IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 18, 2019, accepted July 31, 2019, date of publication August 8, 2019, date of current version August 21, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2933860

Auditable Protocols for Fair Payment and Physical Asset Delivery Based on Smart Contracts

SHANGPING WANG[1], XIXI TANG[1], YALING ZHANG[2], AND JUANJUAN CHEN[1]
[1]School of Science, Xi'an University of Technology, Xi'an 710048, China
[2]School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

Corresponding author: Xixi Tang (xxtang1229@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61572019, in part by the Key Research and Development Program of Shaanxi under Grant 2019GY-028, and in part by the Start-up Fund for PhD Teachers in the Xi'an University of Technology, under Grant 256081502.

ABSTRACT With the rapid development of electronic information technology, online transaction will gradually surpass traditional market transaction, among which online payment and asset delivery become the focus of attention. But in fact, due to the incomplete third-party payment mechanism and the intrusion risk of various charging Trojan, it is easy to cause a trust crisis. The existing centralized framework often leads to information asymmetry between the two parties. Therefore, how to realize the fairness of payment and the auditability of assets in the distributed system is a challenging problem. The emerging blockchain technology provides a new method with its openness, transparency and verifiability. Existing researches do not provide a complete shopping model for consumers, most of which focuses on payments or only on asset delivery. In this paper, we propose an auditable fair payment and physical asset delivery protocol based on smart contracts. Three types of smart contracts are designed to achieve reliable and fair payment among merchants, consumers and logistics companies. The traceability and auditability of blockchain provide an effective method to audit assets and data sharing in the whole transportation. In view of the phenomenon of goods being switched, the way of "pre-verification" is added. In order to prevent the illegal elements to fake pickup code, induce consumers to conduct illegal operations, cause property loss, in our system the pickup codes are generated by consumers to reduce the risk of fraud. In addition, our plan designs a complete return process for the first time, providing better service experience and higher efficiency for consumers. Finally, all the contracts involved in the scheme are implemented and deployed on the ethereum test network. The results of security analysis and evaluation showed that our scheme was improved in cost, with high security and availability.

INDEX TERMS Online transactions, blockchain, assets audit, smart contract.

I. INTRODUCTION

With the rapid development of e-commerce and the Internet, mobile devices and various software resources are widely used, and their diversified functions have great influence on work, life, entertainment and other aspects. Among them, the most worthy of public attention is online shopping. People can buy the goods they want and enjoy the convenience of door-to-door delivery without leaving home. Existing network shopping mode, mainly involving online payment transactions and asset transport services. Online payments are made digitally, based on open Internet platforms, and therefore rely on a central authority as a trusted intermediary to manage the flow of money and store transaction records. However, this payment method managed by a single trusted authority is prone to trust crisis and payment fraud, making both parties lose money. In addition, during the transportation of goods, goods are transferred, lost, damaged and other situations often occur, which is easy to produce disputes or even legal disputes among the three parties. Therefore, it is necessary to publicly certify the consistency and delivery of goods during transportation, to provide auditability, to ensure that the economic interests of the parties involved in the transaction are not impaired and to maintain the reputation of the individual.

Existing delivery systems rely on signed documents as proof of receipt of goods or on hand-held electronic devices to obtain consumer signatures. Therefore, the shipper needs

The associate editor coordinating the review of this manuscript and approving it for publication was Rashid Mehmood.

to verify the validity of the signature and ship the goods to the correct recipient. But for businesses and consumers, the transportation service is a separate system that can be fraudulent. In addition, the single consideration of payment or delivery is no longer enough to meet the needs of consumers, and it is more effective to provide an overall shopping model from online ordering to asset transportation, and finally to payment, or return.

Based on the above analysis, in order to solve the problem of trust and single point of failure in a centralized framework, payment needs to be executed in the trusted execution environment(TEE) and provide an open and transparent distributed storage management. In the TEE, there is no need to worry about transaction data being tampered with, and there is better accountability based on historical transaction data. So we'll turn our attention to bitcoin, whose underlying technology is a decentralized distributed storage database called blockchain [1]. The blockchain consists of blocks containing transactions, which are linked in chronological order and reversed into a linked list. All nodes in the network can verify the transactions in the block and save them in the local storage memory, creating an open and transparent and undisputed Ledger [2], [3]. The application of blockchain can provide a good solution to the problem of over-centralization and trust of transaction data.

Bitcoin provides a kind of peer to peer payment, in which both sides of online transactions can pay directly to each other without the need for trusted authorities or third-party payment institutions [4]–[6]. However, the blockchain of electronic currency led by bitcoin still has many limitations. They tend to focus on payments, and while bitcoin offers fields for non-transactional data, it comes at the expense of memory. If the output script contains other information to verify, the miner is required to perform these operations, which undoubtedly increases transaction fees and imposes burden on individuals or enterprises. We should not only complete the online payment of goods, but also realize the management and certification of goods. In the existing logistics system, the transportation information of express delivery is updated to the network by logistics companies, and merchants and consumers check it. Such one-way notifications do not really share information and have the potential to falsify data. For example, if a product doesn't arrive at its destination, the logistics company's employees update the receipt information, but the consumer doesn't receive the product. Therefore, the delivery of goods requires more favourable evidence. To implement these features, we consider smart contracts, a turing-complete language that can be combined with blockchain to provide greater functionality. Shopping network including all kinds of e-commerce has a huge potential market, people will consider from many aspects when buying all kinds of goods online, especially some luxury goods should be more cautious. So it makes sense for us to explore trustworthy and auditable solutions that can provide a better shopping experience for online users.

In order to ensure the safety of online payment and the auditability of physical assets, our scheme does not need a trusted central authority and reduces certain artificial risks. Transaction data and asset management are recorded in a distributed network, allowing both parties to better implement accountability based on transaction history. However, in order to realize such a complete system, how to realize the fairness of payment, reduce the monopoly of logistics companies on transportation information during transportation, and avoid consumers being cheated, still need to be studied in depth. In this article, we set up a shopping system that merchants, consumers and logistics companies trust so that all members of the network can properly exercise their rights and defend their interests.

The main contributions of this article can be described as follows:

(1)We achieve the flow of funds through the escrow function of the contract [7], [8], reduce costs for participants and protect privacy. Three main types of smart contracts are used to implement product management, and to verify the identity and rights of each participant, ensuring that only callers who meet the conditions set in the contract can perform relevant functions. The whole process of commodity transportation is recorded in the block, and its hard-to-tamper characteristic provides guarantee for commodity audit.

(2)In the express delivery period, there are phenomena such as the courier unpacking and changing the package, which will bring losses to others. Therefore, we use the method of pre-verification to avoid the phenomena. After the logistics company ensures that the goods are consistent with the consumer orders before the transportation, the verification information is uploaded to the blockchain, the accountability system of blockchain makes logistics companies cannot deny their mistakes. In addition, criminals send fake pickup code messages to consumers, and then the goods lost on the grounds of door-to-door compensation, induce consumers to click on illegal links to transfer operations. To this end, we transfer the right to generate pickup codes from logistics companies to consumers to avoid such incidents.

(3)For the first time, we have designed a complete return and refund process to ensure consumers' different needs. During disputes, regulators are introduced to conduct audits outside the chain and impose penalties.

(4)Remix is used to write smart contracts, which are deployed on the Ethereum test network Rinkeby. In addition, the security vulnerability of smart contract is analyzed.

The rest of this paper is organized as follows: the Section II introduces relevant work, the Section III shows some technologies used in the paper, the Section IV proposes a specific framework. The details of the scheme are expanded in the Section V, the Section VI conducts security analysis and testing. Finally, the conclusion is presented.

II. RELATED WORK

In real life, the usual online shopping model is shown in Fig. 1. It consists of four main players, namely merchants,

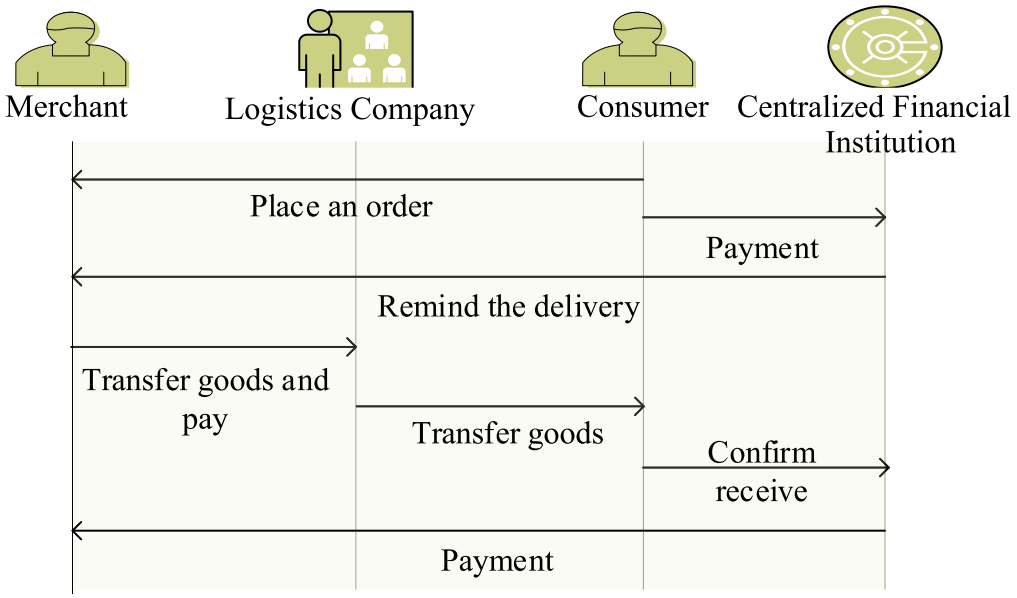

FIGURE 1. Online shopping transaction diagram.

consumers, centralized financial institutions(taking banks as an example) and logistics companies. Here, the payment between merchants and consumers is taken as an example. At the beginning of the transaction, consumers first place orders and payments to the bank, and the bank notifies the merchant to deliver the goods. Then the merchant sends the goods to the consumer through the logistics company. After the consumer confirms the integrity of the goods, the consumer sends a confirmation message to the bank, and the bank makes the payment to the merchant. The payment is not directly transferred to the merchant's account, but relies on a centralized financial institution to complete the transaction. Therefore, the trust of the centralized financial institution becomes the focus of online payment. In order to ensure the fairness of both parties, it is important to study the credible transaction environment. In addition, there have been many problems in the transportation of commodities in recent years, which are mainly reflected in the fact that it is difficult for consumers to protect their rights. If merchants and logistics companies refuse to take responsibility, it is easy to cause losses to consumers. Decentralized trading methods that make transaction data and participants' information publicly available can also reduce transaction fees incurred by a centralized financial institution. Blockchain can also automate the processing of exception records, making real-time auditing possible. Below we will introduce the blockchain technology and previous research work and the motivation of this paper.

A. BLOCKCHAIN TECHNOLOGY

Since the birth of bitcoin in 2009, the decentralization [9], trustworthiness, open source, collective maintenance and other characteristics of blockchain have attracted the attention of a large number of researchers at home and abroad. Although blockchain comes into being with bitcoin, its characteristics of anonymity [10], tamper-proof, auditability, verifiability and so on enable it to exist independently and be applied in fields other than cryptocurrency, such as health care, supply chain, Internet of things [11]–[13], cloud storage, artificial intelligence (AI) [14] and so on. In 2016, rating giant Moddy published a report [15], which discussed 120 blockchain projects involving enterprises and governments.

In recent years, the combination of blockchain technology and smart contract technology has provided solutions for various fields. Nick Szabo, a prolific interdisciplinary legal scholar in 1995, proposed that "A smart contract is a set of promises, specified in digital form, including protocols within which the parties perform on these promises". Most now understand that smart contracts can be automatically executed based on pre-set conditions. I think Ethereum is one of the best platforms for deploying smart contracts [16]. So far, there are more than 2,400 decentralized applications on Ethereum [17], [18].

In addition, to reach a consensus in distributed networks requires the nodes in the network to use the same consensus algorithm. For example, proof of work in bitcoin, in the process of generating blocks, a random number is searched to make SHA256 hash with the information of blocks, which satisfies a difficulty target value varying with the network. This method wastes a lot of energy and is extremely inefficient. However, the use of currency age instead of hash calculation in the proof of stake is easy to create a commercial monopoly [19], [20]. The PBFT [21], [22] algorithm used by HyperLedger is a state machine replication algorithm. State machines replicate at different nodes of the distributed

system, each state machine keeps the state of the service, and additional copies do not improve reliability beyond reducing performance. Ripple [23] consensus uses a collective trusted subnetwork, which the nodes participating in the voting have been aware of beforehand, thus the efficiency is higher, but this also determines the low degree of decentralization of the algorithm [24], [25]. Paxos [26] is a distributed consistency algorithm based on messaging and is the first proven algorithm. Because this algorithm is relatively difficult to understand and implement, a simpler Raft [27] consensus algorithm appears. Raft is a strong consensus protocol reached without a Byzantine failure. Pow, Pos and Ripple are used in permissionless chains, PBFT is available for consortium blockchains, Paxos and Raft are usually used in trusted environments, and mainly in private chains.

B. PREVIOUS RESEARCH WORK AND MOTIVATION

The proposed blockchain technology provides fresh blood for all walks of life, and the application research on blockchain keeps emerging, becoming one of the hot technologies nowadays. For example, in the aspect of the Internet of Things [4], [28]–[30], the blockchain is a scalable and trusted peer-to-peer model. It can transparently operate and distribute data securely, and can provide a good solution for solving the update and maintenance. Full protection of patient privacy in health care [31], [32], as well as supply chain, cloud storage and other aspects [33]–[35].

He *et al*. [36] discussed the incentive mechanism in the distributed network and proposed that users initially reserve a certain amount of money in the transaction, and users who honestly abide by the agreement can get the returned deposit. Zhao *et al*. [37] proposed a fair institutional scheme between publishers and subscribers based on blockchain technology. Subscribers specify topics of interest by submitting certain deposits, and when the subscriber decrypt the encrypted content uploaded by the publisher to the blockchain, the publisher obtains a mortgage deposit. Other fair payment schemes can also be seen in literature [32], [38]–[40]. This payment method is implemented on the blockchain based on cryptocurrency, and a large amount of verification and matching work is implemented by the miners in the network. Users need to monitor the network at all times, and the authentication of users falls into one category.

In terms of asset auditing, Toyoda *et al*. proposed a new product ownership management system in [41]. Because of the problem that the anti-counterfeit labels are easily copied in the post-supply chain, the ownership of the products is declared during the transportation process, ensuring that each handover can verify the previous handover and prevent the secondary sale of counterfeit goods. Altawy *et al*. [42] proposed an anonymous delivery system scheme for physical assets. Consumers upload their real addresses to the blockchain in an encrypted manner, and interact with the transporter to provide the next delivery point during transportation. This approach increases computing costs and complexity for consumers, and we believe that the best interests of consumers are the key to providing better services. Salah and Hasan [43] proposed a solution of physical asset delivery based on blockchain, in which funds are entrusted to the contract, and the transport company updates the transfer of assets in the contract, which can be verified by the seller and buyer at any time. At the same time, the seller provides two keys to the express company and the buyer respectively for the asset handover verification. Although it is pointed out in the literature that the key and the asset are transported together, the relationship between the key and the asset is not stated. The key and the asset are only transported as proof, and there is no guarantee for the authenticity of the asset itself, Hasan and Salah [44] changed the key from two to one, which is only given to the transportation company. When the asset arrives at one party, it is verified with the key hash already in the contract, but the relationship with the asset is not pointed out. In addition, none of the previous schemes have discussed the situation of consumer returns in detail.

Based on the above problems, in our scheme, for fair payment and asset audit, we use smart contract to increase the automation of the scheme, and put the fund into the contract to ensure the fund flows in the preset direction. By storing commodity attributes and transport information on the blockchain, the verifiable and auditable of blockchain can supervise and verify the behaviors in the network and reduce the risk of cheating by merchants and logistics companies. In addition, we suggest that the method of pre-verification should be adopted to deal with the behavior of couriers to change parcels. Transfer the right to generate pickup code from the logistics company to the hands of consumers, increase consumers' control over assets, so that consumers feel more secure. Aiming at product quality problems and consumers' personal choices, we have designed a complete rejection and return framework for the first time, providing a perfect shopping mode for consumers.

III. PRELIMINARIES

A. BLOCKCHAIN TECHNOLOGY AND ETHEREUM

The blockchain itself can act like a log, Transaction records in the network are processed into time-stamped blocks, and each block can be identified with a unique hash and stored in the next block, thus forming a chain structure of reverse connection. Currently, the most popular one is ethereum blockchain, and our solution also adopts ethereum blockchain. Compared to the Bitcoin blockchain, the Ethereum blockchain contains more information and functions, such as greater system throughput and smaller transaction confirmation intervals. The main thing is that it supports smart contracts, where anyone can deploy different applications, and we can think of ethereum as a programmable blockchain.

Ethereum includes two types of accounts: externally owned accounts (EOAs) and contract accounts. Externally owned accounts can create transactions with private key signatures to send messages to other external accounts and contract accounts, while contract accounts are generated by

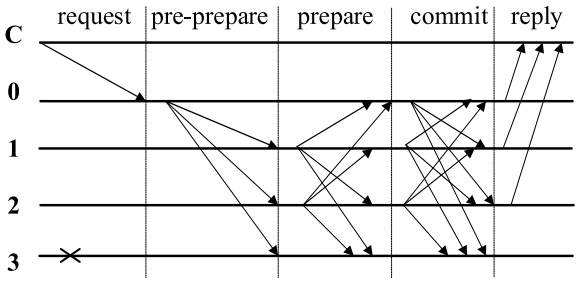

FIGURE 2. PBFT consensus algorithm.

contract codes and cannot be executed autonomously, and various operations in the contract are executed after being triggered by the transaction. Ethereum contains three different Merkle tries, a status tree, a transaction tree, and a receipt tree, making it easy for all the light nodes to create and validate transactions.

B. PBFT CONSENSUS ALGORITHM

In a decentralized network, the same consensus algorithm must be used if global nodes are to agree. Our solution is implemented with a permissioned chain and relies on high decentralization. According to the discussion on consensus algorithms, our scheme selects Practical Byzantine Fault Tolerance consensus algorithms. Fig. 2 shows the specific process of reaching consensus:

PBFT consensus algorithm has the following five states [45]:

(1) request: the client sends a request to the master node.

(2) pre-prepared: The master node receives the request from the client and needs to verify whether the signature of the client request message is correct. If correct, a pre-prepare message is broadcast to other replica nodes.

(3) prepared: the replica node receives the pre-prepare message from the master node and carries out verification. If correct, the replica node sends a prepare message to other nodes, including the primary.

(4) committed: the master node and the replica node receive the prepare message and carry out the verification. If the replica node receives a $2f + 1$ verified prepare message indicating that most nodes in the network have received the consent message, a commit message is sent to other nodes, including the primary.

(5) reply: the master node and the replica node receive the commit message. If the replica node receives $2f + 1$ verified commit messages, it indicates that most nodes in the current network have reached a consensus.

If the master node drops calls or does not broadcast the client request, the client sets a timeout mechanism that broadcasts the request message to all replica nodes. The replica node detects that the master node has committed a crime or is offline and initiates the view rotation protocol.

View change: the node receives $2f + 1$ view change messages with the same block number.

Table. 1 shows the details of the sent message.

TABLE 1. message content.

pre-prepare	prepare	commit
<pre-prepare,v, n, d, m>	<prepare,v, n, d, i>	<commit,$v, n, D(m), i$>

Where v represents the number of the current view, n represents the number of the current request, m represents the content of the message, d or $D(m)$ represents the summary of the content of the message, and i represents the number of the node.

C. SMART CONTRACTS

In the beginning, smart contracts did not catch on due to the lack of a credible execution environment. Since the blockchain technology was proposed, people gradually paid attention to it [46].

1) MODIFIERS

you can check whether the sender of the message is a caller of the function based on the Ethernet address before the function executes, or some other preconditions. The modifier parameter can be an arbitrary expression, and in the corresponding context, the symbols introduced in all functions are visible in the modifier, but the symbols introduced in the modifier are not visible in the function.

2) EVENTS

when an event is triggered, the event and its parameters are stored in the ethereum log. Each event has a maximum of three parameters that can use the indexed keyword to set the index, after setting the index, you can find the log according to the parameters, the unindexed parameters will be stored as part of the log.

3) VARIABLE

variables are used to store information that may change with the transaction. They are mainly used to store the ethereum address of the participating entity, the key hash to be compared during key verification, and the properties and states of assets in the contract.

The creation and invocation of the contract are shown in Fig. 3. First write a smart contract, written in an easy-to-read high-level language, the Ethereum virtual machine (EVM) will automatically compile into bytecode, then package the bytecode into the data field in the transaction and upload it to the blockchain network. After the node in the network receives the transaction, it will check whether the transaction is valid, the format is correct, and calculate the maximum possible transaction fee. The account of the initiator of the contract must have a balance greater than or equal to the transaction fee, otherwise the node will not forward. Finally, the transaction is put into the block, and a consensus is reached on the block in the whole network, which is connected to the local blockchain. Other parties to the contract can call functions in the contract or get parameter

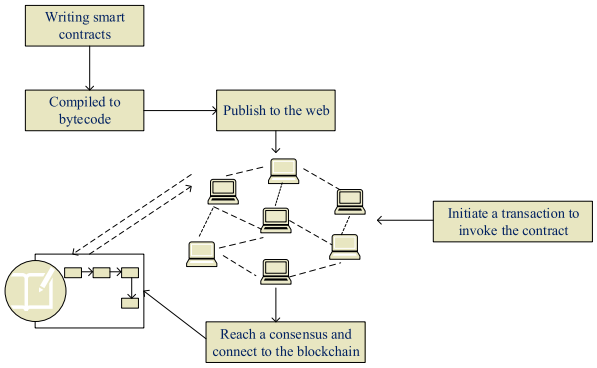

FIGURE 3. Smart contract creation and invocation.

value information through the returned contract account and ABI. When writing smart contracts, be sure to double-check to make sure your code is correct, because contracts cannot be changed once deployed. In order to prevent the cost of gas caused by code errors in ethereum, specify the gas upper limit when creating a transaction to reduce the loss caused by the contract creator due to code loop.

The smart contract periodically checks the state of the automaton, iterates through the state machine contained in each contract, the transaction, and the trigger conditions, and pushes the satisfied transaction to the queue to be verified. Wait until the next round of consensus, spread to each node, after the completion of the execution, the state machine judge the contract state. When all transactions in a contract are executed sequentially, the state machine marks the state of the contract as completed, or it marks it as not completed and waits for the next round of processing. These processes are automated by the smart contract system built into the underlying blockchain.

D. IPFS

InterPlanetary File System(IPFS) [47], [48] is a point-to-point distributed hypermedia distribution protocol. It is a permanent, decentralized method to save and share files. The principle is to replace the domain-based address with the content-based address, the so-called content addressing is to use the hash value of the file to find the file without knowing where the file is stored. IPFS integrates the best distributed system ideas in recent years. When uploading files, file contents are stored in chunks based on file size. Each node will maintain a DHTs(Distributed Hash Tables) [49] when need to download the file by file hash value of the request, the system will be based on the hash from the nearest node to composite file, it will also be verified. Fig. 4 shows the IPFS storage process.

IV. SYSTEM FRAMEWORK

Before describing our solution, first consider some key issues about online shopping. Before purchasing commodities, consumers should be able to correctly identify merchants and products. For unofficial websites and products, they should

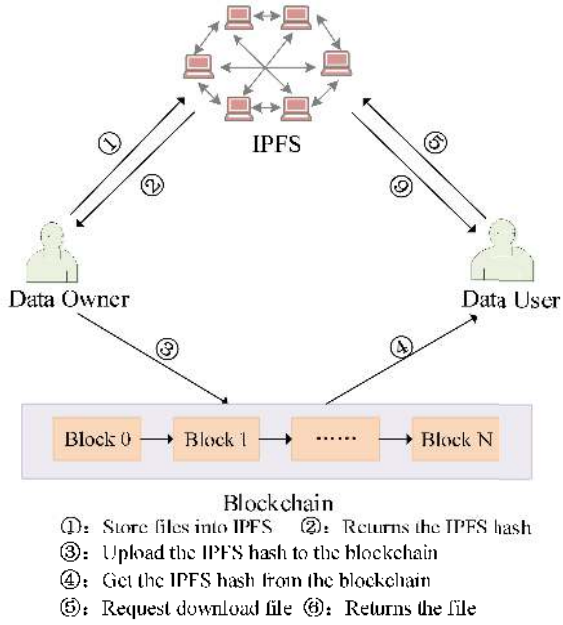

FIGURE 4. IPFS storage system.

carefully browse and purchase. When purchasing expensive products, they can ask merchants to provide official product identification certificates. In order to ensure that consumers can buy products with confidence, consumers can ask merchants to upload product attributes and other information to the blockchain before making payment. Due to the size limit of the smart contract, the merchants upload the IPFS hash of the product content to the blockchain. This article will not elaborate on how IPFS is generated. In the network we set up, merchants and logistics companies need to provide business certificates before they can join, and consumers can join unconditionally, as long as they provide an ethereum address. Therefore, we only protect the anonymity of consumers, which is also consistent with the reality.

As the online shopping consumers have certain requirements on time, the delivery time needs to be set in the scheme. Mainly considering the distance factor, suppose the specified delivery time range is $[T_1, T_2]$, where $T_2 \geq T_1$. According to the distance, the time range for logistics companies to transport commodities is set as follows:

$$t = \begin{cases} t_1 & T_1 \leq t_1 \leq \dfrac{2T_1 + T_2}{3} \quad \text{Short distance} \\ t_2 & T_1 \leq t_2 \leq \dfrac{T_1 + 2T_2}{3} \quad \text{Medium distance} \\ t_3 & T_1 \leq t_3 \leq T_2 \quad \text{Long distance} \end{cases}$$

In addition, consumers can return goods within t_0 days of receipt, and cannot return goods after this period.

The main participants of our programmer are as follows:

Merchants: The merchant mainly serves as the seller of the products, provide the products to be sold in the operating stores, creates smart contracts for each sold product, records the attributes and purchase information of the products, and mortgages certain funds. This is to ensure that the nodes in

the network can do the right thing. If there is no accident, the merchant will get the payment and the mortgage capital after $t_0 + 2t$ days.

Logistics Company: Logistics companies are responsible for shipping goods, creating smart contracts for each item shipped, recording specific shipping routes and times, updating the location and status of the goods. Similarly, in order to ensure the credibility of the logistics company, a certain amount of capital should also be mortgaged into the contract, and the logistics company will get the corresponding transportation fee and deposit after confirming the receipt of goods. According to most express cases in real life, we assume that the transportation route is city A branch - city A headquarters - city B headquarters - city B branch.

Consumers: Consumers play the role of purchasing goods. They can choose the right goods according to their own needs, and they can communicate with merchants properly when purchasing so as to fully understand the functions of the products. In the successful purchase transaction, we hope to minimize the participation of consumers, so as to provide better services and customer experience.

Regulator: Regulators supervise the entire network. Authenticate the identity of the nodes that are added to the network, verify the disputes arising in the process of the transaction out of the chain, and punish dishonest parties to some extent. Therefore, regulators must be trusted by all.

Smart Contract Attestation Authority (SCAA): The SCAA certifies that all contracts are subject to an agreed treaty to ensure the normal operation of the network.

In our scheme, the value transfer and information exchange between participants are mainly carried out through the establishment of three types of smart contracts. The functions contained in the contracts are respectively called by specific ethereum addresses to realize different functions, as shown in Fig. 5. An IC contract represents an identity contract, created by a regulator that implements the function of authenticating the identity of nodes that join the network. The MC represents merchant and consumer contracts, enabling asset information management and payment functions. ML contract is a contract between a merchant and a logistics company, which realizes the function of asset delivery and payment. The specific function information is as follows:

A. IDENTITY CONTRACT (IC)

The contract mainly contains two fields, the user's ethereum address and identity type. There are three types of identity: merchants, logistics companies and consumers. We use M to indicate that the owner of the ethereum address is a merchant, L to indicate that the owner of the ethereum address is a logistics company, and C to indicate the consumer. The nodes newly added to the network are authenticated by the regulator and entered into the contract. The contract is stored in the blockchain, and anyone can view or prove their identity. If someone wants to leave the network, the regulator will call the contract and destroy it.

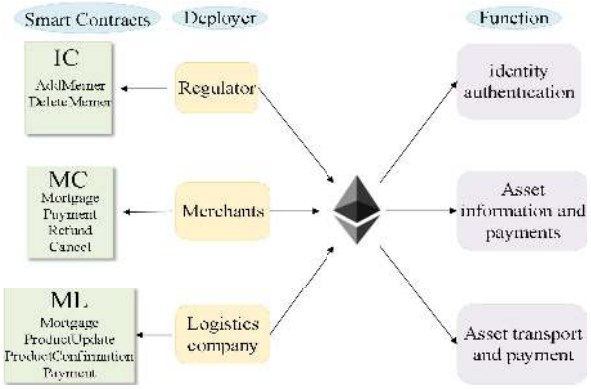

FIGURE 5. Three smart contracts.

Algorithm 1 AddMember to Internet

```
Input: Regulators,Ethereum address,type
If msg.sender==Regulators
    If keccak256(type)==keccak256(M)
        Member is a merchant
    Else if keccak256(type)==keccak256(L)
        Member is a logisticcompany
    Else if keccak256(type)==keccak256(C)
        Member is a consumer
End
```

Algorithm1 is an add member operation, the regulator enters the ethereum address and identity type of the member. The member type is public and anyone can view the type represented by any ethereum address.

B. MERCHANT AND CONSUMER CONTRACTS (MC)

This contract is mainly used to solve the problem of product management of merchants and consumers. The main functions included in the contract are mortgage function, payment function, refund function and cancellation function, including mortgage function mainly used to deposit the deposit of merchants and consumers, the amount of funds deposited is determined by the specific commodity. The payment function is used to confirm the payment of funds to the merchant after receipt of the goods or to return the respective mortgage after confirming the return, the refund function returns funds to consumers, the cancellation function can be used to cancel a transaction before it is shipped.

Algorithm 2 Mortgage Funds Into the Contract

```
Input: Merchant,Consumer,deposit
If(msg.sender==Merchant||msg.sender
==Consumer
  &&msg.value==deposit)
        Mortgage success;
End
```

Algorithm2 is a mortgage function called by both the merchant and the consumer. The mortgage funds need to meet

the pre-initialization amount. With this function, you can host funds without fear of manipulation.

Algorithm 3 Payment Function

```
Input: Merchant,k
If(msg.sender==Merchant&&keccak256(k)
==(h1)&& time >t0 + 2t days)
{
    Transfer funds to the merchants;
    The balance is returned to the consumer;
}
```

Algorithm3 is the payment function. If the consumer successfully receives the goods and does not return the goods within t_0 days, the MC contract will send the payment for goods and all the funds pledged by the merchant to the merchant's account after t_0+2t days, and return the remaining balance of the contract to the consumer.

Algorithm 4 Refund Function

```
Input: Consumer,k
if(msg.sender==Consumer&&keccak256(k)
==(h2)&&refuse product)
{
    Transfer deposits to merchants and consumers;
}
else if(msg.sender==Consumer&&keccak256(k)
==(h2)&&return after receipt)
{
    Transfer deposit and shipping fee to
    consumers;
    The balance is returned to the merchant;
}
End
```

Algorithm4 is the refund function. If the consumer refuses to accept the goods, the contract returns each person's mortgage, but if the consumer returns the goods within t_0 days of receipt, the consumer receives an additional transportation fee. Here, k represents the hash primitive to be verified, and the specific content will be explained in the specific scheme.

Algorithm 5 Cancel This Transaction

```
Input: Merchant,Consumer
If(msg.sender==Merchant||msg.sender==
Consumer&& Product not issued)
    The transaction has been cancelled;
else
    The transaction cannot be cancelled
End
```

Algorithm5 is a cancellation function that can only happen when a merchant or consumer cancels before shipping, and if the logistics company deploys the ML contract and the merchant updates the ML contract address to the blockchain, no one can cancel the transaction.

C. MERCHANT AND LOGISTICS COMPANY CONTRACTS (ML)

This contract is mainly used for logistics companies to manage goods and record receiving information during transportation, including mortgage function, product update function, product confirmation function and payment function. The mortgage function is used to deposit the deposit in the merchant and the logistics company, and the product update function is used to update the current location of the product, and also serves as the handover proof between the two transportation points. The product confirmation function is used to confirm whether the product is received or rejected, and the payment function is used to pay merchants and logistics companies based on the received results. The mortgage function in ML is the same as the mortgage function in the MC contract, which is not described here.

Algorithm 6 Update Product Information

```
Input: LogisticCompany,status
If(msg.sender==LogisticCompany)
{
    If(status==true)
    {
        The current location of the product
        is msg.sender;
        The product is intact;
    }
    Else
        Product is damaged;
}
End
```

Algorithm6 is a product update function. In the process of transporting products, when the transfer point receives the products, the logistics company must confirm that the goods are intact and upload the information to the blockchain.

Algorithm 7 Product Confirmation Information

```
Input: LogisticCompany,Consumer,k,reason
If(msg.sender==LogisticCompany&&keccak256
(k)
    ==(h1)&&status==true;)
    The product has been received by the
    consumer;
Else if(msg.sender==Consumer)
    Product rejection reason;
End
```

Algorithm7 is used to confirm the product's acceptance, and if the acceptance is successful, the logistics company updates the information to confirm the acceptance. If the product is rejected, the consumer uploads the rejection information and states the specific reason.

Algorithm8 is the payment function. When consumers confirm receipt of the goods, the logistics company gets k_1, which

Algorithm 8 Payment Function

```
Input: Merchant,LogisticCompany,k
If(msg.sender==LogisticCompany&&keccak256(k)
   ==(h1)&&time<=t days)
{
    Transfer of mortgage and single transportation fee
    to LogisticsCompany;
    The balance is returned to the Merchant;
}
Else if(msg.sender==LogisticCompany&&
   keccak256(k) ==(h2)&&time<=2*t days)
{
    Transfer of mortgage and double transportation fee
    to LogisticsCompanies;
    The balance is returned to the Merchant;
}
End
```

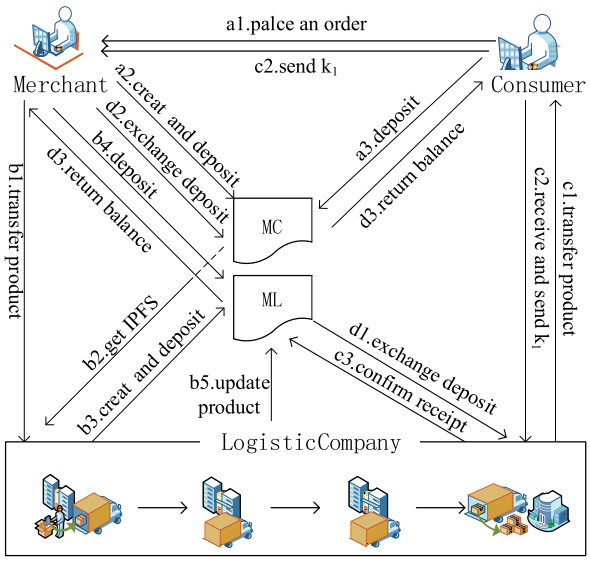

FIGURE 6. Online shopping frame diagram.

can be exchanged for the deposit and the single shipping fee. Considering that most of the commodities sold by merchants now have freight insurance, when consumers refuse to accept them, logistics companies will get k_2 when they return the commodities to merchants, at which time they will exchange the deposit and double the transportation fee.

V. SCHEME DETAILS

Specific steps of the program:

1) **Place an order:** According to their own needs, consumers choose the right products on the merchants' websites. After confirming the purchase, consumers can randomly select a series of Numbers, define it as k_1(we hope to have built-in software to solve this problem with one key), and then get the hash of k_1, which is denoted as $h_1 = keccak256(k_1)$. The obtained hash value h_1 is sent to the merchant together with the product information(see a1 in Fig. 6). k_1 is used as a pickup code here. What is different from the past is that the pickup code is decided by the consumer instead of the logistics companies.
2) **Order receiving:** The merchant receives the order information from the consumer and creates the MC contract for the product according to the template of the MC contract. The product attribute information is uploaded to the IPFS network to obtain the IPFS hash, and the attribute information includes the product appearance picture, product parameters, manufacturer, price and the inspection certificate of the authority. At the same time, the merchant will also select a random value, defined as k_2, and obtain the hash value $h_2 = keccak256(k_2)$. k_2 is mainly used for consumers to return goods, which will be explained later. Then, the merchant will deposit the ethereum address of consumers, IPFS hash, hash values h_1 and h_2, and a return period t_0 days into the contract. The merchant deploys the MC contract and sends the contract address to the consumer and the logistics company that will ship the product. Consumers can obtain IPFS hash and check whether the properties of commodities are consistent with their expectations. The merchant and the consumer call the mortgage function in the MC contract, the consumer deposits the ether equivalent of the price of the product, and then each party deposits a fine, which will be punished by the regulator if dishonest behavior occurs. In addition, because freight insurance is included, the merchant should also store an extra transportation fee, in case the consumer may return the goods within t_0 days(see a2 and a3 in Fig. 6). Both the merchant and the consumer have the right to call the cancellation function to cancel the transaction before the product is shipped, and no one can cancel the transaction if the ML contract has been created. Since the ML contract is created, it means that the logistics company has accepted the order and started shipping.
3) **Delivery:** When the logistics company receives the product(see b1 in Fig. 6), needs to carry out pre-verification, it first obtains the IPFS hash according to the MC contract address, and compare with real products to prevent courier from replacing products during delivery(see b2 in Fig. 6). Then the logistics company plans the route to be delivered according to the delivery address, creates ML contract, stores the ethereum address of merchants and consumers, hashes h_1 and h_2, and delivery time t days. Among them, the logistics company's ethereum address is a total of four, representing four transportation points, the first one is used to deploy the contract and the funds are managed, the last one is responsible for the handover with the consumer, and the middle is used as the transshipment point.

Finally, the contract address is sent to the merchant and the consumer, and the merchant initiates a transaction with the logistics company to call the mortgage function in the ML contract. Merchants deposit twice the shipping fee, logistics companies deposit ether equal to the price of the goods, and each party deposits a fine(see b3 and b4 in Fig. 6). The merchants then call the get contract address function in the MC contract, stores the ML contract address, and updates the item to sent (this is not shown in Fig. 6). So that the consumer and the merchant can't call the cancellation function, as in Algorithm9(The CL contract here will be described in detail in the process of returning the goods). The logistics company calls the product update function to update the current location of the goods and mark the status of the goods as unopened. This represents the result of pre-verification, the logistics company confirmed the authenticity of the product, to prevent the possibility of product replacement. After the start of transportation, each transfer point is reached, the transfer point must ensure the packaging of the product, and call the product update function in the contract to update the current location and owner(see b5 in Fig. 6). At the same time, merchants and consumers will get the logistics information and compare and verify the information on the blockchain at any time.

Algorithm 9 Get Conteact Address

```
    Input: contract address,Merchant,Consumer
1   If(msg.sender==Merchant)
2   {
3       Get ML contract address;
4       Transaction cannot be cancelled;
5   }
6   Else if(msg.sender==Consumer)
7   {
8       Get CL contract address;
9       Product has been returned;
10  }
11  End
```

4) **Receiving goods:** When the last transshipment point delivers the product to the consumer's location, and within t days from the beginning of the delivery, the consumer confirms that the product is correct and then delivers k_1 to the logistics company, as well as to the merchant(see c1 and c2 in Fig. 6). The logistics company sends a transaction to the ML contract notifying it that the product has been received(see c3 in Fig. 6), and then the logistics companies call the payment function in the ML contract. If $h_1 = keccak256(k_1)$, they get all the mortgaged funds and a transportation fee(see d1 in Fig. 6). After the merchant receives the notification, it calls the payment function in the MC contract to get the amount equivalent to the product and the penalty of mortgage(see d2 in Fig. 6), and then the two contracts automatically send the remaining balance to the consumer's account(see d3 in Fig. 6).

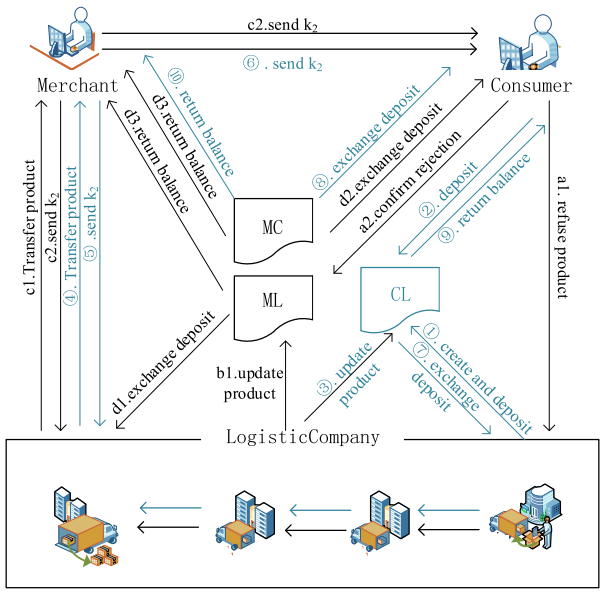

FIGURE 7. Return of the goods.

5) **Returns:** The return of goods is divided into two types. The first type is that when consumers receive products, the delivery time of the products exceeds t days or the products are damaged. Of course, they can return the products without any reason. Then the consumer refuses to receive the product, and calls the product confirmation function in the ML contract to update the product information to rejected(see a1 and a2 in Fig. 7). Logistics companies will also return products according to the original route, and also update the product status during transportation(see b1 in Fig. 7). Upon arrival, the merchant confirms the handover and give k_2 to the logistics company, as well as to the consumer(see c1 and c2 in Fig. 7). The logistics company calls the payment function in the ML contract to confirm that $h_2 = keccak256(k_2)$ and obtains the double transportation fee and redeem the penalty of their mortgage. At the same time, consumers also get the full amount of the mortgage according to k_2, and the contract will return the remaining balance to the merchant(see d1-d3 in Fig. 7).

The second is that the consumer has received the product, but after use is not satisfied, the request returns, at this time the product receiving time should be within t_0 days and confirm with the merchant that the product is not damaged. In order to facilitate consumers, they can choose a suitable logistics company nearby, which can be the same as or different from the previous one. Then the newly selected logistics company establishes a contract with consumers, which is defined as CL contract. The CL contract is the same as the ML contract, in which case the consumer becomes the merchant and the merchant becomes the consumer. First,

FIGURE 8. The result of regulators adding merchant information.

consumer and logistics companies mortgage funds, consumer mortgage transportation fees and fines, logistics companies deposit ether equal to the price of the goods and a fine(see ① and ② in Fig. 7). The product is then shipped and, as before, its location and status are updated with each arrival of the transit point(see ③ in Fig. 7). Upon arrival at the merchant, the merchant checks and sends k_2 to the logistics company and the consumer without any mistake(see ④-⑥ in Fig. 7). The logistics company uses k_2 to verify with $h_2 = keccak256(k_2)$, obtains all the funds for transportation and mortgage from the CL contract(see ⑦ in Fig. 7). Consumers use k_2 to call the payment function in the MC contract to obtain all the mortgage funds and an additional transportation fee(see ⑧ in Fig. 7). The CL contract returns the remaining balance to the consumer's account, the MC contract returns the remaining balance to the merchant account(see ⑨ and ⑩ in Fig. 7). As shown in Fig. 7, the black solid line indicates that the consumers reject the goods, the logistics company returns the goods, and the blue solid line indicates that the consumers receive the products and return the goods within t_0 days.

Algorithm 10 Mediation Dispute

```
    Input: Regolators
1   If(msg.sender==Regolators)
2       Regulators.trandfer(this.balance);
3   End
```

In addition, if a dispute arises between any of the three parties during the transaction and cannot be resolved on its own, the regulator can mediate. All funds in the contract established between the parties in dispute will be transferred to the ethereum address of the regulator, as shown in Algorithm 10. Regulators take evidence judgment out of the chain, allocate funds reasonably according to the results, and collect a fine as punishment for the wrong party, which is also owned by regulators as mediation fees.

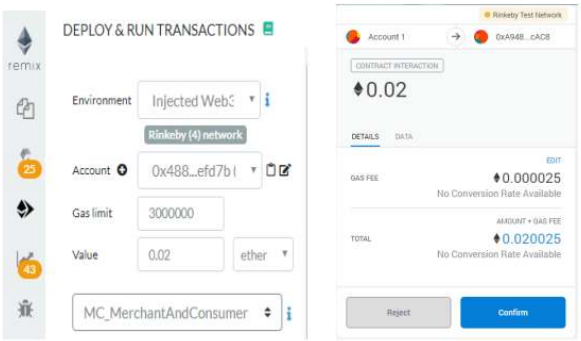

FIGURE 9. (a)mortgage funds into the contract (b)Signature of transaction.

VI. TESTING AND SECURITY ANALYSIS:

A. TESTING AND EVALUATION

1) TESTING

We wrote the contract with solidity in the online editor ''remix''. The operating environment is Injected Web3 and the contract is deployed on Rinkeby(Clique) Testnet.

a: IC CONTRACT

In the IC contract, only the regulator can execute the add member function. Anyone who enters the ethereum address of the member they want to authenticate can get the identity information. This contract can ensure the legitimate rights and interests of merchants and logistics companies, and prevent irregular service providers to join the network as consumers and commit illegal acts. Fig. 8 is the result of regulators adding merchant information. It does not cost gas to view the identity types represented by ethereum addresses.

b: MC CONTRACT

Merchants deploy MC contracts into ethereum, the most important of which is the mortgage function that transfers merchants' and consumers' funds from their accounts to

FIGURE 10. Log of mortgage success.

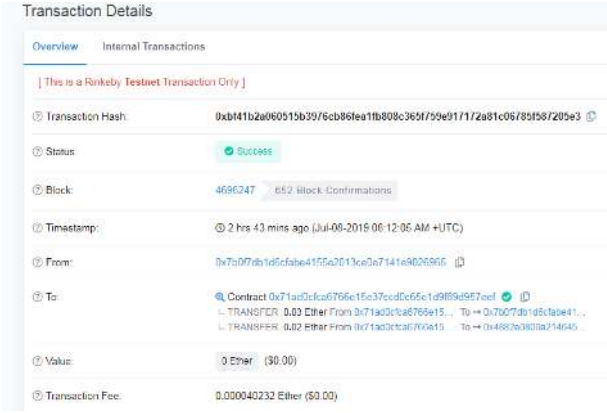

FIGURE 11. The result of execution of the payment function in the ML contract.

the contract accounts. When initiating the transaction call function, fill in the amount to be pledged in the value field, noting that the balance of the account should be greater than the value to be pledged in order for the call to succeed. As shown in Fig. 9(a), the merchant mortgages 0.02 ether into the contract, and Fig. 9(b) shows the signature of the transaction. The generated log is shown in Fig. 10, and the consumer also needs to call the mortgage function to deposit funds.

c: ML CONTRACT

The mortgage function in the ML contract implements the same function as the mortgage function in the MC, and is not explained here. The product update function in ML contract is called by the logistics company to update the current owner and status of the product. The product confirmation function is used to confirm that the product has been accepted or rejected. The results of these two functions are not shown here, we mainly focus on the payment function. The logistics company uses k_1 to exchange the transport fee and mortgage amount, and the contract will return the remaining funds to the merchant. As shown in Fig.11, the from field means that ether are from ML contract, the to field means that 0.03 ether are sent to the logistics company, and 0.02 ether are sent to the merchant. Again, the payment function in MC performs the same function.

2) COST EVALUATION

We tested the ethereum gas cost of the function in the IC contract. Table. 2 shows the gas and actual ether costs for

TABLE 2. Identity Contract cost.

Function	Gas Used	Actual Tx Cost	USD
AddMerber	45096	0.000045096	0.0139
DeleteMerber	29150	0.00002915	0.0090

TABLE 3. Merchant and consumer contract cost.

Function	Gas Used	Actual Tx Cost	USD
MortgageFunction	25238(M)	0.000025238	0.0078
	25228(C)	0.000025228	0.0078
PaymentFunction2	39532	0.000039532	0.0122
CancellationFunction	24221	0.000024221	0.0075
GetContractAddress	43508	0.000043508	0.0135
MediationDispute	30244	0.000030244	0.0094

TABLE 4. Merchant and LogisticsCompany contract cost.

Function	Gas Used	Actual Tx Cost	USD
MortgageFunction	25083(L)	0.000025083	0.0078
	25065(M)	0.000025065	0.0078
ProductUpdate	34337	0.000034337	0.0106
ProductConfirmation	24452	0.000024452	0.0076
PaymentFunction	40232	0.000040232	0.0124

TABLE 5. Comparison with other schemes.

Function	Riham[12]	Zhao[33]	Our scheme
Create order	210102	89027	50466
Product update	176488	-	27788
Accept	8617	111471	34337
Payment	-	48316	39532
Total cost	405207	248814	152123

adding and removing members, respectively. (1gas=1Gwei, 1eth=309.18USD).

Table. 3 shows the gas cost for each function in the MC contract. It can be seen that both sides of the deposit of funds spent gas are basically the same, two payment functions are written in the contract, one is used to pay the merchant, and the other is used to refund the customer. The cost of gas tested is essentially the same, so take the second example. Get the contract function. When the logistics company deployed ML contract and consumers deploy CL contract, the merchants and consumers respectively stored the contract address into the MC contract, and the gas cost was shown in the figure. The cancellation function and dispute resolution functions are used to cancel the transaction and transfer the contract balance to the regulator's account in the event of a dispute, respectively.

Just like MC contract, in the ML contract, the gas cost is basically the same for deposit. The product update function, called at least four times in our scenario, and the gas cost for each time is shown in Table. 4. The product confirmation and payment function are also shown in the table.

Compared with the previous scheme, we evaluated gas. As shown in Table. 5, when creating an order, the customer creates the order in reference [42]. In order to ensure the anonymity of the delivery address, the customer needs to input ciphertext information such as the transportation route. Compared with reference [37] and our scheme, the burden

of consumers is greater. In addition, when transferring goods in literature [42], we need to interact with consumers to obtain the next shipping address and verify the current owner, so it costs more gas. Literature [37] does not involve discussion of recording the delivery information of goods on the blockchain, and our scheme only needs simple update. When receiving goods, literature [37] needs to verify the identity of the receiver in a large number, while our scheme only needs to be confirmed by the corresponding consumers according to the pre-set conditions. Finally, when paying, our scheme can be realized through a simple hash matching process. Under the premise of ensuring safety, reliability, fairness and reasonableness, our scheme reduces the verification work in the transportation process, reduces the workload of logistics companies and the burden of consumers, and improves the transportation efficiency. Most importantly, we designed a complete return process.

B. SECURITY ANALYSIS

1) SECURITY FEATURES

Theorem 1: The proposed payment protocol satisfies the security requirement of fairness.

Proof: First, we assume that the merchant is honest and the consumer wants to get the goods without paying for them. Suppose that when the product reaches the consumer, the consumer only sends k_1 to the logistics company, but not to the merchant. At this point, in order to redeem their own mortgage funds, consumers need k_1 or k_2. But Keccak256 is a very safe and powerful algorithm, and nobody can get k_1 and k_2. In this case, the consumer cannot redeem the money in the contract. And after 2*t days, the merchant can request the regulator to review and get the payment. Therefore, the probability that the merchant cannot obtain payment for goods is negligible.

Theorem 2: The proposed auditable protocol meets the satisfy requirements of accountability.

Proof: The blockchain's public history makes it impossible for participants to deny their actions. ECDSA-secp256k1 signature algorithm is used in ethereum, and all transactions related to an ethereum address will be signed. Malicious acts cannot be denied as long as the standard signature scheme is protected. It can also be said that a signature cannot be forged. As a result, cheating by dishonest people is very unlikely to succeed.

Theorem 3: The proposed auditable protocol satisfies availability.

Proof: Smart contracts deployed on the blockchain provide availability for participants to execute transactions. Anyone can view the transaction history and verify what happened. In addition, ethereum is a distributed organization with thousands of mining nodes maintaining the ledger using consensus algorithms with a high degree of integrity and consistency. Therefore, Ethereum public ledger is highly robust and resistant to Distributed Denial of Service(DDos) attacks, making our system protected by Denial of Service(Dos) attacks.

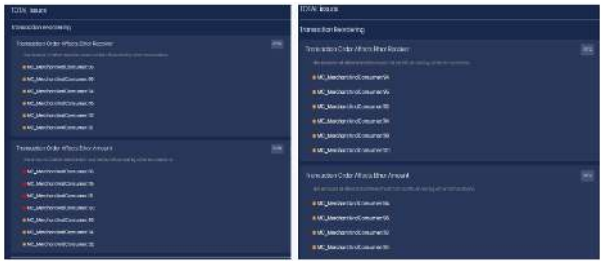

FIGURE 12. (a)safety analysis report (b)Revised safety analysis report.

2) VULNERABILITY ANALYSIS

Once deployed, smart contracts are difficult to modify, so if there are security holes in smart contracts, it is difficult to prevent attacks by hackers. In this case, it's important to ensure that you don't write code that has any security threats. Smart contracts belong to emerging things, so there are still many defects and security holes.

The Decentralized Autonomous Organization(DAO) was one of the major hacking incidents during ethereum's early development. The contract lost 3.6 million ethers and resulted in a hard fork in ethereum's network. Other vulnerabilities in smart contracts include Transaction-Ordering Dependence (TOD), Timestamp Dependency, Error Handling Exception, etc., which can cause significant losses. Therefore, using secure analysis tools to analyze code is critical.

SECURIFY is a security scanner of ethereum smart contracts, created by ICE center, ETH Zurich and ChainSecurity AG, a top provider for smart contract audits. The contract bytecode is first converted into their own custom language, and then compared with a validation module to verify whether its semantics are satisfied. Finally, the security report is generated. Fig. 12 shows the security analysis report for the smart contract. Problems with smart contracts are classified, and info displays detailed reports. The red box said Violation: the contract is guaranteed to violate the vulnerability, orange said Warning: the contract may, but us not guaranteed to violate the vulnerability. Fig. 12(a) is the contract of this scheme, and Fig. 12(b) is the contract modified according to the safety analysis report. We ensure that the contract without any Violation. There are other security analysis tools available at [50], [51].

VII. CONCLUSION

Existing network transactions inherit the shortcomings of centralized frameworks. Buyers and sellers have information asymmetry to a large extent, single point failure, poor credibility and other problems, and are prone to siphon effect. Under such conditions, consumers are vulnerable to fraud. How to implement a distributed network transaction system and ensure the openness, transparency, verifiability and trustworthiness is the main research purpose. Blockchain has core characteristics such as decentralization, data tampering resistance, autonomy, openness and anonymity. It can form a chain to record the state of the network at discontinuous time points. In addition, an external account controlled by the

private key triggers conditions in the smart contract to enforce the agreed rules. In this paper, we propose an auditable protocol for fair payment and physical asset delivery based on smart contracts among merchants, consumers and logistics companies. In our protocol the blockchain with the properties of open, transparent, tamper-proof and verifiable is used to solve the trust problem of transaction nodes in the network. Smart contract is used to manage funds, its Turing-complete function is used to provide good support for the realization of the scheme. The test shows that our auditable solution of fair payment and physical asset delivery based on blockchain is of h high efficiency, high security and high scalability, and transaction costs and risks of participants is reduced.

However, our plan does not involve how consumers ensure the quality and practicality of the products they buy, which involves the credibility of merchants. Therefore, the reputation of the provider becomes our next research direction.

APPENDICES

On Ribkeby Testnet, the address of the contract in our proposal is as follows:

*Identity Contract (IC) Address:*0x4fB6dC2708BD71208a7c16cfF946d189a88c55f6

Merchants And Consumer contracts (MC) Address: 0xA94821c1037e3B4E7B4b7D2c509140c35B18cAC8

Merchant and Logistics Company Contracts (ML) Address: 0x71AD0cFca6766E15E37cCd0c65e1d9f89D957Eef

You can see these contracts and the execution of the functions in the contracts in https://rinkeby.etherscan.io/

The full code can be seen on github with the following link: https://github.com/txxing/SmartContract/tree/master

REFERENCES

[1] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*. Accessed: Dec. 16, 2018. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[2] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in IoT," in *Europe and MENA Cooperation Advances in Information and Communication Technologies* (Advances in Intelligent Systems and Computing), vol. 520, Á. Rocha, M. Serrhini, and C. Felgueiras, Eds. 2017, pp. 523–533.

[3] K. Zhang and H. A. Jacobsen, "Towards dependable, scalable, and pervasive distributed ledgers with blockchains," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Vienna, Austria, vol. 1, Jul. 2018, pp. 1337–1346.

[4] M. Andrychowicz, S. Dziembowski, D. Malinowski, and Ł. Mazurek, "Fair two-party computations via bitcoin deposits," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, vol. 8438, 2014, pp. 105–121.

[5] H. Huang, X. Chen, Q. Wu, X. Huang, and J. Shen, "Bitcoin-based fair payments for outsourcing computations of fog devices," *Future Gener. Comput. Syst.*, vol. 78, pp. 850–858, Jan. 2018.

[6] Y. Zhu, Y. Qin, G. Gan, Y. Shuai, and W. C.-C. Chu, "TBAC: Transaction-based access control on blockchain for resource sharing with cryptographically decentralized authorization," in *Proc. IEEE 42nd Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2018, pp. 535–544.

[7] *Double Deposit Escrow-Bitbay*. Accessed: Nov. 13, 2018. [Online]. Available: https://bitbay.market/double-deposit-escrow/

[8] *Two Party Contracts*. Accessed: Nov. 25, 2018. [Online]. Available: https://dappsforbeginners.wordpress.com/tutorials/two-party-contracts/

[9] L. W. Cong and Z. He. *Blockchain Disruption and Smart Contracts*. Accessed: Dec. 19, 2018. [Online]. Available: https://ssrn.com/abstract=2985764

[10] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep./Oct. 2018.

[11] R. Almadhoun, M. Kadadha, M. Alhemeiri, M. Alshehhi, and K. Salah, "A user authentication scheme of IoT devices using blockchain-enabled fog nodes," in *Proc. IEEE/ACS 15th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Oct./Nov. 2018, pp. 1–8.

[12] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," *Future Gener. Comput. Syst.*, vol. 82, pp. 395–411, May 2018.

[13] A. Suliman, Z. Husain, M. Abououf, M. Alblooshi, and K. Salah, "Monetization of IoT data using smart contracts," *IET Netw.*, vol. 8, no. 1, pp. 32–37, Jan. 2019.

[14] K. Salah, M. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019.

[15] *Robust, Cost-Effiective Applications Key to Unlocking Blockchain's Potential Credit Benefits*. Accessed: Dec. 21, 2018. [Online]. Available: https://www.moodys.com/

[16] *Ethereum White Paper*. Accessed: Dec. 27, 2018. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper

[17] *Ethereum*. Accessed: Dec. 28, 2018. [Online]. Available: https://www.stateofthedapps.com/zh

[18] *BTC*. Accessed: Dec. 28, 2018. [Online]. Available: https://www.8btc.com/article/117055

[19] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.

[20] B. Cao, Y. Li, L. Zhang, L. Zhang, S. Mumtaz, Z. Zhou, and M. Peng, "When Internet of Things meets blockchain: Challenges in distributed consensus," *IEEE Netw.*, to be published.

[21] *Smart Contracts Running on a BFT Hardened Raft*. Accessed: Mar. 15, 2018. [Online]. Available: https://github.com/buckie/juno

[22] *Quorum Whitepaper*. Accessed: Jan. 27, 2019. [Online]. Available: https://github.com/jpmorganchase/quorum-docs

[23] *Ripple*. Accessed: Dec. 28, 2018. [Online]. Available: https://ripple.com/

[24] D. Schwartz, N. Youngs, and A. Britto, *The Ripple Protocol Consensus Algorithm*. Accessed: Jan. 5, 2019. [Online]. Available: http://www.naation.com/ripple-consensus-whitepaper.pdf

[25] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, Jun. 2017, pp. 557–564.

[26] *Quorum*. Accessed: Jan. 27, 2019. [Online]. Available: https://github.com/jpmorganchase

[27] D. Huang, X. Ma, and S. Zhang, "Performance analysis of the raft consensus algorithm for private blockchains," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.

[28] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[29] S. Huckle, R. Bhattacharya, M. White, and N. Beloff, "Internet of Things, blockchain and shared economy applications," *Procedia Comput. Sci.*, vol. 98, pp. 461–466, Jan. 2016.

[30] O. Novo, "Blockchain meets IoT: An architecture for scalable access management in IoT," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1184–1195, Apr. 2018.

[31] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustain. Cities Soc.*, vol. 39, pp. 283–297, May 2018.

[32] Y. Zhang, R. H. Deng, X. Liu, and D. Zheng, "Blockchain based efficient and robust fair payment for outsourcing services in cloud computing," *Inf. Sci.*, vol. 462, pp. 262–277, Sep. 2018.

[33] S. Underwood, "Blockchain beyond bitcoin," *Commun. ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[34] Q. Lin, H. Yan, Z. Huang, W. Chen, J. Shen, and Y. Tang, "An ID-based linearly homomorphic signature scheme and its application in blockchain," *IEEE Access*, vol. 6, pp. 20632–20640, 2018.

[35] Y. Wang and J. Gao, "A regulation scheme based on the ciphertext-policy hierarchical attribute-based encryption in bitcoin system," *IEEE Access*, vol. 6, pp. 16267–16278, 2018.

[36] Y. He, H. Li, X. Cheng, Y. Liu, C. Yang, and L. Sun, "A blockchain based truthful incentive mechanism for distributed P2P applications," *IEEE Access*, vol. 6, pp. 27324–27335, 2018.

[37] Y. Zhao, Y. Li, Q. Mu, B. Yang, and Y. Yu, "Secure pub-sub: Blockchain-based fair payment with reputation for reliable cyber physical systems," *IEEE Access*, vol. 6, pp. 12295–12303, 2018.

[38] J. Ahn, M. Park, and J. Paek, "Reptor: A model for deriving trust and reputation on blockchain-based electronic payment system," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, vol. 6, Oct. 2018, pp. 1431–1436.

[39] *What is Escrow? How Does Escrow Work? -Escrow.com*. Accessed: Oct. 15, 2018. [Online]. Available: https://www.escrow.com/what-is-escrow

[40] *How Our Escrow Smart Contract Works*. Accessed: Oct. 16, 2018. [Online]. Available: https://blog.localethereum.com/how-our-escrow-smart-contract-works

[41] K. Toyod, P. T. Mathiopoulo, I. Sasase, and T. Ohtsuk, "A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain," *IEEE Access*, vol. 5, pp. 17465–17477, 2017.

[42] R. Altawy, M. ElSheikh, A. M. Youssef, and G. Gong, "Lelantos: A blockchain-based anonymous physical delivery system," in *Proc. 15th Annu. Conf. Privacy, Secur. Trust (PST)*, Aug. 2018, pp. 15–24.

[43] K. Salah and H. Hasan, "Blockchain-based solution for proof of delivery of physical assets," in *Proc. Int. Conf. Blockchain (ICBC)*. Cham, Switzerland: Springer, vol. 10974, 2018, pp. 139–152.

[44] H. Hasan and K. Salah, "Blockchain-based proof of delivery of physical assets with single and multiple transporters," *IEEE Access*, vol. 6, no. 1, pp. 46781–46793, Dec. 2018.

[45] *Istanbul Byzantine Fault Tolerance*. Accessed: Jan. 27, 2019. [Online]. Available: https://github.com/ethereum/EIPs/issues/650

[46] C. Li, B. Palanisamy, and R. Xu, "Scalable and Privacy-preserving Design of On/Off-chain Smart Contracts," 2019, *arXiv:1902.06359*. [Online]. Available: https://arxiv.org/abs/1902.06359

[47] J. Benet. *IPFS—Content Addressed, Versioned, P2P File System*. Accessed: Mar. 27, 2019. [Online]. Available: https://github.com/ipfs/ipfs/blob/master/papers/ipfs-cap2pfs/ipfs-p2p-file-system.pdf

[48] Y. Chen, H. Li, K. Li, and J. Zhang, "An improved P2P file system scheme based on IPFS and Blockchain," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2017, pp. 2652–2657.

[49] H. Hasan and K. Salah, "Combating deepfake videos using blockchain and smart contracts," *IEEE Access*, vol. 7, no. 1, pp. 41596–41606, Dec. 2019.

[50] H. Hasan and K. Salah, "Proof of delivery of digital assets using blockchain and smart contracts," *IEEE Access*, vol. 6, pp. 65439–65448, 2018.

[51] *Ethereum Smart Contract Best Practices*. Accessed: Jul. 5, 2019, [Online]. Available: https://consensys.github.io/smart-contract-best-practices/

SHANGPING WANG received the B.S. degree in mathematics from the Xi'an University of Technology, Xi'an, China, in 1982, the M.S. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, in 1989, and the Ph.D. degree in cryptology from Xidian University, Xi'an, in 2003. He is currently a Professor with the Xi'an University of Technology. His current research interests include cryptography and information security.

XIXI TANG received the B.S. degree from the School of Science, Xi'an University of Technology, Xi'an, China, in 2017, where she is currently pursuing the M.S. degree. Her research interests include information security and blockchain technology.

YALING ZHANG received the B.S. degree in computer science from Northwest University, Xi'an, China, in 1988, and the M.S. degree in computer science and the Ph.D. degree in mechanism electron engineering from the Xi'an University of Technology, Xi'an, in 2001 and 2008, respectively, where she is currently a Professor. Her current research interests include cryptography and network security.

JUANJUAN CHEN received the Ph.D. degree from Shaanxi Normal University, Xi'an, Shaanxi, China, in 2014. She is currently a Lecturer with the Xi'an University of Technology. Her current research interests include cryptography and information security.