.3 of Fig. 1.

Step5. Consensus nodes reach consensus on transactions and execute them. After the verifying that the transaction is legal, the consensus node executes the transaction. And the homomorphic encrypted precompiled smart contract is invoked, the account balance

of the buyer node is deducted, and transferred to the account of the seller node. The seller node will send the corresponding quantity of commodities to the buyer node. If the seller node refuses to ship the commodities or has other malicious behaviors, 10 times the transaction amount of the seller node's account will be deducted as the corresponding penalty and the seller node will be added to the blacklist. Ensuring the smart contract is executed, the result of the ciphertext calculation of the transaction is obtained. After the transaction is executed, the consensus nodes will broadcast the transaction and verify it, and the transaction will be synchronized to the ledger after successful verification, as shown in step 5 of Fig. 1. The ledger remains the ciphertext after homomorphic encryption of the transaction, i.e., the account balance of the buyer node is $En_p(b')$, thus ensuring the privacy and security of the buyers and sellers as well as the transaction data.

During the whole process of the transaction, the user privacy information of buyers and sellers as well as transaction data are encrypted by the improved paillier encryption algorithm, so that all nodes in the blockchain system cannot obtain the privacy data of buyers and sellers, which protects the privacy of users; in addition, all the messages sent by buyers and sellers as well as consensus nodes are signed with their own digital signatures, which can prevent message forgery and tampering and ensure the authenticity and security of data. In addition, compared with other privacy protection models based on Paillier encryption, our proposed privacy protection model based on lightweight Paillier encryption is able to guarantee the same security while the time consumed by algorithmic operations decreases drastically. So our lightweight privacy protection scheme has better performance than other schemes, and is more adaptable to MEC enabled e-commerce consortium blockchains.

Improved Raft consensus algorithm TD-Raft

This section describes the proposed improved Raft consensus algorithm TD-Raft based on secret trapdoor sharing and VRF in detail. The edge computing nodes are the consensus nodes of the system. We focus on describing the leader election phase of the consensus algorithm in this section. The algorithm utilizes VRF and secret sharing methods to improve the security of leader election. Since the selection of the candidate node using VRF has randomness, thus ensuring the security of candidate node selection. And less than two-thirds of the nodes cannot conspire to get the trapdoor of the chameleon hash. As long as the candidate node has two-thirds of the valid trapdoor pairs, it can recover the trapdoor. Only the node holding the trapdoor has the opportunity to participate in the new round of leader election.

By this method, the occurrence of malicious nodes changing their term numbers at will and thus becoming leader nodes is prevented, and the Byzantine resistance of the consensus algorithm is improved. In addition, it prevents malicious leader nodes from ignoring, tampering or forging client commands by setting a synchronization timeout mechanism. The security and efficiency of the consensus algorithm are guaranteed by the above methods.

The consensus nodes in this model inherit the states from the Raft consensus algorithm, i.e., all consensus nodes have three states, which are follower, candidate and leader states. First of all, the initial state of all nodes are follower nodes. If the follower node does not receive a heartbeat message from the leader node within the heartbeat timeout, the follower node will add its term value and a new candidate node will be selected through VRF. Then the candidate node requests the trapdoor secret key from other follower nodes. If it succeeds in recovering the trapdoor, then it will use the trapdoor secret key to request votes from other follower nodes. If it gets votes from more than two-thirds of the follower nodes, the candidate node will be converted to the leader state. Thereafter, the leader node is responsible for communication with the client and replication of logs. The high level process diagram of the improved consensus TD-Raft is illustrated in Fig. 2.

1. **Leader node generates the chameleon hash and selects the parameters.** First, let the number of all consensus nodes be n. After the candidate node becomes the leader node, it will first get the public and private keys of the chameleon hash by the key generation algorithm through $Ch\_Gen(1^\lambda) = (pk_{ch}, sk_{ch})$, and then generate the hash $Ch\_Hash(pk_{ch}, m, \theta) = (h, k)$ with any message m and random number θ and broadcast (pk_{ch}, m, θ) to the other follower nodes. Then the leader node determines a large prime number p and the chameleon trapdoor is represented by modulo p. All follower nodes are involved in the preservation of the chameleon trapdoor, and at least $\left\lceil \frac{2(n-1)}{3} \right\rceil$ follower nodes are required for reconstruction of sk_{ch} if the trapdoor sk_{ch} is to be recovered.
2. **Leader node splits the chameleon hash trapdoor.** First, the leader node randomly picks $\left\lceil \frac{2(n-1)}{3} \right\rceil - 1$ modulo p numbers, denoted as $s_1, s_2, ..., s_{\left\lceil \frac{2(n-1)}{3} \right\rceil}$ respectively, which enables to obtain the polynomial

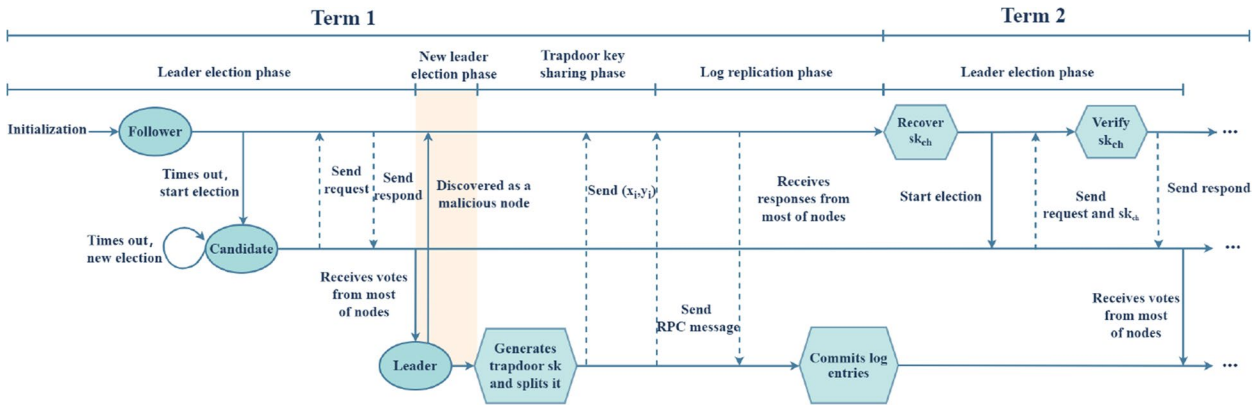

Fig. 2 Diagram for the process of improved Raft consensus

$$sk_{ch}(x) \equiv s + s_1 x + ... + s_{\left\lceil \frac{2(n-1)}{3} \right\rceil - 1} x^{\left\lceil \frac{2(n-1)}{3} \right\rceil - 1} (mod\ p) \tag{1}$$

and this polynomial satisfies $sk_{ch}(0) \equiv s\ mod\ p$. Then, the leader node selects $n-1$ different integers which are less than p, and for each integer calculates the number of pairs (x_i, y_i), where $y_i \equiv sk_{ch}(x_i)\ mod\ p$. Finally, the leader node encrypts the pairs $(x_i, y_i)(i = 1, 2, ..., n-1)$ with each node's public key and then secretly transmits them to each follower node. After the follower node receives the message sent by the leader, it decrypts it using its private key. If the node is unable to decrypt the message, the message will be ignored. The polynomial $sk_{ch}(x)$ is confidential and can be destroyed. The pseudo code of the algorithm for the leader node to generate and split the secret chameleon hash trapdoor key is shown in Algorithm 1.

3. **Candidate node sends request for trapdoor secret key.** When the follower node does not receive a heartbeat message from the leader node within the heartbeat timeout time, the follower node considers that the leader node is down and increases its own term value. Since VRF has the characteristics of verifiability and randomness, i.e., the random number generated by the method can be verified to be valid, and the random number generated to select the candidate node can guarantee the randomness and security of the candidate selection. The chameleon hash value h generated by the leader node is used as the seed for the generation of random number, and the candidate node is selected by the random number. The selected candidate node sends a request for trapdoor to other follower node.
4. **Follower nodes send trapdoor pairs after verifying the request message.** After receiving the request from the candidate node, other follower nodes first verify whether $term > currentTerm$ is valid. If the verification fails, the follower node will reject the request. Otherwise the follower node detect whether it receives the heartbeat message from the leader node, if not, it will send the trapdoor pairs owned to the candidate node after encrypting the message with the candidate node's public key.
5. **Candidate node recovers the trapdoor.** We assume that candidate node can recover the complete trapdoor only if it receives more than two-thirds of the trapdoor pairs sent by follower nodes, and let their pairs be $(x_1, y_1), (x_2, y_2), ..., \left(x_{\left\lceil \frac{2(n-1)}{3} \right\rceil - 1}, y_{\left\lceil \frac{2(n-1)}{3} \right\rceil - 1}\right)$. When the candidate node receives the remaining $\left\lceil \frac{2(n-1)}{3} \right\rceil - 1$ pairs from the follower and decrypts them with its own private key, it calculates the polynomial

$$f(x) \equiv \sum_{t=1}^{\left\lceil \frac{2(n-1)}{3} \right\rceil} y_i \prod_{j=1, j\neq t}^{\left\lceil \frac{2(n-1)}{3} \right\rceil} \frac{x - x_j}{x_t - x_j} (mod\ p) \tag{2}$$

Take the constant term $f(0)$ of the polynomial, which is the trapdoor sk_{ch} sought by the candidate node. And the pseudo code of the algorithm for the candidate node to reconstruct the secret chameleon hash trapdoor key is shown in Algorithm 2.

6. **A new leader node is elected.** The candidate node uses the trapdoor sk_{ch} to implement the collision of the chameleon hash after the computed trapdoor. First of all, the candidate node finds out r' through $Ch\_Forge(sk_{ch}, m, r, m')$ and then broadcasts m' and r'. The other follower nodes receive the message and verify whether $Ch\_Hash(m, r, pk_{ch}) = Ch\_Hash(m', r', pk_{ch})$ holds and verify the validity of the hash collision of the can-

didate node. When the verification is passed, a vote is cast to the candidate node. And if the candidate node receives votes from more than two-thirds of the follower nodes, it will become the new leader node. The new leader node randomly generates new m and r, sends heartbeat messages to other nodes and broadcasts (pk_{ch}, m, r) and finally sends the private key sk_{ch} to other follower nodes after splitting it. The new leader node is responsible for selecting the appropriate number of legitimate transactions in the transaction pool in chronological order and packaging them into a block, which is attached to the distributed ledger. Then the leader node broadcast the information to other consensus nodes to synchronise the world state.

The TD-Raft consensus we proposed can effectively prevent malicious nodes from becoming leader nodes through collusion attacks in the leader election phase, and improve the ability of Raft consensus algorithm to resist Byzantine attacks. Compared to the PBFT consensus algorithm, our proposed TD-Raft consensus algorithm has higher performance, and the anti-Byzantine capability is consistent with the PBFT consensus algorithm. Compared with the Raft consensus algorithm, the performance of the TD-Raft consensus algorithm whose anti-Byzantine capability is greatly improved is slightly lower, but the gap is not large. The leader election process of TD-Raft consensus algorithm is shown in the Fig. 3.

Algorithm 1 Leader node generates and splits Chameleon hash trapdoor

Require: $n, currentLeader, \lambda, pp$;
Ensure: (x_i, y_i);
if $currentLeader \leftarrow -1$ **then**
  election()
end if
if Node.state = currentLeader **then**
  $p \leftarrow GenerateBigPrimeP(100)$
  $sk_{ch}, CH \leftarrow chameleonHash(\lambda, pp)$
  for i←1 to $\lceil \frac{2(n-1)}{3} \rceil - 1$ **do**
    $s_i \leftarrow rand.Seed(time.Now().UnixNano(p))$
  end for
  $s(x) \leftarrow sk_{ch} + s_1 x + ... + s_{\lceil \frac{2(n-1)}{3} \rceil - 1} x^{\lceil \frac{2(n-1)}{3} \rceil - 1}$
  for i←1 to n-1 **do**
    $x_i \leftarrow rand.Seed(time.Now().UnixNano(p))$
    $y_i \leftarrow s(x_i) mod p$
  end for
  return (x_i, y_i)
end if

Algorithm 2 Candidate node reconstructs the trapdoor

Require: $(x, y), n$;
Ensure: sk_{ch};
if $node.state = candidate$ and $heartBeat = false$ **then**
  if Count$((x, y)) > \lceil \frac{2(n-1)}{3} \rceil - 1$ **then**
    for $t \leftarrow 1$ to $\lceil \frac{2(n-1)}{3} \rceil - 1$ **do**
      for $i \leftarrow 1$ to $\lceil \frac{2(n-1)}{3} \rceil$ **do**
        for $j \leftarrow 1, j \neq i$ to $\lceil \frac{2(n-1)}{3} \rceil$ **do**
          $f(x) = \sum_{i=1}^{\lceil \frac{2(n-1)}{3} \rceil} y^i \prod_{j \leftarrow 1, j \neq i}^{\lceil \frac{2(n-1)}{3} \rceil} \frac{x - x_j}{x_t - x_j} (mod\, p)$
        end for
      end for
    end for
    $sk_{ch} \leftarrow f(0)$
    return sk_{ch}
  end if
end if

Security analysis

In the process of e-commerce transactions, users' private information has the possibility of being stolen and tracked by malicious users, in addition, a large amount of transaction data may also face the risk of leakage. To address the above problems, all the edge computing nodes can only join the system if they are authenticated. Then we encrypt users' private information by using lightweight paillier encryption algorithm, and all other nodes and users cannot obtain buyers' and sellers' private data, and the transaction information is also stored on the blockchain in the encrypted ciphertext state, which can realize the verification of transaction legitimacy while ensuring the security of users' private information and the confidentiality of transaction data. In addition, all messages sent by nodes and sellers and buyers are signed using digital signatures, ensuring the authenticity of messages.

In the consensus process, the follower node may force the leader node to crash and replace it as the new leader node by increasing the term value. For this malicious behavior, the leader node split the chameleon hash private key namely the trapdoor. And only when the leader node is down and more than two-thirds of the follower nodes provide trapdoor pairs to the candidate node, the candidate node can recover the trapdoor private key. After verifying the validity of the trapdoor, the other follower nodes vote for the candidate node. If more than two-thirds of the follower nodes vote for the candidate node, the candidate node can become the leader node, which effectively prevents the follower node from becoming the leader node by arbitrarily increasing the

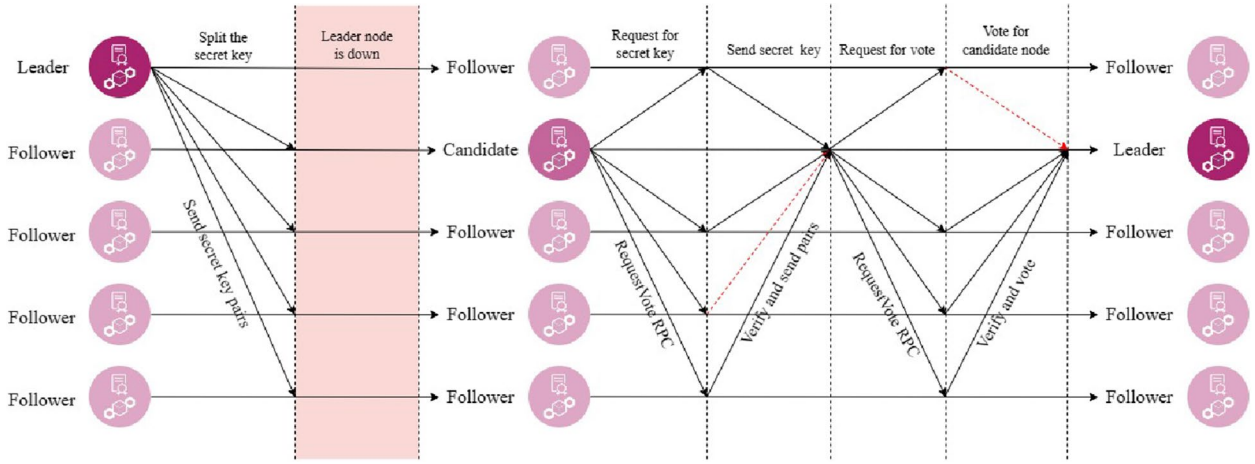

Fig. 3 The leader election process of TD-Raft consensus

term value. In addition, it is possible for a follower node to become a candidate node by increasing the term value and participate in the leader election. In order to prevent this situation, we implement the selection of the candidate node by using VRF. Since the seed in VRF is a randomly generated hash value of the previous leader node, the malicious node cannot be forged in advance, thus ensuring the security and randomness of the candidate node selection, and thus the security of the leader node selection. Therefore, our proposed TD-Raft consensus algorithm increases the ability of Raft consensus algorithm to resist malicious nodes.

Evaluations

Experiment environment

The experiment uses Intel(R) Core(TM) i5-8250U 1.60GHz CPU, 8GB memory hardware, and Windows 11 Home Edition 64-bit operating system. First of all, the consumption time of homomorphic encryption algorithm operation is tested experimentally using Python language in Pycharm platform. And then We deploy and install Hyperledger Fabric 1.4.4 on Ubuntu 20.04 to simulate the experiment environment of the consortium blockchain and install Hyperledger Caliper 0.2.0 to test the throughput and other metrics of the consensus algorithm of the consortium blockchain.

Experiment results

First of all, we compare the lightweight paillier encryption algorithm with the paillier encryption algorithm in terms of secret key generation time, encryption time and decryption time. The algorithm is implemented using Python language and the length of the secret key is set to 2048bit. And Fig. 4 showed that the secret key generation time of the improved paillier encryption algorithm is increased by about 72.0%, while its encryption time and decryption time are both reduced by 73.6% and 76.1% respectively. Table 1 shows the consumption time of operation of the two Paillier encryption algorithm. The lightweight Paillier encryption algorithm transforms the high order power operation into a low order power operation in order to optimise the encryption operation, thus transforming the secret key generation operation from an integer to a power operation and the consumption time of the key generation operation elevates. And since the number of operations of key generation in this transaction model is much smaller than the number of operations of encryption and decryption, the increase in the time of secret key generation is enough to be negligible and the time of encryption and decryption is reduced. Thus, the improved lightweight paillier encryption algorithm is more suitable for the high real-time e-commerce transaction model.

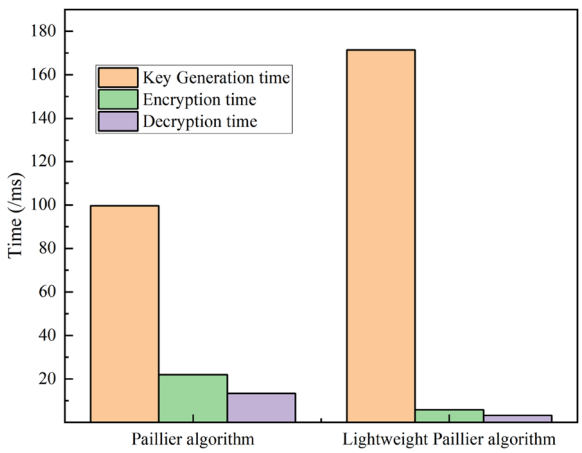

Fig. 4 Comparison of the improved paillier encryption algorithm and paillier encryption algorithm in secret key generation time, encryption time and decryption time

Table 1 Comparison of consumption time for different operations of two Paillier encryption algorithms

Algorithm	Paillier algorithm	Lightweight Paillier algorithm
Consumption time(/ms)		
Key generation time	99.6	171.4
Encryption time	22.0	5.8
Decryption time	13.4	3.2

Furthermore, We compare the performanceof three consensus algorithm. We set the block size to 1MB and the timeout of the three consensus algorithms as shown in the Table 2. We run each consensus algorithm 100 times separately and take the average of all the results as the final experimental results. the traditional PBFT consensus algorithm and Raft consensus algorithm with our proposed TD-Raft consensus algorithm in terms of throughput with different number of nodes to verify the performance of TD-Raft algorithm in terms of system throughput, and the experimental results are shown in Fig. 5.

As shown in Fig. 5, the throughput of Raft consensus algorithm has been the highest among the three algorithms for different number of nodes. The throughput of our proposed TD-Raft consensus algorithm is lower than that of Raft consensus algorithm, and the throughput of PBFT consensus algorithm has been significantly lower than that of the other two algorithms. With the gradual increase of nodes, the throughput of all three consensus algorithms gradually decreases. Compared with the Raft consensus algorithm, the TD-Raft algorithm takes longer time to recover the trapdoor during the leader election phase in order to enhance the security of the consensus algorithm, so the throughput of the TD-Raft consensus

Table 2 The timeout of three consensus algorithm

Algorithm	PBFT	Raft	TD-Raft
Timeout(/s)			
10 nodes	0.020	0.010	0.012
20 nodes	0.035	0.015	0.018
30 nodes	0.053	0.023	0.027
40 nodes	0.078	0.034	0.036
50 nodes	0.105	0.052	0.057
60 nodes	0.138	0.071	0.074
70 nodes	0.169	0.099	0.109
80 nodes	0.201	0.116	0.122
90 nodes	0.216	0.131	0.143
100 nodes	0.250	0.150	0.171

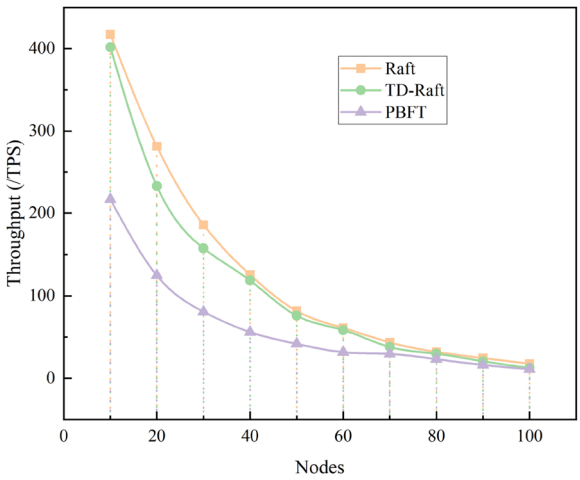

Fig. 5 Comparison of TD-Raft with Raft and PBFT in throughput

algorithm is lower than that of the Raft consensus algorithm. And the communication complexity of TD-Raft consensus algorithm is lower than that of PBFT algorithm, so the throughput of TD-Raft consensus algorithm is higher than that of PBFT algorithm. Thus, it can be concluded that the TD-Raft consensus algorithm proposed in this paper is lower than the Raft consensus algorithm in terms of throughput, but higher than the PBFT consensus algorithm commonly used in consortium blockchain, and the throughput of the TD-Raft consensus algorithm does not decrease significantly with the increase of nodes, and can replace the PBFT consensus algorithm as the consensus algorithm for the security of consortium blockchain.

Similarly, we compare the traditional PBFT consensus algorithm and Raft consensus algorithm with our proposed TD-Raft consensus algorithm in terms of consensus latency time for different number of nodes to verify the performance of TD-Raft algorithm in terms of consensus latency time, and the experimental results are shown in Fig. 6.

As shown in Fig. 6, the latency time of Raft consensus algorithm is the lowest among the three consensus algorithms. The latency time of PBFT consensus algorithm is always higher than the latency time of the other two algorithms, and the latency time of our proposed TD-Raft consensus algorithm is higher than Raft algorithm and lower than PBFT consensus algorithm. With the gradual increase of nodes, the consensus latency time of all three consensus algorithms gradually increases. It can be seen that, similar to the throughput, the TD-Raft consensus algorithm sacrifices some performance in terms of consensus latency time compared to the Raft consensus algorithm because it needs to ensure the Byzantine resistance of the consensus algorithm, but the performance of the

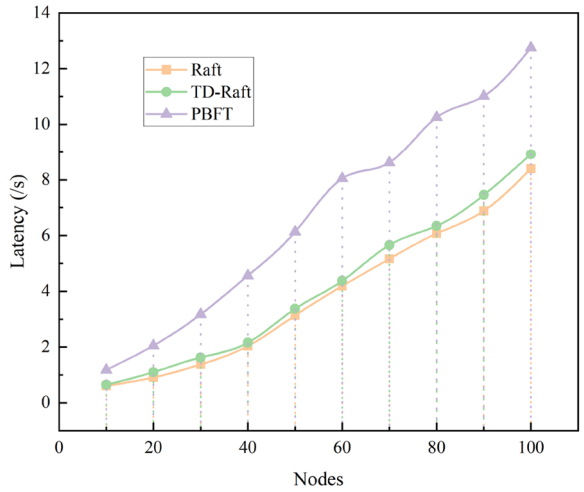

Fig. 6 Comparison of TD-Raft with Raft and PBFT in latency time

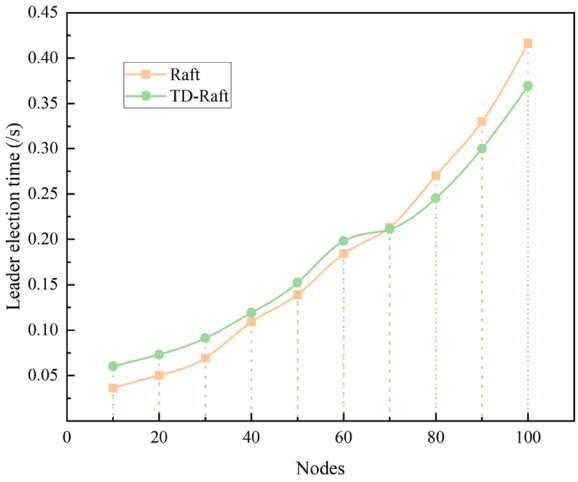

Fig. 7 Comparison of TD-Raft with Raft in leader election time

DB-Raft algorithm is higher than that of the PBFT algorithm with the nearly same security as PBFT.

In addition, to verify the security of the TD-Raft consensus algorithm proposed in this paper, we measure the security of the two consensus algorithms by testing the leader election time in the consensus process of TD-PBFT and Raft with different numbers of nodes, and the experimental results are shown in Fig. 7.

As can be seen in Fig. 7, the leader election time of Raft consensus algorithm is lower than that of our proposed TD-Raft consensus algorithm when the number of nodes is less than 70, while the leader election time of Raft algorithm starts to be gradually higher than that of TD-Raft consensus algorithm when the number of nodes exceeds 70. And with the gradual increase of the number of nodes, the leader election time of both consensus algorithms gradually increases. Therefore, we can conclude from this that although the total consensus latency of the Raft consensus algorithm is lower than that of the TD-Raft consensus algorithm, the leader election time of the Raft consensus algorithm starts to gradually exceed that of the TD-Raft consensus algorithm due to the failure of the leader election as the number of nodes increases. And because the TD-Raft consensus algorithm ensures the security of the leader election phase, the security advantage of this algorithm starts to manifest in the leader election phase, and it is more suitable for the e-commerce consortium blockchain with higher security requirements and more nodes.

Discussion

It can be learnt from the results of simulation experiments that the e-commerce transaction model based on the lightweight paillier encryption proposed in this paper reduces the consumption time of encryption and decryption drastically. The performance of TD-Raft consensus algorithm proposed in this paper is much higher than the PBFT consensus algorithm, and is slightly lower than that of the Raft consensus algorithm. Compared with the Raft consensus algorithm, the anti-Byzantine ability of TD-Raft is substantially improved, so it can basically meet the transaction efficiency requirements of MEC enabled e-commerce consortium blockchain. However, the lightweight paillier encryption algorithm in this paper is still less efficient and requires a more efficient and secure privacy protection scheme to guarantee the security and privacy of MEC e-commerce transactions. In addition, although TD-Raft consensus algorithm can resist the attack of Byzantine nodes, it consumes a long time in the leader election phase due to splitting and recovering the hash secret key. In order to meet the high throughput demand of e-commerce, consensus algorithms which are comprehensive are also needed to guarantee the efficiency and security of the system.

Conclusion

In this paper, to solve the problem of privacy leakage and low consensus algorithm security of the MEC enabled e-commerce consortium blockchain , we propose an e-commerce transaction model based on lightweight Paillier encryption algorithm to solve the above security problems. We use the lightweight Paillier encryption algorithm to encrypt the privacy information of the transaction users before trading and verify the legality of the transaction through the nature of homomorphic encryption. The security and privacy of the MEC enabled e-commerce transaction system is realized, ensuring the successful execution of the transaction, and preventing the leakage of the original transaction data. In addition,

in order to improve the anti-Byzantine failure ability of the Raft consensus algorithm, the shamir threshold secret sharing scheme is used to participate in the leader election phase, so as to prevent malicious nodes from becoming leader nodes through collusion attacks and ensure the security of the consensus algorithm. Through the evaluation of simulation experiments, our model is proved to be effective.

In the future, we will explore more efficient privacy protection solutions for e-commerce consortium blockchain, which can protect user privacy and data security and further improve system efficiency. At the same time, we will study ways to improve the scalability of MEC-enabled blockchain, such as blockchain sharding scheme, to further optimize the throughput of MEC-enabled e-commerce consortium blockchain.

Acknowledgements
We sincerely thank the editors and reviewers for their valuable comments on this paper.

Authors' contributions
G.L and H.W wrote the main manuscript and performed the experiments. J.W contributed significantly to the analysis and manuscript preparation. Z.L helped revise the manuscript. All authors reviewed the manuscript.

Funding
Not applicable.

Availability of data and materials
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 3 February 2024 Accepted: 11 April 2024
Published online: 09 May 2024

References

1. Jain V, Malviya B, Arya S (2021) An overview of electronic commerce (e-commerce). J Contemp Issues Bus Gov 27(3):666
2. Taher G (2021) E-commerce: advantages and limitations. Int J Acad Res Acc Financ Manag Sci 11(1):153–165
3. Taherdoost H, Madanchian M (2023) Blockchain-based e-commerce: A review on applications and challenges. Electronics 12(8):1889
4. Mishra R, Ramesh D, Kanhere SS, Edla DR (2023) Enabling efficient deduplication and secure decentralized public auditing for cloud storage: A redactable blockchain approach. ACM Trans Manag Inf Syst 14(3):1–35
5. Yu T, Lin Z, Tang Q (2018) Blockchain: The introduction and its application in financial accounting. J Corp Account Finan 29(4):37–47
6. Mao D, Hao Z, Wang F, Li H (2019) Novel automatic food trading system using consortium blockchain. Arab J Sci Eng 44:3439–3455
7. Liu Z, Li Z (2020) A blockchain-based framework of cross-border e-commerce supply chain. Int J Inf Manag 52:102059
8. Zhou Z, Wang M, Yang CN, Fu Z, Sun X, Wu QJ (2021) Blockchain-based decentralized reputation system in e-commerce environment. Futur Gener Comput Syst 124:155–167
9. Li M, Shao S, Ye Q, Xu G, Huang GQ (2020) Blockchain-enabled logistics finance execution platform for capital-constrained e-commerce retail. Robot Comput Integr Manuf 65:101962
10. Zheng P, Xu Q, Zheng Z, Zhou Z, Yan Y, Zhang H (2021) Meepo: Sharded consortium blockchain. In: 2021 IEEE 37th International Conference on Data Engineering (ICDE), IEEE, p 1847–1852
11. Zhang K, Leng S, He Y, Maharjan S, Zhang Y (2018) Mobile edge computing and networking for green and low-latency internet of things. IEEE Commun Mag 56(5):39–45. https://doi.org/10.1109/MCOM.2018.1700882
12. Xu X, Gu J, Yan H, Liu W, Qi L, Zhou X (2023) Reputation-aware supplier assessment for blockchain-enabled supply chain in industry 4.0. IEEE Trans Ind Inform 19(4):5485–5494. https://doi.org/10.1109/TII.2022.3190380
13. Zheng X, Feng W (2021) Research on practical byzantine fault tolerant consensus algorithm based on blockchain. In: Journal of Physics: Conference Series, IOP Publishing, vol 1802, p 032022
14. Ongaro D, Ousterhout J (2014) In search of an understandable consensus algorithm. In: 2014 {USENIX} Annual Technical Conference ({USENIX}{ATC} 14), Philadelphia, USA, p 305–319
15. Huang D, Ma X, Zhang S (2019) Performance analysis of the raft consensus algorithm for private blockchains. IEEE Trans Syst Man Cybern Syst 50(1):172–181
16. Liang W, Yang Y, Yang C, Hu Y, Xie S, Li KC, Cao J (2022) Pdpchain: a consortium blockchain-based privacy protection scheme for personal data. IEEE Trans Reliab 1–13. https://doi.org/10.1109/TR.2022.3190932
17. Xu X, Li H, Li Z, Zhou X (2023) Safe: Synergic data filtering for federated learning in cloud-edge computing. IEEE Trans Ind Inform 19(2):1655–1665. https://doi.org/10.1109/TII.2022.3195896
18. Dib O, Brousmiche KL, Durand A, Thea E, Hamida EB (2018) Consortium blockchains: Overview, applications and challenges. Int J Adv Telecommun 11(1):51–64
19. SHAKHMAMETYEV AA, STRELETS IA, LEBEDEV KA (2018) Strategic mechanisms for the future development of the international e-commerce market. Rev Espacios 39(27):21
20. Feng Q, He D, Zeadally S, Khan MK, Kumar N (2019) A survey on privacy protection in blockchain system. J Netw Comput Appl 126:45–58
21. Xu X, Fang Z, Qi L, Zhang X, He Q, Zhou X (2021) Tripres: Traffic flow prediction driven resource reservation for multimedia iov with edge computing. ACM Trans Multimedia Comput Commun Appl 17(2). https://doi.org/10.1145/3401979
22. Yang C, Xu X, Zhou X, Qi L (2022) Deep q network-driven task offloading for efficient multimedia data analysis in edge computing-assisted iov. ACM Trans Multimedia Comput Commun Appl 18(2s). https://doi.org/10.1145/3548687
23. Xu X, Tang S, Qi L, Zhou X, Dai F, Dou W (2023) Cnn partitioning and offloading for vehicular edge networks in web3. IEEE Commun Mag 61(8):36–42. https://doi.org/10.1109/MCOM.002.2200424
24. Qi L, Xu X, Wu X, Ni Q, Yuan Y, Zhang X (2023) Digital-twin-enabled 6g mobile network video streaming using mobile crowdsourcing. IEEE J Sel Areas Commun 41(10):3161–3174. https://doi.org/10.1109/JSAC.2023.3310077
25. Wang Y, Li S, Xu L, Xu L (2021) Improved raft consensus algorithm in high real-time and highly adversarial environment. In: Web Information Systems and Applications: 18th International Conference, WISA 2021, Kaifeng, China, September 24–26, 2021, Proceedings 18, Springer, p 718–726
26. Copeland C, Zhong H (2016) Tangaroa: a byzantine fault tolerant raft[Online]. Available: http://www.scs.stanford.edu/14au-cs244b/labs/projects/copeland_zhong.pdf
27. Fu W, Wei X, Tong S (2021) An improved blockchain consensus algorithm based on raft. Arab J Sci Eng 46(9):8137–8149
28. Tian S, Liu Y, Zhang Y, Zhao Y (2021) A byzantine fault-tolerant raft algorithm combined with Schnorr signature. In: 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM), IEEE, p 1–5
29. Azad MA, Bag S, Hao F (2018) PrivBox: Verifiable decentralized reputation system for online marketplaces. Futur Gener Comput Syst 89:44–57
30. Ghadamyari M, Samet S (2019) Privacy-preserving statistical analysis of health data using paillier homomorphic encryption and permissioned blockchain. In: 2019 IEEE International Conference on Big Data (Big Data), IEEE, p 5474–5479

31. Zhou X, Ye X, Wang KIK, Liang W, Nair NKC, Shimizu S, Yan Z, Jin Q (2023) Hierarchical federated learning with social context clustering-based participant selection for internet of medical things applications. IEEE Trans Comput Soc Syst 10(4):1742–1751. https://doi.org/10.1109/TCSS.2023.3259431
32. Ma Z, Wang L, Zhao W (2020) Blockchain-driven trusted data sharing with privacy protection in IoT sensor network. IEEE Sensors J 21(22):25472–25479
33. Xu G, Zhang J, Wang L (2022) An edge computing data privacy-preserving scheme based on blockchain and homomorphic encryption. In: 2022 International Conference on Blockchain Technology and Information Security (ICBCTIS), IEEE, p 156–159
34. Zhou X, Zheng X, Cui X, Shi J, Liang W, Yan Z, Yang LT, Shimizu S, Wang KIK (2023) Digital twin enhanced federated reinforcement learning with lightweight knowledge distillation in mobile networks. IEEE J Sel Areas Commun 41(10):3191–3211. https://doi.org/10.1109/JSAC.2023.3310046
35. Zhou X, Liang W, Wang KIK, Yan Z, Yang LT, Wei W, Ma J, Jin Q (2023) Decentralized p2p federated learning for privacy-preserving and resilient mobile robotic systems. IEEE Wirel Commun 30(2):82–89. https://doi.org/10.1109/MWC.004.2200381
36. Mishra R, Ramesh D, Edla DR, Trivedi MC (2022) Blockchain assisted privacy-preserving public auditable model for cloud environment with efficient user revocation. Clust Comput 25(5):3103–3127

Publisher's Note