applied sciences

MDPI

Article

A Traceable and Verifiable Tobacco Products Logistics System with GPS and RFID Technologies

Chin-Ling Chen [1,2,3], Zi-Yi Lim [3,*], Hsien-Chou Liao [3,*], Yong-Yuan Deng [3] and Peizhi Chen [1,*]

[1] School of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, China; clc@mail.cyut.edu.tw
[2] School of Information Engineering, Changchun Sci-Tech University, Changchun 130600, China
[3] Department of Computer Science and Information Engineering, Chaoyang University of Technology, 168 Jifeng East Road, Taichung 41349, Taiwan; allendeng@cyut.edu.tw
* Correspondence: zylim@cyut.edu.tw (Z.-Y.L.); hcliao@cyut.edu.tw (H.-C.L.); pzc@xmut.edu.cn (P.C.)

Abstract: Tobacco products are an addictive commodity. According to the World Health Organization's (WHO) latest statistics data, tobacco kills more than eight million people each year. In 2003, the WHO proposed the Framework Convention on Tobacco Control (FCTC) to provide an effective framework for the control of tobacco products to governments around the world. In the field of tobacco products, the hardest problem is how to prevent counterfeit tobacco products and smuggling. To solve the problems, we proposed a blockchain-based traceable and verifiable logistics system for tobacco products with global positioning system (GPS) and radio-frequency identification (RFID) Technologies. In this research, we provide an overview of system architecture, and also define the protocol and the smart contract in every phase that stores data into the blockchain center. We realized a decentralized database and authentication system that uses blockchain and smart contract technology; every protocol in every phase was designed to achieve the integrity of data and non-repudiation of message. Every tobacco product's shipping record will be completed by scanning the RFID tag and retrieving the GPS with a mobile reader, where the record will be updated and validated in the blockchain center. In the end, the security and costs of the system were analyzed, and a comparison was made with the EU's (European Commission) method. Our system is more flexible for transportation, more secure in the communication protocol, and more difficult to tamper and forge data. In general, the proposed scheme solved the problem of tobacco products counterfeiting and tracking issues.

Keywords: tobacco products; blockchain; traceable; global positioning system (GPS); radio-frequency identification (RFID); logistics

Citation: Chen, C.-L.; Lim, Z.-Y.; Liao, H.-C.; Deng, Y.-Y.; Chen, P. A Traceable and Verifiable Tobacco Products Logistics System with GPS and RFID Technologies. *Appl. Sci.* **2021**, *11*, 4939. https://doi.org/10.3390/app11114939

Academic Editors: Marc Kurz and Erik Sonnleitner

Received: 7 May 2021
Accepted: 25 May 2021
Published: 27 May 2021

1. Introduction

1.1. Background

According to the Food and Drug Administration (FDA), tobacco products means "any product made or derived from tobacco that is intended for human consumption, including any component, part, or accessory of a tobacco product (except for raw materials other than tobacco used in manufacturing a component, part, or accessory of a tobacco product)" [1]. Tobacco products include cigarettes, cigars, etc. Tobacco contains the stimulant alkaloid nicotine, which easily makes people addicted.

Since the emergence of tobacco, there have been various studies showing the disadvantages of tobacco products, which can easily damage the lungs, heart, and liver. Common diseases of smoking tobacco include cancer, cardiovascular disease, chronic bronchitis, male sexual dysfunction, pregnant women affect the fetus, etc. Except for the diseases it causes to smokers, smoking tobacco also causes air pollution, and second-hand smoke will also affect the people around smokers. According to the World Health Organization's

(WHO) latest statistical data, tobacco kills more than eight million people each year, with seven million people deaths because of direct tobacco use and 1.2 million non-smokers who are exposed to second-hand smoke [2].

In 2003, the WHO published the Framework Convention on Tobacco Control (FCTC) which provides price and tax measures and non-price measures to reduce the demand for tobacco products [3]. In these decades, many countries worldwide are constantly following the framework established by the WHO, especially by raising the tax rate of tobacco products. Because of the increase in tax rates, the price of tobacco products to the consumer has also increased. Most of the manufacturers or retailers do not report the actual sales volume to the government to more profit, which means that more and more smuggled tobacco products are sold on the market. Smuggled tobacco products cannot be controlled and certified by the government, and there may be elements in the product that affect human health more [4]. In addition, the government has the responsibility to care nation's health as well as the responsibility to eliminate the smuggled tobacco products.

Recently, to eliminate smuggled products, the European Commission has worked out the mechanisms of tobacco product tracking [5]. In addition, the United States' FDA also regulates tobacco products with strict standards [6].

To track and identify products, we need a data carrier along with sticks or prints on the package of products. The most common and simple data carrier on the market is the barcode, which include one-dimensional (1D) and two-dimensional (2D) barcodes. GS1 is the largest not-for-profit organization [7], and they developed the global standards for barcodes, so that the standards can be universally used all over the world. Barcodes have many limitations that make them unsuitable for logistics tracking such as easy to damage, easy to copy, easy to decode, and they must be scanned individually. Because of these shortcomings, other options have to replace the barcode in the next technology, which is radio-frequency identification (RFID). RFID is a technology that is able to store and retrieve data in the RFID tags, so is the best method to use in supply chains [8,9]. RFID accelerates the speed of inventory, and it is more convenient to calculate the total amount of incoming and outgoing logistics.

Unfortunately, data carriers alone cannot track the delivery process of tobacco products. The information in the system can be easily maliciously or deliberately falsified by anyone. Blockchain technology can solve the problem of information reliability. Kamilaris et al. [10] proposed a blockchain-based method for agriculture and food supply chains, where the research showed that blockchain can make the supply chain more transparent and reliable. Therefore, blockchain is a new technology that can realize the traceability and authentication of the logistics record.

In this study, we proposed a traceable and verifiable tobacco products logistics system that involved blockchain, GPS and RFID technologies. All the tobacco packages produced by the manufacturer must have an RFID tag with ID. These IDs are issued by an official organization. After the tobacco products are produced, all the logistics processes must be sent and chained in the blockchain center. The proposed scheme achieves the goal of data decentralization, is hard to tamper with, has traceability, and is authenticated. It is also convenient for the consumer to verify, for manufacturers to manage, and for the auditor to audit.

1.2. Related Works

Several related works are listed in Table 1. The key points of concern are listed in the table.

Table 1. Survey of related works.

Authors	Year	Objective	Technologies	1	2	3	4	5	6
Wyld [9]	2008	RFID tag sticks to cigarettes and taxes	RFID	N	Y	N	N	Y	N
Wang et al.[11]	2011	RFID tag integration into a cigarette pack	RFID	N	Y	N	N	Y	N
Carvalho et al. [8]	2012	Deploy RFID to fashion supply chains	RFID	N	N	Y	Y	Y	N
Shi et al. [12]	2012	Track and trace system for supply chains	RFID	N	N	Y	Y	Y	Y
Prasanna et al. [13]	2012	Logistics vehicle load balancing and tracking mechanism	RFID, GPS, GSM	N	N	Y	Y	Y	N
Li et al. [14]	2016	Tobacco logistics retroactive system	RFID, database	N	Y	Y	Y	Y	N
Liu et al. [15]	2018	Financial service platform for tobacco supply chain	Blockchain	Y	Y	Y	N	N	N
Humayun et al. [16]	2020	Smart logistics and transportation using IoT and Blockchain	Blockchain, IoT	Y	N	Y	Y	N	N

Notes: 1: Focus on a blockchain, 2: Focus on tobacco products, 3: Proposing an architecture or framework, 4: Focus on logistics, 5. RFID/GPS-enabled, 6: Security analysis, Y: Yes, N: No.

RFID technology has been used in the trace system for supply chains for many years. Wyld [9] evaluated the uses of RFID tags on cigarette packs. The research explains how the RFID works as well as how the RFID can solve the products' smuggling problem and inventory control as the technology can be scanned to verify whether the taxes have been paid or not. Wang et al. [11] redesigned a passive ultra high frequency (UHF) RFID tag and integrated it into a cigarette pack, where the tag was only 0.5 mm thick. Neither of these two studies mentioned how to use RFID to implement a supply chain system.

Some research has been implemented in the supply chain of non-tobacco products. In 2012, Carvalho et al. [8] deployed RFID technology in the case of fashion supply chain management (FSCM). Shi et al. [12] also proposed a RFID-enabled trace system implemented with the Electronic Product Code (EPC) global network, which is a global data exchange standard for supply chain networks, where the user can query product information via scanning the RFID with the EPC. These studies have proposed a good architecture for implementation in the supply chain, but the studies lacked integrity and traceability of data.

Aside from implementing RFID technology, global positioning system (GPS) can also be integrated into logistics tracking systems. Prasanna et al. [13] proposed a logistics vehicle tracking mechanism in 2012, where the authors implemented a GPS device in the vehicle to record and track the real-time location and used the global system for mobile communication (GSM) to upload those data to the server. The GPS helps to bind and record the delivery location of the products, and this location can improve the data completeness of the logistics system. Unfortunately, the authors did not analyze or prove the security of their system.

Regarding tobacco product-related systems, Li et al. [14] proposed a retroactive system with a database. Their system could solve the problem of product tracking, but the security of traditional databases may be challenged. For example, if the administrator's account is maliciously logged in, the data in the database can be easily tampered with by the attacker and hard to trace. Liu et al. [15] briefly introduced a financial service platform for the tobacco supply chain. These studies have provided a framework of the tobacco products' supply chain system, however, although the second research mentioned the use of blockchain technology, the description was not complete, and it did not mention how blockchain was applied.

Recently, Humayun et al. [16] applied blockchain technology and IoT in the logistics system. The research analyzed the advantages of applying blockchain and IoT technology, but they did not analyze the security issues.

The studies listed above have advantages and disadvantages for us to refer to. Consequently, we proposed a system with a comprehensive architecture with security. In this research, we applied blockchain, RFID, and GPS technology to achieve the complete tobacco products' logistics system.

The remaining sections of this paper are organized as follows. Section 2 briefly introduces the technologies that are used in our proposed scheme. Section 3 proposes our

scheme. Section 4 analyzes the security issues. The computation cost, communication performance, and comparison discussion are given in Section 5. Finally, Section 6 concludes this paper.

2. Preliminary

2.1. Consortium Blockchain and Smart Contract

Blockchain is a technology that was carried forward by Nakamoto [17], who used blockchain to realize a decentralized peer-to-peer cryptocurrency. Blockchain is a distributed database, with the characteristic that it is difficult to arbitrarily tamper.

A smart contract is a program that can be executed automatically. The most famous blockchain-based smart contract in the world was implemented by Buterin in 2014 [18], who also founded Ethereum. The blockchain-based smart contract can be implemented in various domains, for example, digital property [19], logistics systems [20], the exhibition of cultural relics [21], will management [22], firearm management [23], etc.

In general, the most widely used blockchain currently is Ethereum [18], which is a public blockchain where everyone can store and validate the block data transparently. In order to solve privacy issues and make blockchain technology more suitable for enterprises, more types of blockchain architectures have also been developed such as private blockchain [24] and consortium blockchain [25]. In particular, the consortium blockchain is a blockchain between public and private. Generally, it is possible to specify how many peers can own the ledger, and then which peer can conduct transactions and own part of the ledger. On the other hand, every blockchain system has an important consensus algorithm to validate the blockchain, and there are various types of consensus algorithms such as Proof-of-Work (PoW), Proof-of-Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT) [26].

To solve the problems encountered in business such as data privacy issues, real-time transactions, modular expansion, etc., the Linux Foundation has developed a blockchain architecture of chaincode (also known as a smart contract with additional features) that is more suitable for commercial applications, the Hyperledger Fabric [27]. The Hyperledger Fabric is a permissioned blockchain, where the chain contains the chaincode, ledger, and channel. The common implementation of the consensus algorithm in the blockchain is PBFT. Therefore, it is different from other types of blockchain architecture because it does not need a cryptocurrency-based mechanism to mine, in order to validate the ledger or execute the smart contract.

Figure 1 shows an example of the Hyperledger Fabric network from the official documentation [28]. The notation of nodes in the figures are as follows: Application (A), Certificate Authority (CA), Channel (C), Peer (P), Ordering Service (O), Ledger (L), Chaincode (S), Organization (R), and Channel Configuration (CC). The organization is defined by the Membership Services Provider (MSP), where every organization configures the channel configuration and makes each node join in a secure private channel. The certificate authority generates the certificates to the nodes, and the certificates must be signed in every transaction. When any client executes an application, the application sends a transaction proposal to all endorsing peers via the configured channel. These peers reply to a signed proposal response to the application. The transaction will package into a block by ordering the service node; the node also orders and distributes it to every peer, then the transaction is done and updates to the blockchain network.

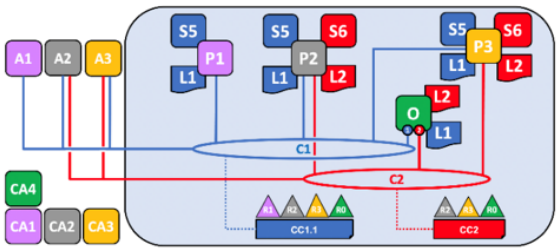

Figure 1. Hyperledger Fabric blockchain network [28].

Our scheme proposes tobacco products logistics with Hyperledger Fabric framework, where the purpose is to have more throughput for the transactions, solve the personal privacy problem, and be more suitable for the government to manage.

2.2. ECDSA

The elliptic curve digital signature algorithm (ECDSA) is the derived type of the digital signature algorithm (DSA) [29]. Johnson et al. [30] introduced that ECDSA be accepted in any global standard. ECDSA is accepted in the following standards: ISO 14888-3, ANSI X9.62, IEEE 1363-2000, FIPS 186-4, ANSI X9.142-2020.

ECDSA involves the concept of elliptic curve cryptography (ECC), the characteristic of ECC reduces the key size in the algorithm and also provides a faster calculation speed compared to DSA. According to the NIST's minimum security-strength requirement [31], the length of n with 224 bits and SHA-512/224 for digital signature generation is required.

2.3. BAN Logic

Burrows–Abadi–Needham Logic (BAN Logic) is a method of authenticating communication protocols that was first proposed by Burrows et al. [32] in 1990. It is important to prove the security and integrity of the protocol, and the main purpose is to prove that there are no security issues in the security protocol, and that the protocol can also meet the designer's expectations of the method.

2.4. Threat Model

According to Table 1, we sorted and reviewed the past research, and found some research gaps. Therefore, we sorted out the threat patterns that need to be overcome. The threats are generally due to system security vulnerabilities, which may cause the system to be attacked illegally by an external malicious person, or may cause the system to crash and leak data. As a result, the tobacco products logistics system will suffer from potential risks. The related risks are defined as follows:

(1) Data integrity issues: All data stored with the system must be integrated. The system must ensure that the data will not be modified by anyone during the transmission and storing process.
(2) Decentralized database: The blockchain center can be known as the decentralized database, with multiple agencies maintaining the same ledger or data in different locations. Once the data are verified and added to the blockchain, the block is chained with a timestamp and the previous block hash value; every modification with the blockchain needs to be verified. Therefore, it is hard to change the data in the blockchain center, and the decentralization characteristics can achieve data transparency and reliability.
(3) Decentralized authentication: The authentication inherits the decentralized database's characteristics. This is more secure than the general database that is set up with the central server architecture.
(4) Message repudiation issues: To ensure the undeniable transmission of the message sent by the sender, a signature mechanism needs to be implemented to prove the message is signed by the sender.
(5) Message transmission issues: The system must be ensured that the message will not be intercepted and altered during transmission.
(6) Tobacco product counterfeiting issues: The counterfeiting of tobacco products can harm the health of the smoker, they never know the legal origin and quality of the products. Furthermore, the counterfeiting products will also cause the original manufacturer to lose their reputation.
(7) Tobacco product tracking issues: These issues are linked to counterfeit issues. The government needs a complete tracking system to manage the logistics status of tobacco products.

(8) Fair arbitration: Every system that is managed or used by a human with multiple parties cannot avoid dispute. To avoid disputing the legality of tobacco products, fair arbitration should be considered to clarify smuggled tobacco products.

(9) Known attacks:

a. Man-in-the-middle attack: The sender needs to communicate to the receiver, the attacker intercepts the message in the middle, then the attacker is able to obtain the message from the sender and resend the same message to the receiver. Therefore, the message will be exposed, because the attack can easily obtain the content of the message in the middle.

b. Replay attack: The attacker sniffs the message sent by the sender, so the attackers can replay the message.

3. The Proposed Scheme

In this research, we proposed a blockchain-based traceable and authenticated tobacco products logistics system with GPS and RFID technologies. The proposed system is constituted of the following twelve parties: ID issuer, tobacco ID tag, mobile reader, competent authorities, manufacturer, distributor, logistics, retailer, consumer, arbitrator, auditors, and blockchain center. The system framework is shown in Figure 2.

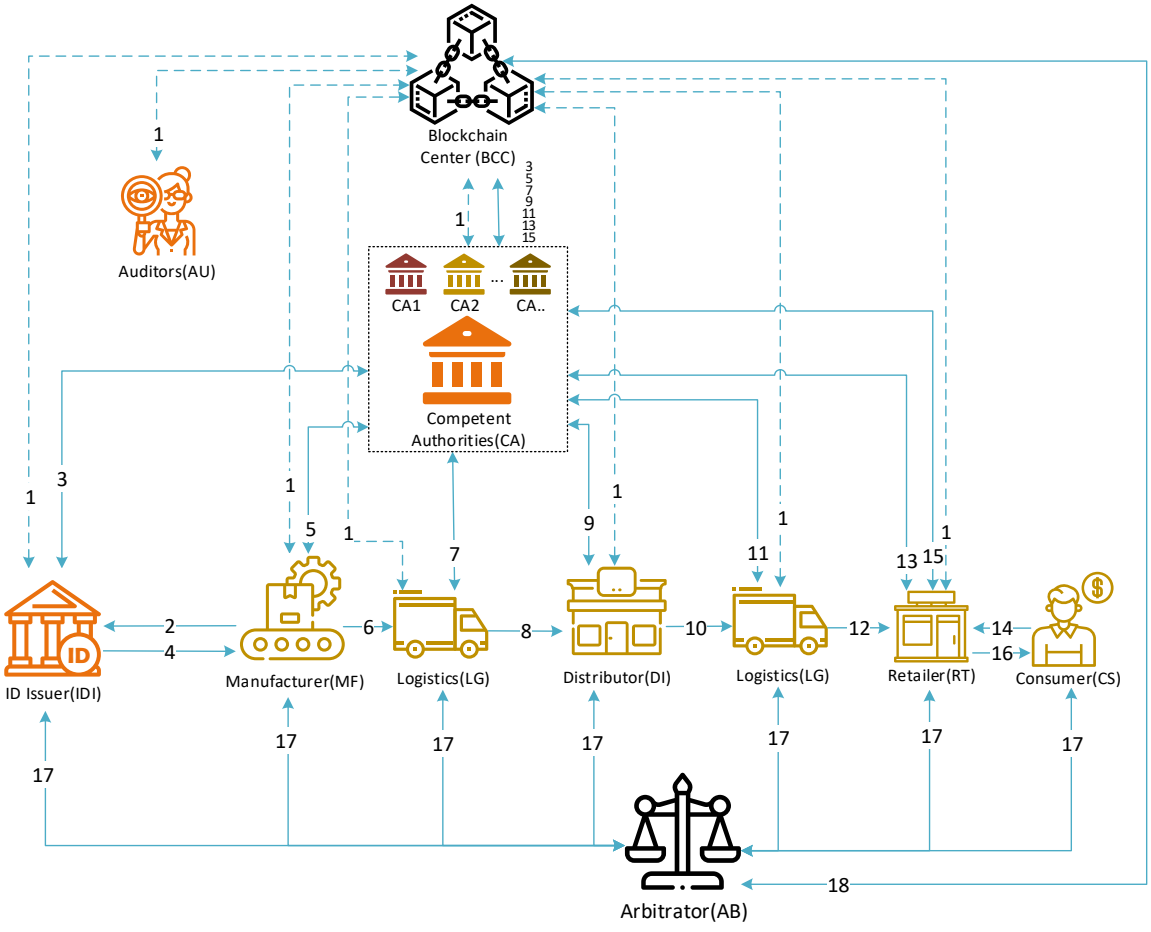

Figure 2. System architecture diagram.

3.1. System Architecture

1. ID Issuer (IDI): An official organization authorized by the government of the country. They receive the application of the tobacco products' ID from the manufacturer.

This party checks the validity of the manufacturer; if they are a legal company, a list of tobacco products will be generated by the chaincode in the blockchain center.

2. Tobacco ID Tag (TID): A unique ID issued by the ID issuer. The ID is a sticker-based material with a RFID tag. Every pack of legal tobacco products should have one ID tag to prevent smuggled products.
3. Mobile reader (MR): A mobile device that can read tobacco ID tags and can position the location with GPS. Every party involved in the logistics phase must have at least one MR such as the manufacturer, logistics, distributor, or retailer. These parties need to log in to the MR with their private key. Every TID on the tobacco products should be scanned by MR when the tobacco products are produced, sent, or received by the shipper or recipient.
4. Competent authorities (CA): Multiple official organizations that are authorized by the government of the country such as the Food and Drug Administration (FDA), Federal Trade Commission (FTC), Tobacco Tax, Trade Bureau (TTB), etc. Every CA must have the ledger in the blockchain center to ensure and verify the integrity of the data. The CA that provide the permits for the production of tobacco products have the highest authority and mainly deal with any connection from other parties.
5. Manufacturer (MF): A company that produces tobacco products; it can also be a tobacco importer. Before producing tobacco products, they need to apply the product's ID from the ID Issuer. The ID that is applied and obtained from the IDI should be applied to the tobacco package.
6. Blockchain Center (BCC): The blockchain that records the logistics information of the tobacco products. When a manufacturer needs to produce tobacco products, they need to apply an ID from the IDI, the ID is generated from the BCC's chaincode, then the TID is sent to IDI via CA before finally going to the MF. Every record of the tobacco products will need to be chained with the given ID. The chaincode in the BCC keeps checking the number of tobaccos during the logistics.
7. Distributor (DI): A company buying a large number of tobacco products from the MF is known as a wholesaler. They are also sellers who sell products to retailers.
8. Logistics (LG): A company responsible for transporting the tobacco products. They mostly use trucks to transport the products. All products entering or leaving the transportation need to scan the logistics information via MR to BCC and chain it, for example, the ID, timestamp, and GPS location.
9. Retailer (RT): A shop or store selling the unit packet of tobacco products to the consumer.
10. Consumer (CS): An ordinary person that needs to buy tobacco products from the retailer.
11. Arbitrator (AB): An official agent that is able to use a mobile device with the Internet to find counterfeit or illegal tobacco products whether the tobacco products are in retailers, distributors, etc.
12. Auditors (AU): A third-party agency. If either party is unsure of the legal source of the tobacco products, the auditors have the right to prove if there are any problems in the logistics process.

Figure 2 presents the scenarios that illustrate the process of tobacco products from the manufacturer to consumers through the distributor, retailer, and multiple logistics. There will be more than one manufacturer, distributor, retailer, and logistics in reality, so we used the basic elements to represent in this figure. A detailed description is as follows:

Step 1. All parties involved in the tobacco products logistics chain must register an account from the BCC to obtain a unique ID and a private and public key pair.

Step 2. When the manufacturer needs to produce a batch of tobacco products, they need to apply to the IDI for the ID to every pack, batch, or any aggregation level of the tobacco products.

Step 3. The IDI sends the application to the CA, then the CA requests from the BCC to execute a chaincode. The BCC responds to the IDI with a list of IDs that corresponds to every pack of tobacco products.
Step 4. The IDI approves the application from the MF and responds a list of IDs to them.
Step 5. After receiving the IDs, MF produces tobacco products. Every pack of tobacco products needs to send the GPS location, ID, the timestamp of production, and the MF's ID by the MR to the CA, then the information is sent and chained in the BCC.
Step 6. The MF requests LG to deliver the products to the DI to distribute the tobacco products.
Step 7. When LG receive the products, they need to scan and send the products' GPS location, ID, timestamp, and LG ID to the CA, then the information is sent and chained in the BCC.
Step 8. After LG arrive at the DI, the information of the GPS location, ID, timestamp, and LG ID also needs to be sent and chained in the BCC via CA.
Step 9. DI needs to scan and send the products' GPS location, ID, timestamp, and DI's ID to CA, the information is sent and chained in the BCC.
Step 10. DI needs LG to deliver the products while the DI sells the tobacco products to the RT.
Step 11. Once LG receives the products from the DI, they need to scan and send the products' GPS location, ID, timestamp, and LG ID to the CA, then the information is sent and chained in the BCC.
Step 12. LG delivers the tobacco products to the RT, the information of the GPS location, ID, timestamp, and LG ID also needs to be sent and chained in the BCC via the CA.
Step 13. To ensure the validity and total amount, the RT requests to scan and send the products' GPS location, ID, timestamp, and RT ID to the CA in the last logistics session, then the information is sent and chained in the BCC. Next, the retailer starts to sell tobacco products to the consumers.
Step 14. A consumer goes to a retailer to buy tobacco products. The consumer needs to provide their ID for the legal transaction.
Step 15. RT sends all the transactional information including the CS ID, transaction ID, RT ID, and timestamp to the CA, then the transactional information is sent and chained in the BCC.
Step 16. The transaction is done between the RT and CS. The tobacco product is traced until this step.
Step 17. If any party has a dispute or doubts the legality of the tobacco product, the party can submit an arbitration request to the arbitrator.
Step 18. The details of the tobacco products' logistic records were chained in the BCC, and the AB can retrieve and verify the logistics record from the BCC.

3.2. Notation

ID_X	X is the identity, issued by blockchain center. The format of the ID is [random serial number + timestamp] (total 144 bits)
TID_Y	Y is the tobacco identity, which is issued by ID issuer. The format of the ID is [random serial numbers + ID Issuer ID + manufacturer ID + manufacturing timestamp] (total 224 bits)
q	A k-bit of prime number
$GF(q)$	Finite group of q
E	The elliptic curve defined on finite group
G	A generating point based on the elliptic curve E
k_i	The i^{th} random value on the elliptic curve
(r_{X_i}, s_{X_i})	Elliptic curve signature value of X
(x_{X_i}, y_{X_i})	An ECDSA signature value of X

d_X	The ECDSA's private key of party X
Q_X	The ECDSA's public key of party X
Puk_X	The public key of party X, issued by the BCC
Prk_X	The private key of party X, issued by the BCC
C_{X_i}	The ith ciphertext of X
$H(M)$	One way hash function
h_{X_i}	The ith hash value of X
T_i	The ith timestamp
τ	The threshold for checking the validity of a timestamp
M_i	The ith message from a sender
$E_{Puk_X}(M) / D_{Prk_X}(M)$	Encrypt/decrypt message M with a public key or private key of party X
$F1 \stackrel{?}{=} F2$	Verify that $F1$ is equal to $F2$ or not

3.3. Initialization Phase

First, we raised a blockchain center network architecture, as shown in Figure 3. There are three types of peers in the network. The CA peer is the peer governed by multiple competent authorities, so every CA's peer has a chaincode and blockchain ledger. The BCC network also has a blockchain certificate authority (BCA). The BCA is authorized by the government department to issue authorization-related certificates to every access party such as the MF, LG, DI, and RT. Every access party has its own private channel that connects to the CA's network, and the ledger is synchronized with the CA's peers. Every information update through the execution of chaincode must be verified and endorsed by the CA's peer. If it is valid, the ordering peers will order the transaction record to all of the CA's peers.

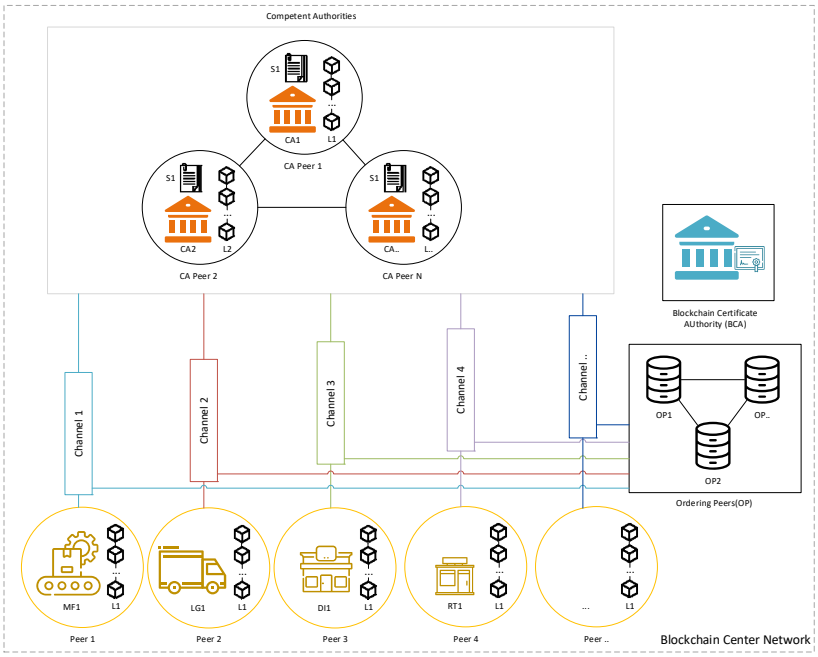

Figure 3. Blockchain center network architecture.

Figures 4 and 5 show the fundamental chaincode structure of our scheme. Figure 4 shows the structure to store the information of the access parties (APs); the enumeration

of the role type is defined on the right side. Figure 5 shows the structure to store the tobacco product information, where every detail of the tobacco product will be appended to the structure when it passes through each access party.

```
type AP_Information struct {
        ID string
        Name string
        Detail string
        var RoleTypes Roles
}
```

```
type Roles string
const(
        IDIssuer
        Manufacturer
        Distributor
        Logistics
        Retailer
        Arbitrator
        Auditors
)
```

Figure 4. Chaincode structure of the access party and the enumeration of the role type.

```
type Tobacco_Product struct {
        TID string
        Product_Information string
        Generate_Datetime time.Time
        Manufacturer_ID string
        Manufacturing_Datetime DateTime
        Manufacturing_Factory_ID string
        Manufacturing_GPSLocation string
        var TRecord []TransportRecord
        Retailer_ID string
        Payment_Datetime time.Time
        Payment_GPSLocation string
        Purchase_ID string
        Payment_Price float32
        CA_Signature string
        IDI_Signature string
        MF_Signature string
        RT_Signature string
        BatchTID string
}
```

```
type TransportRecord struct {
        Shipper_ID string
        Shipper_GPSLocation string
        Shipper_Datetime time.Time
        Recipient_ID string
        Recipient_GPS string
        Recipient_Datetime time.Time
        SHP_Signature string
        RCP_Signature string
}

type Batch struct {
        BatchID string
        TIDs []string
        CA_Signature string
        IDI_Signature string
        MF_Signature string
}
```

Figure 5. The chaincode structure of the tobacco products.

3.4. Registration Phase

All parties that want to be a part of the system need to register from the BCA. The BCA generates and sends the public key and private key pair to the parties. The registration process from any access party to the blockchain center is shown in Figure 6.

Step 1. AP provides the primary information (e.g., name, role) and sends a registration request to the BCC.

Step 2. BCC generates an ECDSA private key d_X and calculates public key Q_X :

$$Q_X = d_X G \tag{1}$$

The application of registration needs to be manually approved by the CA. If the application is approved, the chaincode "Registration" will be triggered; the algorithm is shown in Algorithm 1. Then, BCC sends (ID_X, d_X, Q_X) parameters to the AP.

Step 3. AP receives (ID_X, d_X, Q_X) and keeps the parameters for signing the signature message.

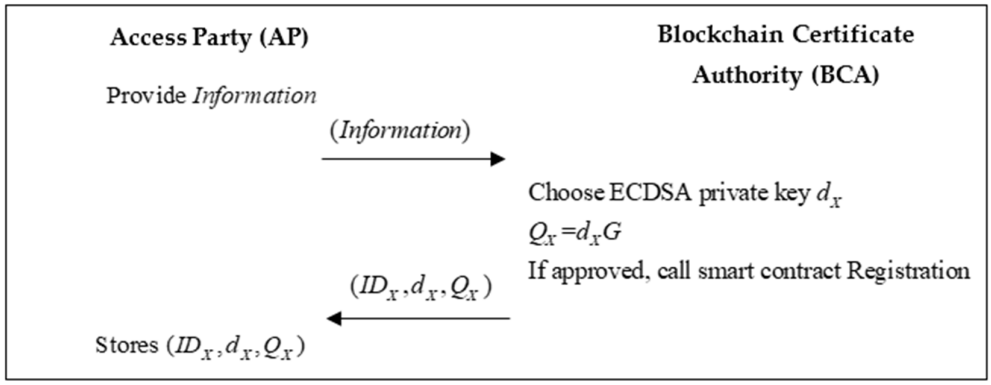

Figure 6. The flowchart of the registration phase.

Algorithm 1. Chaincode registration of the proposed scheme.

```
var AP []AP_Information
func Registration (X_name string, X_detail string, var X_role RoleType) (UID string) {
        UID = GenerateUniqueID()
        AP = append (AP, AP_Information{
            ID: UID,
            Name: X_name,
            Detail: X_detail,
            Role: X_role,
        })
        return UID
}
```

3.5. Authentication Phase

In this phase, we assumed that user A is a sender, and user B is a receiver. Every sender and receiver needs to sign and verify their message with the "Sign" and "Verify" function that is shown in Algorithm 2. These two functions implement the ECDSA to achieve identity authentication. The sender generates a signature with a "Sign" function to the receiver. When the receiver receives the message, they execute a "Verify" function to verify. Similarly, when the receiver needs to respond to a message from the sender, the receiver also needs to execute a "Sign" function, then the sender needs to execute the "Verify" function when receiving the response message to ensure each other's true identities. The flow of the process is shown in Figure 7.

Step 1. Firstly, user A chooses a random number k_1, then generates the message:

$$M_1 = (ID_A \| ID_B \| T_1) \tag{2}$$

Next, user A calculates the parameters of ECDSA:

$$h_1 = H(M_1) \tag{3}$$

$$(x_{A_1}, y_{A_1}) = k_1 G \tag{4}$$

Afterward, the signatures are generated by executing the function "Sign" in Algorithm 2. In detail, it generates the signatures with the parameters:

$$r_{A_1} = x_{A_1} \bmod n \tag{5}$$

$$s_{A_1} = x_{A_1}^{-1}(h_1 + r_{A_1} d_A) \bmod n \tag{6}$$

A message is encrypted by user B's public key:

$$C_{A_1} = E_{Puk_B}(M_1) \tag{7}$$

User A sends a message $(ID_A, ID_B, C_{A_1}, (r_{A_1}, s_{A_1}))$ to user B.

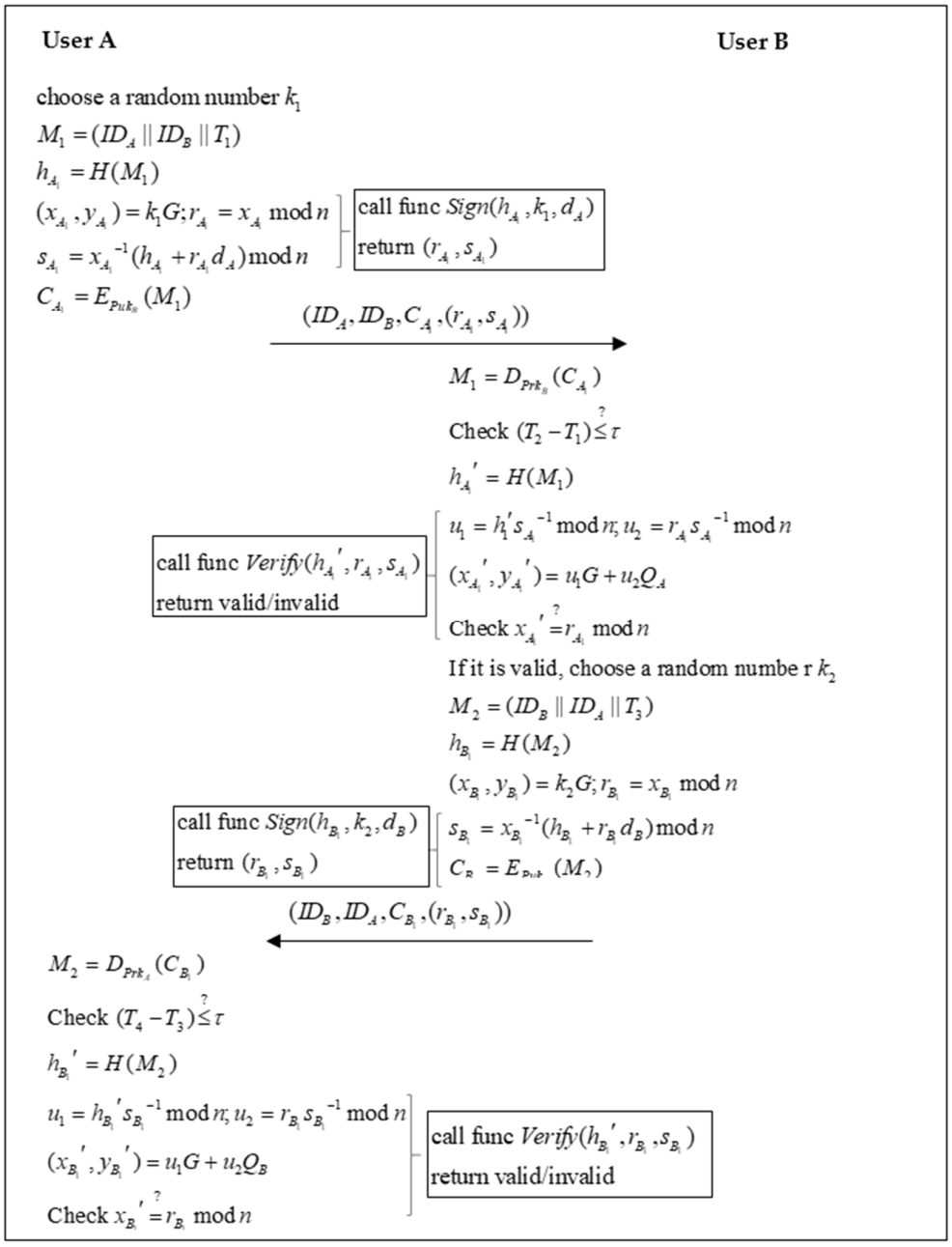

Figure 7. The flowchart of the authentication phase.

Step 2. IDI receives the message at T_2 and uses its private key to decrypt C_{A_1}:

$$M_1 = D_{Prk_B}(C_{A_1}) \tag{8}$$

Then, user B checks for the validation of the timestamp:

$$(T_2 - T_1) \overset{?}{\leq} \tau \tag{9}$$

Next, user B executes the function "Verify" in Algorithm 2. In detail, it calculates the following parameters:

$$h_{A_1}' = H(M_1) \tag{10}$$

$$u_1 = h_{A_1}' s_{A_1}^{-1} \bmod n \tag{11}$$

$$u_2 = r_{A_1} s_{A_1}^{-1} \bmod n \tag{12}$$

$$(x_{A_1}', y_{A_1}') = u_1 G + u_2 Q_A \tag{13}$$

User B uses those calculated parameters to validate the signature:

$$x_{A_1}' \stackrel{?}{=} r_{A_1} \bmod n \tag{14}$$

If the signature is valid, then user B selects a random number k_2 and generates a message:

$$M_2 = (ID_B \,||\, ID_A \,||\, T_3) \tag{15}$$

Next, user B calculates the hash value and the parameters of ECDSA to generate the signatures (r_{B_1}, s_{B_1}). The signatures are generated by executing the function "Sign" in Algorithm 2:

$$h_{B_1} = H(M_2) \tag{16}$$

$$(x_{B_1}, y_{B_1}) = k_2 G \tag{17}$$

$$r_{B_1} = x_{B_1} \bmod n \tag{18}$$

$$s_{B_1} = x_{B_1}^{-1}(h_{B_1} + r_{B_1} d_B) \bmod n \tag{19}$$

A message is encrypted by user A's public key:

$$C_{B_1} = E_{Puk_A}(M_2) \tag{20}$$

Then, user B sends the message $(ID_B, ID_A, C_{B_1}, (r_{B_1}, s_{B_1}))$ to user A.

Step 3. User A receives the message at T_4, then decrypts the cipher message by its private key:

$$M_2 = D_{Prk_A}(C_{B_1}) \tag{21}$$

Then, user A checks for the validation of the timestamp:

$$(T_4 - T_3) \stackrel{?}{\leq} \tau \tag{22}$$

Next, user A executes the function "Verify" in Algorithm 2. User A calculates the parameters:

$$h_{B_1}' = H(M_2) \tag{23}$$

$$u_1 = h_{B_1}' s_{B_1}^{-1} \bmod n \tag{24}$$

$$u_2 = r_{B_1} s_{B_1}^{-1} \bmod n \tag{25}$$

$$(x_{B_1}', y_{B_1}') = u_1 G + u_2 Q_B \tag{26}$$

User A uses those calculated parameters to validate the signature:

$$x_{B_1}' \stackrel{?}{=} r_{B_1} \bmod n \tag{27}$$

Algorithm 2. Authentication of the proposed scheme.

```
func Sign (h string, k string, d string) (r string, s string) {
        (x, y) = k * G;
        r = x % n
        s = (h + r * d)/x % n
        return r, s
}

func Verify (h string, r string, s string) (result string) {
        u1 = h/s % n
        u2 = r/s % n
        (x, y) = u1 * G + u2 * Q
        if x == r {
                return "valid"
        }else{
                return "invalid"
        }
}
```

3.6. Issuing ID and Manufacture Phase

This is the most important phase to generate the ID of the tobacco products. The flowchart is shown in Figure 8. The related chaincode is shown in Algorithm 3.

Step 1. Before the MF starts manufacturing the tobacco products, they need to send an application to the IDI to get a obtain of TID. First, the MF chooses a random number k_3, then generates the message with the number and information of the tobacco products:

$$M_3 = (ID_{MF} \,||\, ID_{IDI} \,||\, Num \,||\, Info \,||\, T_5) \tag{28}$$

Next, the MF calculates the hash value with the message:

$$h_{MF_1} = H(M_3) \tag{29}$$

and executes the function "Sign" shown in Algorithm 2 to generate the signatures:

$$(r_{MF_1}, s_{MF_1}) = Sign(h_{MF_1}, k_3, d_{MF}) \tag{30}$$

A message is encrypted by the IDI's public key:

$$C_{MF_1} = E_{Puk_{IDI}}(M_3) \tag{31}$$

MF sends a message $(ID_{MF}, ID_{IDI}, C_{MF_1}, (r_{MF_1}, s_{MF_1}))$ to IDI to apply TID.

Step 2. IDI receives the message at T_6 and uses its private key to decrypt C_{MF_1}:

$$M_3 = D_{Prk_{IDI}}(C_{MF_1}) \tag{32}$$

Then, the IDI checks for the validation of the timestamp:

$$(T_6 - T_5) \overset{?}{\leq} \tau \tag{33}$$

Next, IDI calculates the parameters:

$$h_{MF_1}{}' = H(M_3) \tag{34}$$

IDI executes the function "Verify" in Algorithm 2 to validate the signature:

$$Verify(h_{MF_1}{}', r_{MF_1}, s_{MF_1}) \tag{35}$$

If the signature is valid, then IDI sends a request to the CA for the further ID issuing process. IDI selects a random number k_4 and generates a message:

$$M_4 = (ID_{MF} \,||\, ID_{IDI} \,||\, ID_{CA} \,||\, Num \,||\, Info \,||\, T_7) \tag{36}$$

Next, IDI calculates the hash value and executes the function "Sign" in Algorithm 2 to generate the signature:

$$h_{IDI_1} = H(M_4) \tag{37}$$

$$(r_{IDI_1}, s_{IDI_1}) = Sign(h_{IDI_1}, k_4, d_{IDI}) \tag{38}$$

A message is encrypted by the CA's public key:

$$C_{IDI_1} = E_{Puk_{CA}}(M_4) \tag{39}$$

Then, the IDI sends the message $(ID_{IDI}, ID_{CA}, C_{IDI_1}, (r_{IDI_1}, s_{IDI_1}))$ to the CA.

Step 3. Once CA receives the message at T_8, then decrypt the cipher message by its private key:

$$M_4 = D_{Prk_{CA}}(C_{IDI_1}) \tag{40}$$

Then, CA checks for the validation of the timestamp:

$$(T_8 - T_7) \overset{?}{\leq} \tau \tag{41}$$

Next, CA calculates the hash value:

$$h_{IDI_1}{}' = H(M_4) \tag{42}$$

The CA executes the function "Verify" in Algorithm 2 to validate the signature:

$$Verify(h_{IDI_1}{}', r_{IDI_1}, s_{IDI_1}) \tag{43}$$

If the signature is valid, then the chaincode "IDIssue" is triggered to generate a list of TID, the algorithm of which is shown in Algorithm 3. First, CA selects a random number k_5 and generates a message with a list of TID:

$$M_5 = (ID_{CA} \,||\, ID_{IDI} \,||\, ID_{MF} \,||\, List < TID > || \, T_9) \tag{44}$$

Next, the CA calculates the hash value and executes "Sign" in Algorithm 2 to generate the signature:

$$h_{CA_1} = H(M_5) \tag{45}$$

$$(r_{CA_1}, s_{CA_1}) = Sign(h_{CA_1}, k_5, d_{CA}) \tag{46}$$

A message is encrypted by the CA's public key:

$$C_{CA_1} = E_{Puk_{IDI}}(M_5) \tag{47}$$

Then, CA sends the message $(ID_{\mathrm{CA}}, ID_{IDI}, C_{CA_1}, (r_{CA_1}, s_{CA_1}))$ to IDI.

Step 4. IDI receives the message at T_{10}, then decrypts the cipher message by its private key:

$$M_5 = D_{Prk_{\mathrm{IDI}}}(C_{CA_1}) \tag{48}$$

Then, IDI checks for the validation of the timestamp:

$$(T_{10} - T_9) \overset{?}{\leq} \tau \tag{49}$$

Next, IDI calculates the hash value:

$$h_{CA_1}{}' = H(M_5) \tag{50}$$

IDI executes the function "Verify" in Algorithm 2 to validate the signature:

$$Verify(h_{CA_1}{}', r_{CA_1}, s_{CA_1}) \tag{51}$$

If the signature is valid, IDI sends a response message to MF with a list of TID. IDI selects a random number k_6 and generates a message:

$$M_6 = (ID_{MF} \,||\, ID_{IDI} \,||\, List < TID > || \, T_{11}) \tag{52}$$

Next, IDI calculates the hash value and executes the function "Sign" in Algorithm 2 to generate the signature:

$$h_{IDI_2} = H(M_6) \tag{53}$$

$$(r_{IDI_2}, s_{IDI_2}) = Sign(h_{IDI_2}, k_6, d_{IDI}) \tag{54}$$

A message is encrypted with MF's public key:

$$C_{IDI_2} = E_{Puk_{MF}}(M_6) \tag{55}$$

Then, IDI sends the message $(ID_{IDI}, ID_{MF}, C_{IDI_2}, (r_{IDI_2}, s_{IDI_2}))$ to MF.

Step 5. MF receives the message from IDI, then decrypts the cipher message by its private key:

$$M_6 = D_{Prk_{MF}}(C_{IDI_2}) \tag{56}$$

Then, MF checks for the validation of the timestamp:

$$(T_{12} - T_{11}) \overset{?}{\leq} \tau \tag{57}$$

Next, MF calculates the parameter:

$$h_{IDI_2}{}' = H(M_6) \tag{58}$$

MF executes the function "Verify" in Algorithm 2 to validate the signature:

$$Verify(h_{IDI_2}{}', r_{IDI_2}, s_{IDI_2}) \tag{59}$$

If the signature is valid, then the MF is able to use the list of TID to manufacture. When the MF produces a tobacco product, a TID should be stuck with the pack, then the MR is used to scan the tag, and a chaincode "Manufacture" is triggered to update the information of the tobacco product, the algorithm of which is shown in Algorithm 3.

Algorithm 3. Chaincode of issuing ID and manufacture of the proposed scheme.

```
var TP []Tobacco_Product
count:= 0
func IDIssue (Info string, string ID_MF, int num, string CA_Signature, IDI_Signature
string, MF_Signature string) (list_TID []string){
    for i:= 0 ; i < num ; i++ {
    count = count + 1
        TP = append (TP, new Tobacco_Product{
            TID: GenerateTID(),
            Product_Information: Info,
            Generate_Datetime: time.Now(),
            Manufacturer_ID: MID,
            CA_Signature: CA_Signature,
            IDI_Signature: IDI_Signature,
            MF_Signature: MF_Signature
        })

        list_TID = append(list_TID, TP[count].TID)
    }
    return list_TID
}
func Manufacture (
ID_MF string, TID string, GPSLoc string){
    index:= SearchTID(TP, TID)
    TP[index].Manufacturing_Datetime = time.Now()
    TP[index].Factory_ID = ID_MF
    TP[index].GPSLocation = GPSLoc
}
```

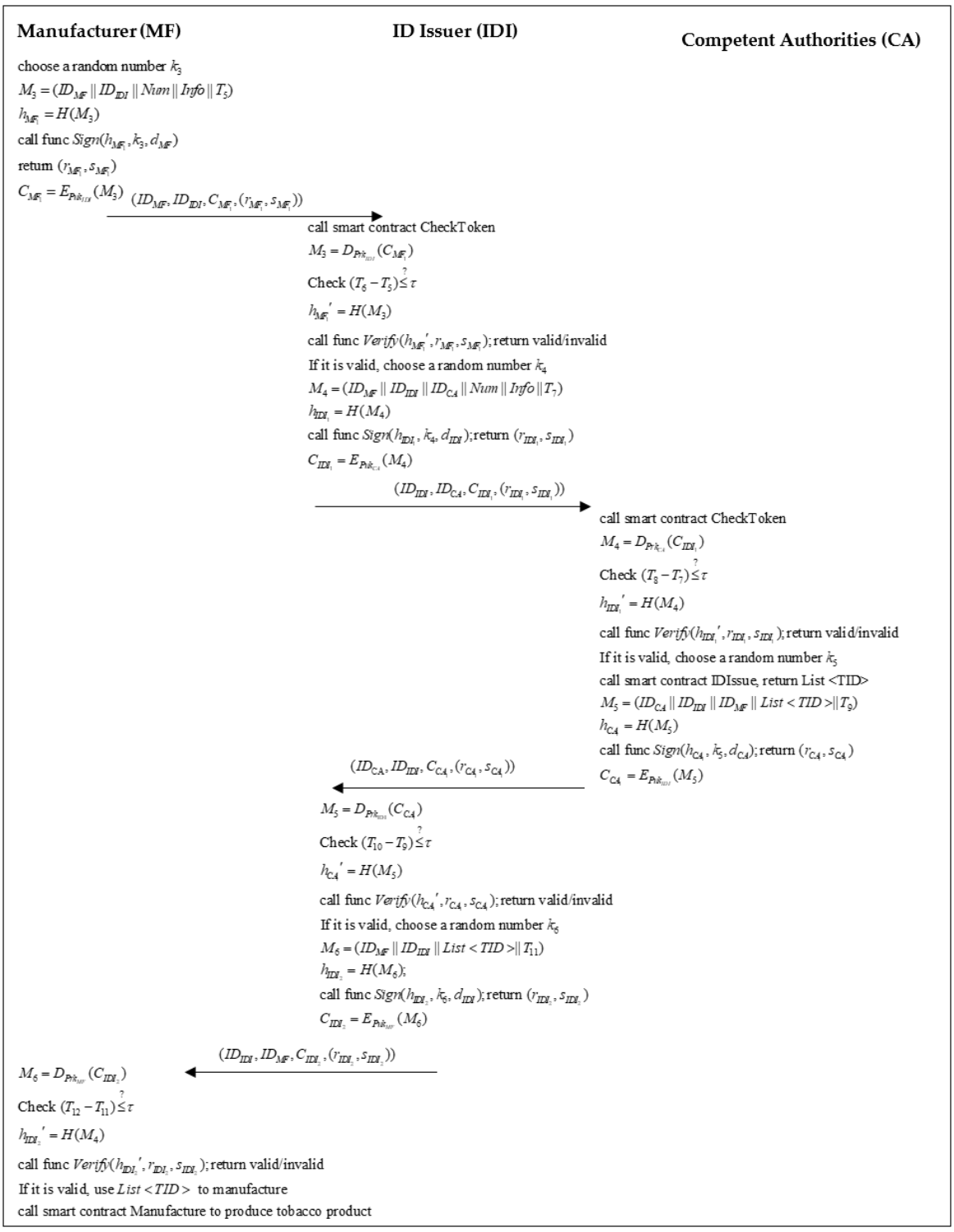

Figure 8. The flowchart of the issuing ID and manufacture phase.

3.7. Logistics Phase

We assumed three roles to operate in this phase: shipper (SHP), logistics (LG), and the recipient (RCP). As per the system architecture shown in Figure 2, we can assume that the shipper is a manufacturer, distributor, or retailer; the recipient is a distributor or retailer; and the logistics is a company that transfers tobacco products between the shipper

and recipient. The flowchart of the logistics phase is shown in Figures 9 and 10. The chaincode of logistics is shown in Algorithm 4.

Step 1. When the SHP needs to ship a batch of tobacco products to the RCP, SHP must ship via LG. Initially, SHP selects a random number k_7 and generates a message with a list of TID:

$$M_7 = (ID_{SHP} \,||\, ID_{LG} \,||\, List < TID > || \, T_{13}) \tag{60}$$

SHP calculates the hash value and executes "Sign" in Algorithm 2 to generate signatures (r_{SHP_1}, s_{SHP_1}):

$$h_{SHP_1} = H(M_7) \tag{61}$$

$$(r_{SHP_1}, s_{SHP_1}) = Sign(h_{SHP_1}, k_7, d_{SHP}) \tag{62}$$

A message is encrypted by LG's public key:

$$C_{SHP_1} = E_{Puk_{LG}}(M_7) \tag{63}$$

Then, SHP executes the chaincode function "Shipping", as shown in Algorithm 4. Next, SHP sends the message $(ID_{SHP}, ID_{LG}, C_{SHP_1}, (r_{SHP_1}, s_{SHP_1}))$ to LG.

Step 2. LG receives the message at T_{14}, then decrypts the cipher message by its private key:

$$M_7 = D_{Prk_{LG}}(C_{SHP_1}) \tag{64}$$

Then, LG checks for the validation of the timestamp:

$$(T_{14} - T_{13}) \overset{?}{\leq} \tau \tag{65}$$

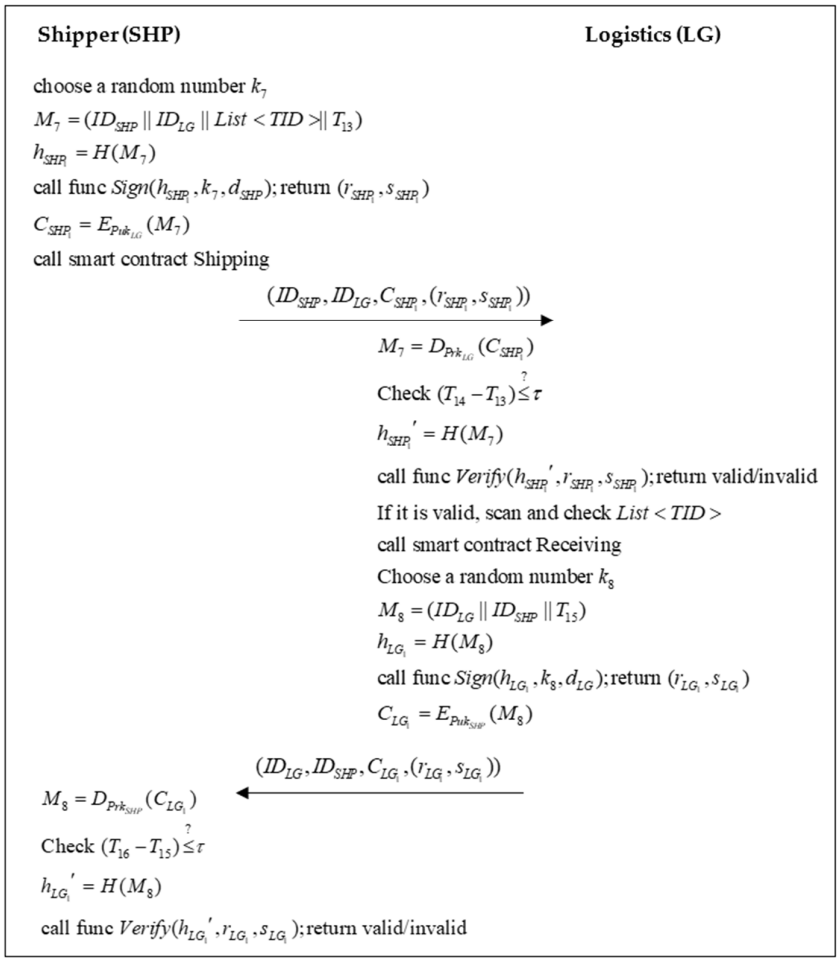

Figure 9. The flowchart of the logistics phase from shipper to logistics.

Next, LG calculates the hash value to validate the signature via "Verify" in Algorithm 2:

$$h_{SHP_1}{}' = H(M_7) \tag{66}$$

$$Verify(h_{SHP_1}{}', r_{SHP_1}, s_{SHP_1}) \tag{67}$$

If the signature is valid, LG using MR triggers a chaincode function "Receiving" to scan and check that $List < TID >$ is equal to the actual tobacco products in the real world, the function of which is shown in Algorithm 4. Then, LG sends a response message to the SHP. LG selects a random number k_8 and generates a message:

$$M_8 = (ID_{LG} \,||\, ID_{SHP} \,||\, T_{15}) \tag{68}$$

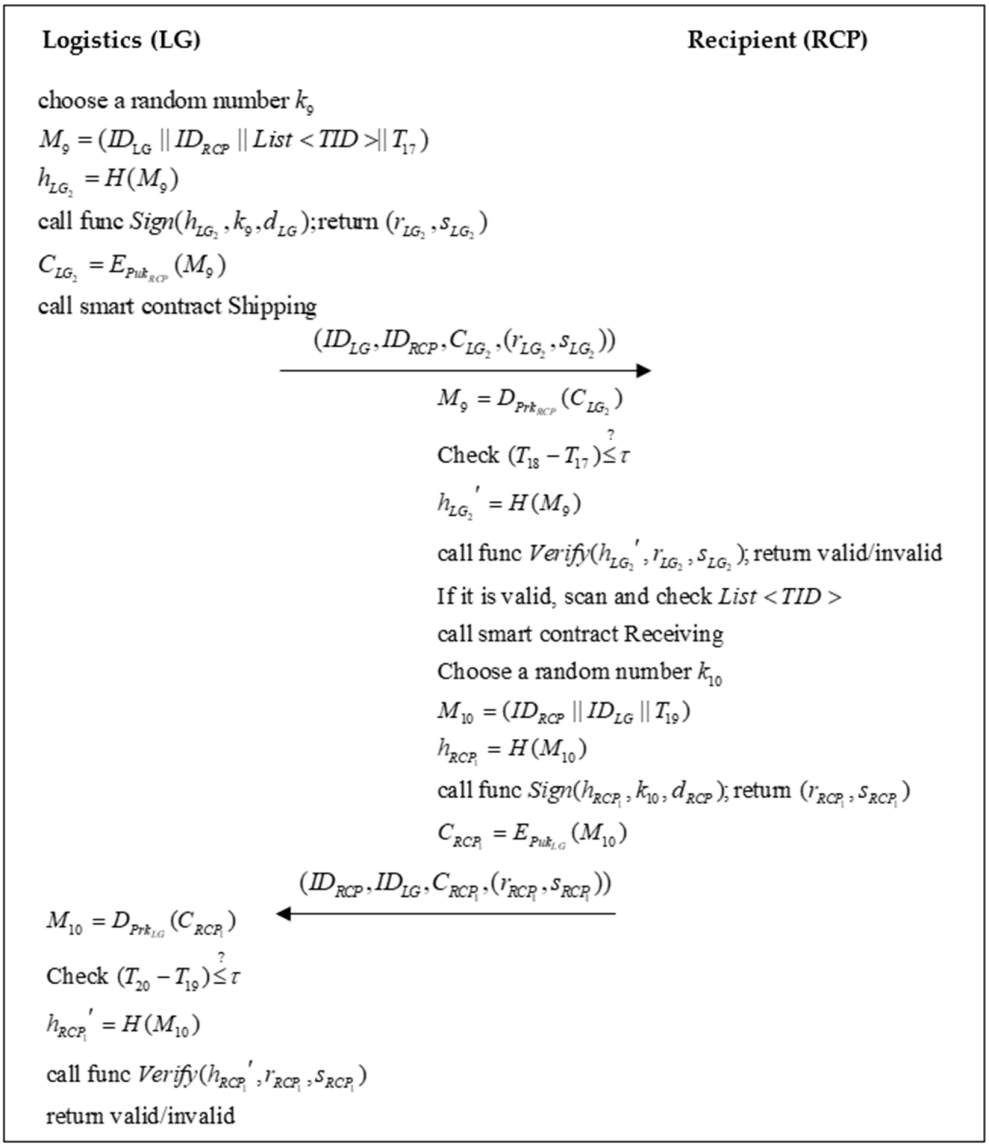

Figure 10. The flowchart of the logistics phase from logistics to the recipient.

Next, LG calculates the hash value to generate the signatures via executing a "Sign" function in Algorithm 2:

$$h_{LG_1} = H(M_8) \tag{69}$$

$$(r_{LG_1}, s_{LG_1}) = Sign(h_{LG_1}, k_8, d_{LG}) \quad (70)$$

A response message is encrypted with SHP's public key:

$$C_{LG_1} = E_{Puk_{SHP}}(M_8) \quad (71)$$

Then, LG sends the message $(ID_{LG}, ID_{SHP}, C_{LG_1}, (r_{LG_1}, s_{LG_1}))$ to SHP.

Step 3. SHP receives the message from LG, then uses its private key to decrypt the cipher message:

$$M_8 = D_{Prk_{SHP}}(C_{LG_1}) \quad (72)$$

Then, SHP checks for the validation of the timestamp:

$$(T_{16} - T_{15}) \stackrel{?}{\leq} \tau \quad (73)$$

If the timestamp is valid, SHP calculates the hash value to validate the signatures via executing a "Verify" function in Algorithm 2:

$$h_{LG_1}{}' = H(M_8) \quad (74)$$

$$Verify(h_{LG_1}{}', r_{LG_1}, s_{LG_1}) \quad (75)$$

If the signature is valid, then SHP starts shipping these tobacco products to the RCP.

Algorithm 4. Chaincode of logistics of the proposed scheme.

```
func Shipping (
ID_SHP string, ID_RCP string, TIDs []string, GPSLoc string, Signature string) {
    for i:= 0 ; i < TIDs.Length ; i++ {
        index:= SearchTID(TIDs[i]);
        TP[index].TransportRecord.Shipper_ID = ID_SHP
        TP[index].TransportRecord.Shipper_GPSLocation = GPSLoc
        TP[index].TransportRecord.Shipper_Datetime = time.Now()
        TP[index].TransportRecord.Recipient_ID = ID_LG
        TP[index].TransportRecord.SHP_Signature = Signature
    }
}
func Receiving (
ID_SHP string, ID_RCP string, TIDs []string, GPSLoc string, Signature string) {
    for i:= 0 ; i < TIDs.Length ; i++ {
        index:= SearchTID(TIDs[i])
        TP[index].TransportRecord.Recipient_GPSLocation = GPSLoc
        TP[index].TransportRecord.Recipient_Datetime = time.Now()
        TP[index].TransportRecord.RCP_Signature = Signature
    }
}
```

Step 1. When LG arrives at the RCP location, LG starts communication and sends the batch of tobacco products to RCP. First, LG selects a random number k_9 and generates a message with a list of TID:

$$M_9 = (ID_{LG} \,||\, ID_{RCP} \,||\, List < TID > || \, T_{17}) \quad (76)$$

LG calculates the hash value and executes the function "Sign" in Algorithm 2 to generate the signature:

$$h_{LG_2} = H(M_9) \quad (77)$$

$$(r_{LG_2}, s_{LG_2}) = Sign(h_{LG_2}, k_9, d_{LG}) \quad (78)$$

A message is encrypted by the RCP's public key:

$$C_{LG_2} = E_{Puk_{RCP}}(M_9) \quad (79)$$

Then, LG executes chaincode "Shipping", as shown in Algorithm 4. Next, LG sends the message $(ID_{LG}, ID_{RCP}, C_{LG_2}, (r_{LG_2}, s_{LG_2}))$ to RCP.

Step 2. RCP receives the message at T_{18}, then decrypts the cipher message by its private key:

$$M_9 = D_{Prk_{RCP}}(C_{LG_2}) \tag{80}$$

Then, RCP checks for the validation of the timestamp:

$$(T_{18} - T_{17}) \stackrel{?}{\leq} \tau \tag{81}$$

Next, RCP calculates the hash value and executes the function "Verify" in Algorithm 2 to validate the signature:

$$h_{LG_2}{}' = H(M_9) \tag{82}$$

$$Verify(h_{LG_2}{}', r_{LG_2}, s_{LG_2}) \tag{83}$$

If the signature is valid, RCP uses MR to trigger a chaincode "Receiving" to scan and check that $List < TID >$ is equal to the actual tobacco products, as shown in Algorithm 4. If the batch of tobacco products has no problem, then RCP sends a response message to LG. RCP selects a random number k_{10} and generates a message:

$$M_{10} = (ID_{RCP} \,||\, ID_{LG} \,||\, T_{19}) \tag{84}$$

Next, RCP calculates the hash value and executes the function "Sign" to generate the signatures:

$$h_{RCP_1} = H(M_{10}) \tag{85}$$

$$(r_{RCP_1}, s_{RCP_1}) = Sign(h_{RCP_1}, k_{10}, d_{RCP}) \tag{86}$$

A response message is encrypted with the LG's public key:

$$C_{RCP_1} = E_{Puk_{LG}}(M_{10}) \tag{87}$$

Then, the RCP sends the message $(ID_{RCP}, ID_{LG}, C_{RCP_1}, (r_{RCP_1}, s_{RCP_1}))$ to LG.

Step 3. LG receives the message, then uses its private key to decrypt the cipher message:

$$M_{10} = D_{Prk_{LG}}(C_{RCP_1}) \tag{88}$$

Then, LG checks for the validation of the timestamp:

$$(T_{20} - T_{19}) \stackrel{?}{\leq} \tau \tag{89}$$

If the timestamp is valid, LG calculates the hash value and executes the function "Verify" to validate the received signatures:

$$h_{RCP_1}{}' = H(M_{10}) \tag{90}$$

$$Verify(h_{RCP_1}{}', r_{RCP_1}, s_{RCP_1}) \tag{91}$$

If the signature is valid, then a round of shipping from the SHP to RCP is done.

3.8. Consumption Phase

When a consumer goes to purchase tobacco products from a retailer, the processing flow is shown in Figure 11.

Step 1. CS goes into the RT location and selects one or more tobacco products, then proceeds to the checkout process.

Step 2. RT starts the checkout process. First, the RT scans the TID of chosen tobacco products from CS. Then, RT selects a random number k_{11} and generates a message with a list of TID:

$$M_{11} = (ID_{RT} \,||\, ID_{CA} \,||\, List < (TID, \mathit{Price}, DateTime) > || \, T_{21}) \tag{92}$$

RT calculates the hash value and executes the function "Sign" in Algorithm 2 to generate the signatures:

$$h_{RT_1} = H(M_{11}) \tag{93}$$

$$(r_{RT_1}, s_{RT_1}) = Sign(h_{RT_1}, k_{11}, d_{RT}) \tag{94}$$

A message is encrypted by CA's public key:

$$C_{RT_1} = E_{Puk_{CA}}(M_{11}) \tag{95}$$

Then, RT sends the message $(ID_{LG}, ID_{RCP}, C_{LG_2}, (r_{LG_2}, s_{LG_2}))$ to CA.

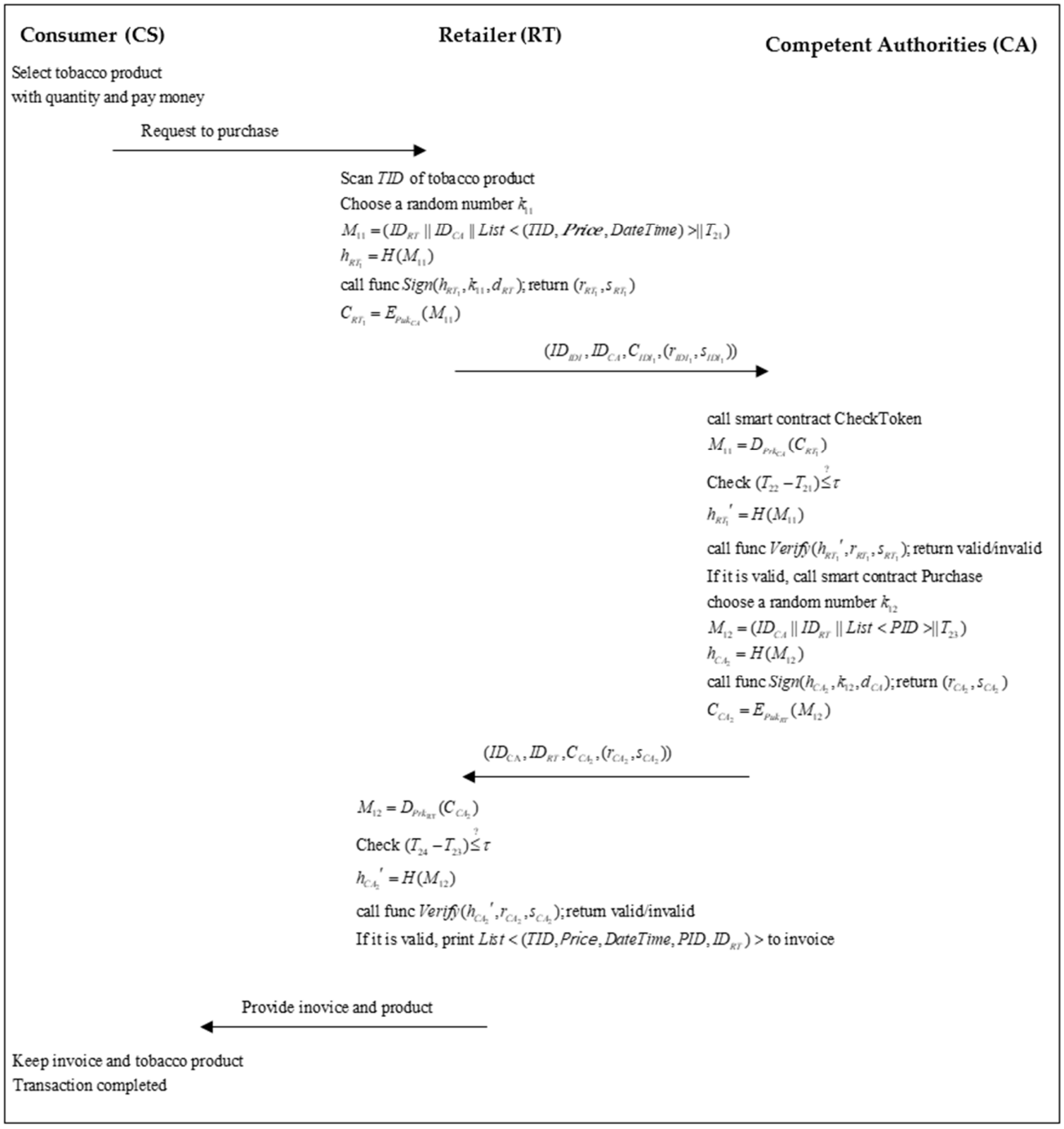

Figure 11. The flowchart of the consumption phase.

Step 3. CA receives the message at T_{22}, then decrypts the cipher message by its private key:

$$M_{11} = D_{Prk_{CA}}(C_{RT_1}) \tag{96}$$

Then, CA checks for the validation of the timestamp:

$$(T_{22} - T_{21}) \stackrel{?}{\leq} \tau \tag{97}$$

Next, the CA calculates the hash value and executes the function "Verify" in Algorithm 2 to validate the signatures:

$$h_{RT_1}{}' = H(M_{11}) \tag{98}$$

$$Verify(h_{RT_1}{}', r_{RT_1}, s_{RT_1}) \tag{99}$$

If the signature is valid, CA executes a chaincode "Purchase", as shown in Algorithm 5, where a purchasing record is received and updated to the BCC. Then, CA selects a random number k_{12} and generates a message:

$$M_{12} = (ID_{CA} \parallel ID_{RT} \parallel List < PID > \parallel T_{23}) \tag{100}$$

Next, CA calculates the parameters and executes the function "Sign" to generate the signatures:

$$h_{CA_2} = H(M_{12}) \tag{101}$$

$$(r_{CA_2}, s_{CA_2}) = Sign(h_{CA_2}, k_{12}, d_{CA}) \tag{102}$$

A response message is encrypted with RT's public key:

$$C_{CA_2} = E_{Puk_{RT}}(M_{12}) \tag{103}$$

Then, CA sends the message $(ID_{\mathrm{CA}}, ID_{RT}, C_{CA_2}, (r_{CA_2}, s_{CA_2}))$ to LG.

Step 4. RT receives the message, then uses its private key to decrypt the cipher message:

$$M_{12} = D_{Prk_{\mathrm{RT}}}(C_{CA_2}) \tag{104}$$

Then, RT checks for the validation of the timestamp:

$$(T_{24} - T_{23}) \overset{?}{\leq} \tau \tag{105}$$

If the timestamp is valid, RT calculates the hash value and executes the function "Verify" to validate the signatures:

$$h_{CA_2}{}' = H(M_{12}) \tag{106}$$

$$Verify(h_{CA_2}{}', r_{CA_2}, s_{CA_2}) \tag{107}$$

If the signature is valid, then RT prints an invoice with the purchase information $List < (TID, \mathit{Price}, DateTime, PID, ID_{RT}) >$

Step 5. RT gives the printed invoice and tobacco products to the CS, then the transaction is completed.

Algorithm 5. Chaincode of consumption of the proposed scheme.

```
func Purchase (
ID_RT string, TIDs []string, Price []float, GPSLoc string) (Purchase_ID []string, Pay-
ment_DT []time.Time){

    for i:= 0 ; i < TIDs.Length ; i++ {
        index:= SearchTID(TIDs[i])
        TP[index].Retailer_ID = ID_RT
        TP[index].Payment_Datetime = time.Now()
        TP[index].Payment_GPSLocation = GPSLoc
        TP[index].Purchase_ID = GeneratePurchaseID()
        TP[index].Payment_Price = Price[i]
        Purchase_ID = append(Purchase_ID, TP[index].Purchase_ID)
        Payment_Datetime = append(Payment_Datetime, TP[index].Payment_DT)
    }
    return Purchase_IDs
}
```

3.9. Verification Phase (Consumer-End)

Sometimes consumers may doubt the legality of the tobacco products that they purchased from the retailer, so they can use the mobile application to check the legality of the

tobacco products. Figure 12 shows the consumer checking flowchart where the detailed steps are as follows:

Step 1. Consumers are required to provide the consumption information by mobile application such as the tobacco ID (TID), retailer ID, purchase ID, and the timestamp purchase that is printed in the invoice.

Step 2. The application executes a chaincode "GetTobaccoInfo", as shown in Algorithm 6.

Step 3. If the TID is valid, the BCC returns the tobacco detailed information. Otherwise, the BCC returns that the TID is not valid, so we can determine that this is an illegal tobacco product or trade.

Step 4. The mobile application shows the results of the chaincode, where consumers can obtain the legality and logistics status of the tobacco products.

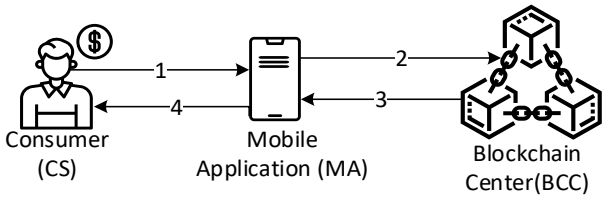

Figure 12. Consumer checking tobacco in the verification phase.

Algorithm 6. Chaincode verification of the proposed scheme.

```
func GetTobaccoInfo (TID string, RetailerID string, PurchaseID string, PurchaseDateTime time.Time) (result string){
        index:= SearchTID(TID)
        if index >= 0
        && TP[index]. Retailer_ID == RetailerID
        && TP[index]. Purchase_ID == PurchaseID
        && TP[index]. Payment_Datetime == PurchaseDateTime {
                result = TP[index]'s information
        }else{
                result = "Tobacco invalid"
        }
        return result

}

func Verification (string ID_AP, string TID, RoleType type) (Is_legal bool){
        index:= SearchTID(TID)
        if type == RoleType.Manufacturer {
                If ID_AP!= TP[index].Manufacturer_ID {
                        return false
                }
        }else{
                last_index:= len(TransportRecord)-1
                If ID_AP!= TP[index].TransportRecord[last_index].Recipient_ID {
                        return false
                }
        }
        return true
}
```

3.10. Verification Phase (Auditor-End)

In particular, government agencies will regularly check whether there are counterfeit tobacco products on the market that have not been approved or authenticated by the CA. Figure 13 shows the audit mechanism flow to verify the tobacco products. The detailed steps are as follows:

Step 1. AU uses the MR to scan the TID of tobacco products randomly from the company such as a retailer, logistics, distributor, or manufacturer.

Step 2. The application will send the selected tobacco ID, company ID, and its role type to CA.

Step 3. CA triggers a chaincode "Verification" in the BCC to verify the legality of tobacco products, where the chaincode is shown in Algorithm 6.

Step 4. BCC returns the legality result to the CA.

Step 5. CA responds to the result to the AU, if it is not legal, then the AU enforces the law on the illegal tobacco product and company.

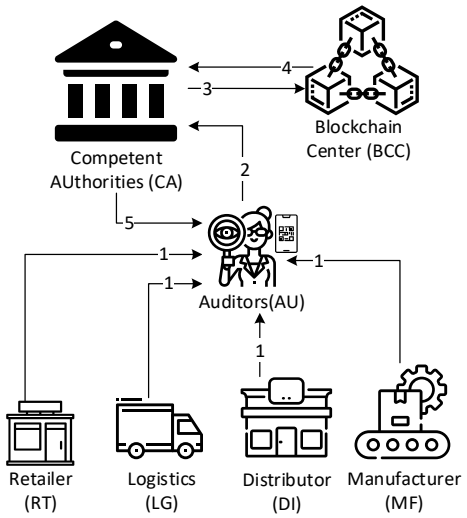

Figure 13. The audit mechanism in the verification phase.

3.11. Arbitration Phase

When any access party doubts the authenticity of the origin of the tobacco products, they can arbitrate its legality through a third-party arbitrator. The arbitration flow is shown in Figure 14. The detailed steps are described as follows:

Step 1. AP provides the specific tobacco products' TID to the AU.

Step 2. AU sends a request message with their signature and TID to the BCC.

Step 3. BCC checks the signature; if the signature is valid, then the AU responds to a list of signatures to the AU.

Step 4. The AU starts to check the validity of signatures.

a. If the RT signature is illegal, then the record is forged by the RT.
b. Then, the AU extracts the list of the "Transport Record" of tobacco information and obtains the RCP and SHP signature.
c. If the RCP signature is illegal, then the record is forged by the RCP.
d. If the SHP signature is illegal, then the record is forged by the SHP.
e. Confirm that the record is the last data of the list of "Transport Record". If it is not, go to step 4b, otherwise go to the next step.
f. If the MF signature is illegal, then the record is forged by the MF.
g. If there are no illegal signatures, then the logistics record is legal and verified by the AU.

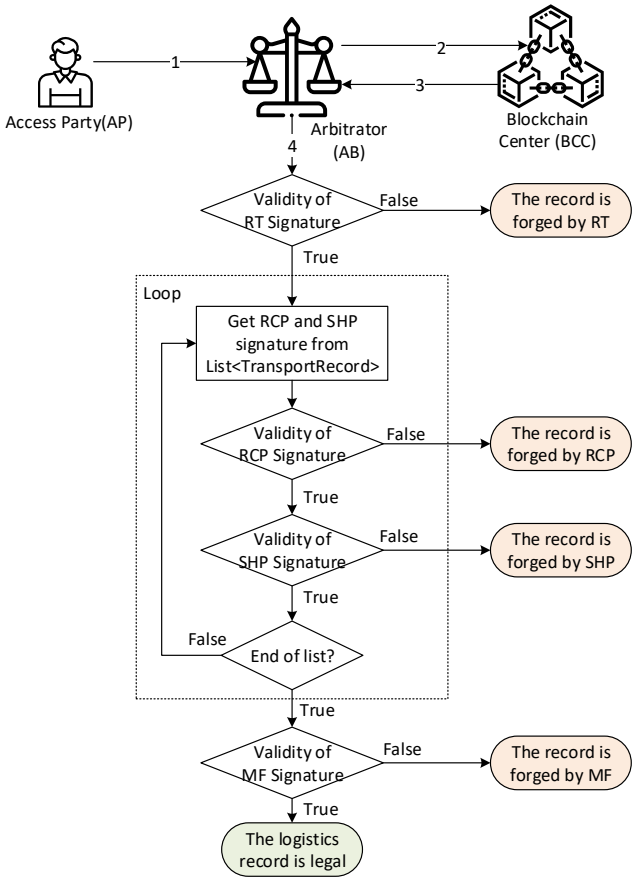

Figure 14. The validation flow in the arbitration phase.

4. Security Analysis

4.1. Mutual Authentication

We used BAN logic to verify the mutual authentication in the authentication phase. The notation of BAN logic is described below.

$\{X\}_K$	The message X is encrypted by a key K
$P \overset{SK}{\leftrightarrow} Q$	P and Q use a shared key SK to communicate
$P \mid\equiv X$	P believes X (belief rule)
$P \lhd X$	P sees X (seeing rule)
$P \mid\sim X$	P once said X (message meaning rule)
$P \mid\Rightarrow X$	P has jurisdiction over X (jurisdiction rule)
$\#(X)$	The message X is new (freshness rule)

The goals of the authentication analysis are:

G1: $A \mid\equiv A \overset{x_{A_1}}{\longleftrightarrow} B$

G2: $A \mid\equiv B \mid\equiv A \overset{x_{A_1}}{\longleftrightarrow} B$

G3: $B \mid\equiv A \overset{x_{B_1}}{\longleftrightarrow} B$

G4: $B \mid\equiv A \mid\equiv A \overset{x_{B_1}}{\longleftrightarrow} B$

G5: $A \mid\equiv ID_B$

G6: $A \mid\equiv B \mid\equiv ID_B$

G7: $B \mid\equiv ID_A$

G8: $B \mid\equiv A \mid\equiv ID_A$

According to the authentication algorithm, BAN logic was used to produce an idealized form as follows:

M1: $UserA \rightarrow UserB \quad (\{ID_A, ID_B, T_1\}_{Puk_B}, r_{A_1}, s_{A_1})$

M2: $UserB \rightarrow UserA \quad (\{ID_B, ID_A, T_3\}_{Puk_A}, r_{B_1}, s_{B_1})$

To analyze the proposed scheme, the following assumptions were made:

A1: $A \mid\equiv \#(T_1)$

A2: $B \mid\equiv \#(T_1)$

A3: $A \mid\equiv \#(T_3)$

A4: $B \mid\equiv \#(T_3)$

A5: $A \mid\equiv B \mid\Rightarrow B \xleftrightarrow{x_{B_1}} A$

A6: $B \mid\equiv A \mid\Rightarrow A \xleftrightarrow{x_{A_1}} B$

A7: $A \mid\equiv B \mid\Rightarrow ID_B$

A8: $B \mid\equiv A \mid\Rightarrow ID_A$

a. User B authenticates user A

By *M1* and the seeing rule, derive:

$B \lhd (\{ID_A, ID_B, T_1\}_{Puk_B}, r_{A_1}, s_{A_1})$ (*Statement 1*)

By *A2* and the freshness rule, derive:

$B \mid\equiv \#(\{ID_A, ID_B, T_1\}_{Puk_B}, r_{A_1}, s_{A_1})$ (*Statement 2*)

By (*Statement 1*) and the message meaning rule derive:

$B \mid\equiv A \mid\sim (ID_A, ID_B, T_1, r_{A_1}, s_{A_1})$ (*Statement 3*)

By (*Statement 2*), (*Statement 3*) and the nonce verification rule, derive:

$B \mid\equiv A \mid\equiv (ID_A, ID_B, T_1, r_{A_1}, s_{A_1})$ (*Statement 4*)

By (*Statement 4*) and the belief rule, derive (G4):

$B \mid\equiv A \mid\equiv A \xleftrightarrow{x_{A_1}} B$ (*Statement 5*)

By (*Statement 5*), *A6,* and the jurisdiction rule, derive (G3):

$B \mid\equiv A \xleftrightarrow{x_{A_1}} B$ (*Statement 6*)

By (*Statement 4*) and the belief rule, derive (G8):

$B \mid\equiv A \mid\Rightarrow ID_A$ (*Statement 7*)

By (*Statement 7*), *A8,* and the belief rule, derive (G7):

$B \mid\equiv ID_A$ (*Statement 8*)

b. User A authenticates user B

By *M1* and the seeing rule, derive:

$A \lhd (\{ID_B, ID_A, T_3\}_{Puk_A}, r_{B_1}, s_{B_1})$ (*Statement 9*)

By *A3* and the freshness rule, derive:

$A \mid\equiv \#(\{ID_B, ID_A, T_3\}_{Puk_A}, r_{B_1}, s_{B_1})$ (*Statement 10*)

By (*Statement 9*) and the message meaning rule derive:

$A \mid\equiv B \mid\sim (ID_B, ID_A, T_3, r_{B_1}, s_{B_1})$ (*Statement 11*)

By (*Statement 10*), (*Statement 11*) and the nonce verification rule, derive:

$A \mid\equiv B \mid\equiv (ID_B, ID_A, T_3, r_{B_1}, s_{B_1})$ (*Statement 12*)

By (*Statement 12*) and the belief rule, derive (G2):

$A \mid\equiv B \mid\equiv B \xleftrightarrow{x_{B_1}} A$ (*Statement 13*)

By (*Statement 13*), *A5* and the jurisdiction rule, derive (G1):

$A \mid\equiv B \xleftrightarrow{x_{B_1}} A$ (*Statement 14*)

By (*Statement 12*) and the belief rule, derive (G6):

$A \mid\equiv B \mid\Rightarrow ID_B$ (*Statement 15*)

By (*Statement 15*), *A7*, and the belief rule, derive (G5):

$A \mid\equiv ID_B$ (*Statement 16*)

By (*Statement 6*), (*Statement 8*), (*Statement 14*), and (*Statement 16*), these statements authenticate the identities of user A and user B mutually in the proposed scheme.

4.2. Unforgeable Record

We implemented a blockchain-based system in the proposed scheme shown in Figure 15. Each CA had a ledger, and the blocks in the ledger were synchronized with the PBFT consensus algorithm. When an accessing party executes a chaincode function, the modification of the chaincode will be chained to the ledger. The latest ledger will be validated from every CA department that participates in the blockchain center. It is hard to modify and forge the data in the ledger.

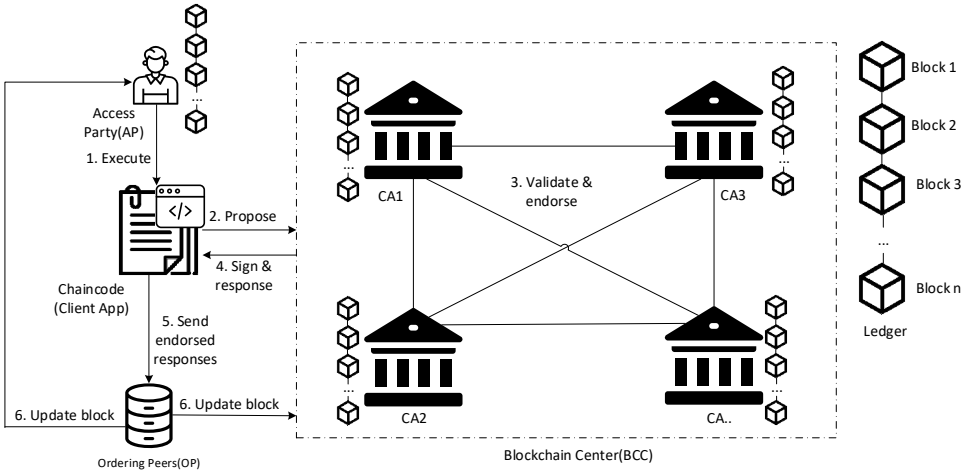

Figure 15. The mechanism of the blockchain architecture.

Furthermore, we designed the structure of data as shown in Figures 4 and 5. The structure of the chaincode stores the complete information of tobacco including the manufacturer and logistics record with GPS location. Therefore, all information and the logistics records of tobacco can be tracked. Consequently, the logistics records cannot be forged and can be tracked correctly in our blockchain-based system.

4.3. Non-Repudiation

In the proposed scheme, we used ECDSA to achieve the repudiation issues. Every message transmitted by the sender must sign with their private key, then the receiver can be verified using its public key. We have sorted a list of verification equations that need to be verified by the receiver in every phase, as shown in Table 2.

Table 2. Non-repudiation's verification of the proposed scheme.

Phase	Party		Verification Function
	Sender	Receiver	
Authentication Phase	A	B	$Verify(h_{A_1}', r_{A_1}, s_{A_1})$
	B	A	$Verify(h_B', r_{B_1}, s_{B_1})$

Issuing and Manufacture Phase	MF	IDI	$Verify(h_{MF_1}{}', r_{MF_1}, s_{MF_1})$
	IDI	CA	$Verify(h_{IDI_1}{}', r_{IDI_1}, s_{IDI_1})$
	CA	IDI	$Verify(h_{CA_1}{}', r_{CA_1}, s_{CA_1})$
	IDI	MF	$Verify(h_{IDI_2}{}', r_{IDI_2}, s_{IDI_2})$
Logistics Phase (Shipping)	SHP	LG	$Verify(h_{SHP_1}{}', r_{SHP_1}, s_{SHP_1})$
	LG	SHP	$Verify(h_{LG_1}{}', r_{LG_1}, s_{LG_1})$
Logistics Phase (Receiving)	LG	RCP	$Verify(h_{LG_2}{}', r_{LG_2}, s_{LG_2})$
	RCP	LG	$Verify(h_{RCP_1}{}', r_{RCP_1}, s_{RCP_1})$
Consumption Phase	RT	CA	$Verify(h_{RT_1}{}', r_{RT_1}, s_{RT_1})$
	CA	RT	$Verify(h_{CA_2}{}', r_{CA_2}, s_{CA_2})$

4.4. Integrity

To ensure the integrity of data when communicating between the sender and receiver, we used the hash function to hash data in the signature value. The signatures with the hash value of all the phases are listed in Table 3, for example, in the issuing and manufacture phase, the MF signs the signature s_{MF_1} with a hash value h_{MF_1}. All the signatures were calculated with the ECDSA and verified by the receiver, as shown in Table 2.

Table 3. Non-repudiation's verification of the proposed scheme.

Phase	Party		Signature
	Sender	Receiver	
Authentication Phase	A	B	$(r_{A_1}, s_{A_1}) = Sign(h_{A_1}, k_1, d_A)$
	B	A	$(r_{B_1}, s_{B_1}) = Sign(h_{B_1}, k_2, d_B)$
Issuing and Manufacture Phase	MF	IDI	$(r_{MF_1}, s_{MF_1}) = Sign(h_{MF_1}, k_3, d_{MF})$
	IDI	CA	$(r_{IDI_1}, s_{IDI_1}) = Sign(h_{IDI_1}, k_4, d_{IDI})$
	CA	IDI	$(r_{CA_1}, s_{CA_1}) = Sign(h_{CA_1}, k_5, d_{CA})$
	IDI	MF	$(r_{IDI_2}, s_{IDI_2}) = Sign(h_{IDI_2}, k_6, d_{IDI})$
Logistics Phase (Shipping)	SHP	LG	$(r_{SHP_1}, s_{SHP_1}) = Sign(h_{SHP_1}, k_7, d_{SHP})$
	LG	SHP	$(r_{LG_1}, s_{LG_1}) = Sign(h_{LG_1}, k_8, d_{LG})$
Logistics Phase (Receiving)	LG	RCP	$(r_{LG_2}, s_{LG_2}) = Sign(h_{LG_2}, k_9, d_{LG})$
	RCP	LG	$(r_{RCP_1}, s_{RCP_1}) = Sign(h_{RCP_1}, k_{10}, d_{RCP})$
Consumption Phase	RT	CA	$(r_{RT_1}, s_{RT_1}) = Sign(h_{RT_1}, k_{11}, d_{RT})$
	CA	RT	$(r_{CA_2}, s_{CA_2}) = Sign(h_{CA_2}, k_{12}, d_{CA})$

4.5. Resist Known Attacks

4.5.1. Replay Attack

To resist replay attack, every encrypted message is added to a sending timestamp and check to see whether the timespan is valid. The timestamp is validated in every phase to resist replay attack, and the validations are shown in Equations (9), (22), (33), (41), (49), (57), (65), (73), (81), (89), (97), and (105). For example, in the issuing ID and manufacture

phase, MF adds a timestamp T_5 in the message M_3, then encrypts the M_3 with a cipher message C_{MF_1}. The IDI decrypts the message after receiving the cipher message, then checks for the validation of the timestamp. The related equations are shown as follows:

$$M_3 = (ID_{MF} \,||\, ID_{IDI} \,||\, Num \,||\, Info \,||\, T_5) \quad (108)$$

$$C_{MF_1} = E_{Puk_{IDI}}(M_3) \quad (109)$$

$$M_3 = D_{Prk_{IDI}}(C_{MF_1}) \quad (110)$$

$$(T_6 - T_5) \stackrel{?}{\leq} \tau \quad (111)$$

Scenario: The attacker listens to a message that is sent from the sender to the receiver. After that, the attacker re-sends the same message to the receiver to achieve a replay attack.

Analysis: The receiver decrypts and obtains the timestamp in the message, then subtracts the timestamp with the current time; if the timespan is larger than the threshold, it means that the message is a replay attack, so the attack is foiled.

4.5.2. Main-in-the-Middle Attack (MITM)

The MITM is an attack where an attacker will be in the middle of the sender and receiver, listening or modifying the messages between each other. In the method, we added an encryption and decryption mechanism to the communication protocol. These encrypted and decrypted scenarios are shown in Equations (7), (8), (20), (21), (31), (32), (39), (40), (47), (48), (55), (56), (63), (64), (71), (72), (79), (80), (87), (88), (95), (96), (103), and (104). For example, in the logistics shipping phase, SHP encrypts the message M_7 to a cipher message C_{SHP_1} with LG's public key Puk_{LG}. The LG decrypts the message after receiving the cipher message with their private key Prk_{LG}. The related equations are shown as follows:

$$C_{SHP_1} = E_{Puk_{LG}}(M_7) \quad (112)$$

$$M_7 = D_{Prk_{LG}}(C_{SHP_1}) \quad (113)$$

Scenario: The attacker eavesdrops or tampers with the communication messages between the sender and receiver and analyzes the content.

Analysis: The message is encrypted with the receiver's public key, and the receiver must have a relevant private key to decrypt the message. However, the attacker did not have the private key of the receiver, so they are unable to decrypt the message to learn the content of the transmission.

5. Discussion

5.1. Computation Cost

First, we analyzed the computational costs of each phase. We used asymmetrical encryption/decryption, hash functions, addition, subtraction, multiplication, and division operation as the basis for calculating the costs. The costs of each phase are shown in Table 4.

Table 4. Computation costs of the proposed scheme.

Phase	1st Role	2nd Role	3rd Role
Authentication Phase	User A: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 4T_{mul} + 3T_{div}$	User B: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 3T_{mul} + 3T_{div}$	N/A
Issuing and Manufacture Phase	Manufacturer: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 4T_{mul} + 3T_{div}$	IDI Issuer: $4T_{asy} + 4T_h + 4T_{add} + 2T_{sub} + 6T_{mul} + 6T_{div}$	Competent Authorities: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 3T_{mul} + 3T_{div}$
Logistics Phase (Shipping)	Shipper: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 4T_{mul} + 3T_{div}$	Logistics: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 3T_{mul} + 3T_{div}$	N/A
Logistics Phase (Receiving)	Logistics: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 4T_{mul} + 3T_{div}$	Recipient: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 3T_{mul} + 3T_{div}$	N/A

Consumption Phase	Consumer: N/A	Retailer: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 4T_{mul} + 3T_{div}$	Competent Authorities: $2T_{asy} + 2T_h + 2T_{add} + 1T_{sub} + 3T_{mul} + 3T_{div}$

Notes: T_{asy}: asymmetrical encryption/decryption; T_h: a hash operation; T_{add}: an additional operation; T_{sub}: a subtraction operation. T_{mul}: a multiplication operation; T_{div}: a division operation; T_{exp}: an exponential operation

5.2. Communication Performance

In Table 5, the communication performance was analyzed within every phase. The maximum transmission speed is 100 Mbps in a 4G environment, and the maximum transmission speed is 20 Gbps in a 5G environment [33].

Table 5. Communication costs of the proposed scheme.

	Message Length	4G (100 Mbps)	5G (20 Gbps)
Authentication Phase	3648 bits	36 μs	0.18 μs
Issuing and Manufacture Phase	7296 bits	71 μs	0.36 μs
Logistics Phase (Shipping)	3648 bits	36 μs	0.18 μs
Logistics Phase (Receiving)	3648 bits	36 μs	0.18 μs
Consumption Phase	3648 bits	36 μs	0.18 μs

Notes: L_{ID}: an ID (144 bits); L_m: a cipher message (512 bits); L_{sig}: signature parameters (1024 bits).

In our analysis, it was assumed that an ID message required 144 bits, a cipher message required at least 512 bits, and a signature message required 1024 bits. We assumed 512 bits of a cipher message by the minimum length of the message (2 IDs, 1 Timestamp, and 2 others) that is sent at the issuing ID and manufacture phase, which is 2 × 144 bits + 1 × 64 bits + 2 × 80 bits = 512 bits.

With the above definition, we used the transmitted message as the calculated communication cost. For example, in the logistics shipping phase, SHP sends two IDs, one cipher message, one signature to LG, then LG sends two IDs, one cipher message, and one signature to SHP for a response message. In total, it requires 4 × 144 bits + 2 × 512 bits + 2 × 1024 bits. In a 4G environment, it only takes 36 μs to transfer all the messages. In a 5G environment, the transmission time only needs 0.18 us to complete the authentication phase.

5.3. Comparison

We used the EU's final report [5] as the target of comparison, and Table 6 shows the comparison between our scheme and the EU's tobacco products logistics tracking system.

Table 6. Comparison of the tobacco products logistics system.

	EU's Scheme [5]	Our Scheme
Database	General database	Blockchain-based Decentralized ledger
Flexibility	Low	High
Data integrity	Low	High
Message non-repudiation	N/A	High
Traceable record	Yes	Yes
Verifiable record	No	Yes
Tampering data	Easy	Hard
Tracking record establishment	Pre-set routing	Update routing dynamically
Security analysis	Yes	Yes

The EU has proposed a scheme with a general database with the defined field. The flexibility of the general database is very low, because the predefined field is fixed in ad-

vance, and the query connection between multiple tables is hard to modify when the system needs to upgrade. Our scheme used a blockchain-based decentralized ledger, so the blockchain has the characteristics of high flexibility and data integrity. By the way, we used the signature mechanism to ensure the message non-repudiation issue.

Additionally, both systems can trace the data record. However, our scheme provides verifiability between the records as the blockchain system has a strong connection between every record with blocks, and the newly added block contains the hash value of the previous block to ensure that the data are not easily tampered with.

According to the EU's scheme, they need to set up the routing path before generating the tobacco ID. Compared to the EU scheme, our method can update the routing path dynamically via the logistics phase, where every record is appended to the chaincode's tobacco information. Therefore, the manufacturer, distributor, or retailer can transport the tobacco products more flexibly.

To sum up, our blockchain-based method can improve the security and integrity of data and provide greater flexibility in updating logistics information.

6. Conclusions

We proposed a traceable and verifiable blockchain-based logistics system for tobacco products. We also proposed the chaincode algorithm and complete system architecture was raised in the scheme. Every tobacco product's logistics record can be scanned and recorded to the blockchain center via a mobile reader, so the authorized department or consumer can easily track the logistics of tobacco products. If there is any doubt about the tobacco products, any party is able to send a request to a third-party arbitrator to check which part in the supply chain is illegal.

For the security of our system, we applied the ECDSA in the communication protocol to achieve a more secure system. The security of the system was analyzed such as mutual authentication, unforgeable data, non-repudiation, and integrity. In particular, for mutual authentication, we used BAN logic to prove that the communication was secure. Furthermore, the proposed protocol also prevents replay attacks and man-in-the-middle attacks, and it is hard to attack our system with validation of the timestamp and encryption/decryption of communication messages.

We also compared our proposed scheme with the EU's scheme. The EU provides a general framework for the system, whereas our scheme focused more on how to realize the system with blockchain-based, GPS and RFID technologies. In the comparison, we found that our system had higher flexibility and was more secure.

To sum up, our research achieved the following contributions:

(1) Clarifies the Hyperledger Fabric architecture and applies the blockchain technology to the field of transporting tobacco products.
(2) The proposed blockchain-based system synchronizes the logistics record in the ledger via a secure private channel. Tobacco products are hard to forge or smuggle by a malicious party.
(3) A verification phase was designed to provide consumers and auditors with the ability to check the legality of tobacco products. When illegal tobacco products are found, the access party can raise any doubts and the arbitrator can find out which party acted illegally in the logistics process.
(4) Compared with the EU's scheme, the access party can scan the TID and update to the blockchain during the logistics phase, so the mechanism provides a flexible way to update the logistics records to the blockchain.
(5) Provide security verification and simulation against known threats or attacks.
(6) Use BAN Logic to prove the security of mutual authentication.

This research proposes a system architecture that uses blockchain, GPS and RFID technologies to trace logistics records. This structure is also applicable to the application example of information tracking of all enterprise organizations and alliance chains.

Author Contributions: The authors' contributions are summarized below. Z.-Y.L. and C.-L.C. made substantial contributions to the conception and design. Z.-Y.L. and C.-L.C. were involved in drafting the manuscript. Z.-Y.L. and Y.-Y.D. acquired data and analyzed and conducted the interpretation of the data. The critically important intellectual content of this manuscript was revised by H.-C.L., Y.-Y.D., and P.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 61801 413).

Institutional Review Board Statement: This study was based entirely on theoretical basic research and did not involve humans.

Informed Consent Statement: This study was based entirely on theoretical basic research and did not involve humans.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. CTP Glossary|FDA—Tobacco Product. Available online: https://www.fda.gov/tobacco-products/compliance-enforcement-training/ctp-glossary#:~:text=Tobacco%20product%20%2D%20 (accessed on 12 March 2021).
2. WHO—Tobacco. Available online: https://www.who.int/en/news-room/fact-sheets/detail/tobacco (accessed on 12 March 2021).
3. *WHO Framework Convention on Tobacco Control*; World Health Organization: Geneva, Switzerland, 2003.
4. Joossens, L.; Raw, M. From cigarette smuggling to illicit tobacco trade: Table 1. *Tob. Control.* **2012**, *21*, 230–234, doi:10.1136/tobaccocontrol-2011-050205.
5. *Implementation Analysis Regarding the Technical Specifications and Other Key Elements for a Future EU System for Traceability and Security Features in the Field of Tobacco Products: Final Report*; Consumers, Health, Agriculture and Food Executive Agency, Health Unit: Luxembourg, 2018; p. 114.
6. Products, Guidance & Regulations|FDA. Available online: https://www.fda.gov/tobacco-products/products-guidance-regulations (accessed on 12 March 2021).
7. GS1 Barcodes—Standards. Available online: https://www.gs1.org/standards/barcodes (accessed on 12 March 2021).
8. Azevedo, S.G.; Carvalho, H. Contribution of RFID technology to better management of fashion supply chains. *Int. J. Retail. Distrib. Manag.* **2012**, *40*, 128–156, doi:10.1108/09590551211201874.
9. Wyld, D.C. Death sticks and taxes: RFID tagging of cigarettes. *Int. J. Retail. Distrib. Manag.* **2008**, *36*, 571–582, doi:10.1108/09590550810880598.
10. Kamilaris, A.; Fonts, A.; Prenafeta-BoldúF.X. The rise of blockchain technology in agriculture and food supply chains. *Trends Food Sci. Technol.* **2019**, *91*, 640–652, doi:10.1016/j.tifs.2019.07.034.
11. Wang, S.; Tao, Y.; Wang, G. UHF RFID Tag for Integration into a Cigarette Pack. *IEEE Antennas Wirel. Propag. Lett.* **2011**, *10*, 1433–1436, doi:10.1109/LAWP.2011.2179279.
12. Shi, J.; Li, Y.; He, W.; Sim, D. SecTTS: A secure track & trace system for RFID-enabled supply chains. *Comput. Ind.* **2012**, *63*, 574–585, doi:10.1016/j.compind.2012.03.006.
13. Prasanna, K.; Hemalatha, M. RFID GPS and GSM based logistics vehicle load balancing and tracking mechanism. *Procedia Eng.* **2012**, *30*, 726–729, doi:10.1016/j.proeng.2012.01.920.
14. Li, Y.; Xu, L.; Zhao, S. Tobacco Logistics Retroactive System Research Based on RFID Technology. *Internet Things Cloud Comput.* **2016**, *4*, 39, doi:10.11648/j.iotcc.20160404.11.
15. Liu, H.; Li, Z.; Cao, N. Framework Design of Financial Service Platform for Tobacco Supply Chain Based on Blockchain. In Proceedings of the International Conference on Algorithms and Architectures for Parallel Processing, Guangzhou, China, 15–17 November 2018; pp. 145–150.
16. Humayun, M.; Jhanjhi, N.; Hamid, B.; Ahmed, G. Emerging Smart Logistics and Transportation Using IoT and Blockchain. *IEEE Internet Things Mag.* **2020**, *3*, 58–62, doi:10.1109/iotm.0001.1900097.
17. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2019. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 11 February 2012).
18. Buterin, V. A next-generation smart contract and decentralized application platform. *Ethereum White Paper* **2014**, *3*, 36.
19. Wang, Y.-C.; Chen, C.-L.; Deng, Y.-Y. Authorization Mechanism Based on Blockchain Technology for Protecting Museum-Digital Property Rights. *Appl. Sci.* **2021**, *11*, 1085, doi:10.3390/app11031085.
20. Chen, C.-L.; Deng, Y.-Y.; Weng, W.; Zhou, M.; Sun, H. A blockchain-based intelligent anti-switch package in tracing logistics system. *J. Supercomput.* 2021, 1–42, doi:10.1007/s11227-020-03558-7.
21. Wang, Y.-C.; Chen, C.-L.; Deng, Y.-Y. Museum-Authorization of Digital Rights: A Sustainable and Traceable Cultural Relics Exhibition Mechanism. *Sustainability* **2021**, *13*, 2046, doi:10.3390/su13042046.

22. Chen, C.-L.; Lin, C.-Y.; Chiang, M.-L.; Deng, Y.-Y.; Chen, P.; Chiu, Y.-J. A Traceable Online Will System Based on Blockchain and Smart Contract Technology. *Symmetry* **2021**, *13*, 466, doi:10.3390/sym13030466.
23. Chen, C.-L.; Chiang, M.-L.; Deng, Y.-Y.; Weng, W.; Wang, K.; Liu, C.-C. A Traceable Firearm Management System Based on Blockchain and IoT Technology. *Symmetry* **2021**, *13*, 439, doi:10.3390/sym13030439.
24. Kwak, K.H.; Kong, J.T.; Cho, S.I.; Phuong, H.T.; Gim, G.Y. A Study on the Design of Efficient Private Blockchain. In *Artificial Intelligence: Foundations, Theory, and Algorithms*; Springer Science and Business Media LLC: Berlin/Heidelberg, Germany, 2018; pp. 93–121.
25. Li, Z.; Kang, J.; Yu, R.; Ye, D.; Deng, Q.; Zhang, Y. Consortium Blockchain for Secure Energy Trading in Industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2017**, *14*, 1, doi:10.1109/tii.2017.2786307.
26. Mingxiao, D.; Xiaofeng, M.; Zhe, Z.; Xiangwei, W.; Qijun, C. A review on consensus algorithm of blockchain. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 2567–2572.
27. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; Caro, A.D.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference, Porto, Portugal, 23–26 April 2018.
28. How Fabric Networks Are Structured—Hyperledger-Fabricdocs Main Documentation. Available online: https://hyperledger-fabric.readthedocs.io/en/latest/network/network.html (accessed on 30 April 2021).
29. Vanstone, S. Responses to NIST's proposal. *Commun. ACM* **1992**, *35*, 50–52.
30. Johnson, D.; Menezes, A.; Vanstone, S. The Elliptic Curve Digital Signature Algorithm (ECDSA). *Int. J. Inf. Secur.* **2001**, *1*, 36–63, doi:10.1007/s102070100002.
31. Elaine Barker, A.R. *Transitioning the Use of Cryptographic Algorithms and Key Lengths*; National Institute of Standards and Technology, Computer Security Division, Information Technology Laboratory: Gaithersburg, MD, USA, 2019.
32. Burrows, M.; Abadi, M.; Needham, R.M. A logic of authentication. *ACM Trans. Comput. Syst.* **1990**, *8*, 18–36, doi:10.1145/77648.77649.
33. Kaur, K.; Kumar, S.; Baliyan, A. 5G: A new era of wireless communication. *Int. J. Inf. Technol.* **2020**, *12*, 619–624, doi:10.1007/s41870-018-0197-x.