Blockchain-based Supply Chain System for Traceability, Regulation and Anti-counterfeiting

Wang Fat Lau
Department of Computing
The Hong Kong Polytechnic University
Hong Kong, China
wf-franky.lau@connect.polyu.hk

Dennis Y. W. Liu
Department of Computing
The Hong Kong Polytechnic University
Hong Kong, China
csdennis@comp.polyu.edu.hk

Man Ho Au
Department of Computer Science
The University of Hong Kong
Hong Kong, China
allenau@cs.hku.hk

***Abstract*—Supply chain management is essential to regulated industries, e.g., food and drug manufacturing. Due to the complexity of the logistic and manufacturing procedures in reality, regulation enforcement and anti-counterfeiting is hard to achieve. To solve this issue, we present a blockchain-based supply chain system which the design is based on the logistic behavioral patterns in the literature, as well as the real-world settings of the industries. Our system had deployed to existing drugs manufacturers and wholesalers in Chain, however, it is possible to apply on any industry. Our design also incorporates digital identity management and quality monitoring, which are the essential services of supply chain management systems. We also analyze the efficiency of our implementation on a typical cloud platform. Our system is efficient and practical for real deployment based on our experimental results.**

***Index Terms*—blockchain; supply chain; traceability; quality management; anti-counterfeiting**

I. Introduction

Due to globalization, manufacturing process becomes more complex nowadays. The study of supply chain management system has gained considerable amount of interest over the years. This is especially the case in industries where there are extensive regulatory requirements from government such as the food and drug manufacturing industries, since it may cause huge impact on human life. However, due to the complex logistic flows in the supply chain, it dramatically increases the difficultly for government to enforces regulation and for customers to determine counterfeit goods. Thus, supply chain management system is essential to providing traceability for the products and accountability for the manufacturing and logistic processes.

In supply chain management, one of the largest challenge is the communications among a huge number of stakeholders, including suppliers, manufacturers, wholesalers, distributors and retailers. An overview of a supply chain with regulatory compliance is illustrated in Fig. 1. This kind of large-scale inter-organizational systems (IOSs) was originally considered as a market strategic decision for reducing transaction cost [1]. Choudhury [2] investigated the problem and classified three patterns of IOSs. Based on this, Benchini et al. [3] modeled the behavioral patterns and proposed a supply chain traceability system based on Electronic Data Interchange (EDI). However, when the system is designed for achieving cost efficiency, the traceability is highly dependent on the trust of all participants. As such, there is a risk of data being tampered during investigation by regulation authority.

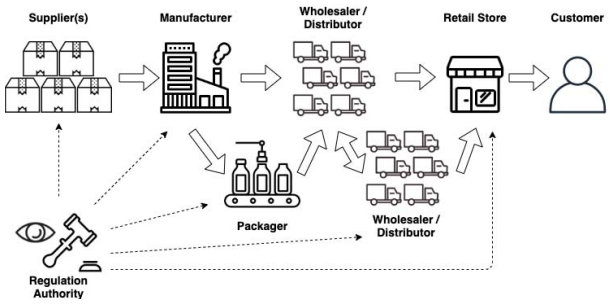

Fig. 1 An overview of a supply chain with regulatory compliance

With the advancement of the Internet and wireless communication technologies, the Internet of Things (IoT) is being adopted in supply chain systems [4]. Numerous sensors are installed to cargos and attached to products for keeping track the locations and environment information, such as temperature and humidity, during transportation. Data are sent back to computer servers periodically to improve the accuracy and quality for realtime product tracking and monitoring. Nevertheless, the data integrity issue in supply chain system is still not being properly addressed and it is left as an open problem to the industry.

Blockchain-based supply chain systems are being proposed to address the issue. According to a recent work [5], blockchain-based solutions have a positive impact on the customer trust, especially on unfamiliar retailers. As a new distributed ledger technology, blockchain is known for its decentralization and immutability properties. As such, it appears to be an effective solution for regulation authorities and general customers to detect data tampering in the above-mentioned data integrity issue in supply chain systems. Early academic efforts [6], [7] discussed how blockchain can be applied to food and drug traceability. However, these proposals are at the conceptual state and its efficiency are evaluated based on simulations or prototyping. Recently, comprehensive solutions [8]–[10] were developed. However, these solutions focus mainly on products tracking and do not consider other essential

978-1-6654-1760-0/21/$31.00 ©2021 IEEE
DOI 10.1109/Blockchain53845.2021.00022

features [11] of practical relevance, such as digital identity management and quality monitoring. Moreover, these solutions do not consider the existing logistic behavioral patterns of real-world scenarios, where supply chain flows form a complex directed acyclic graph (DAG). To the best of our knowledge, only [7] and [9] in the existing blockchain-based solutions consider a supply chain with a DAG structure. Besides, none of them explicitly implements multi-hop tracing and considers how to trace product without unique identifier (UID). These features are essential in anti-counterfeiting scenarios in reality.

We worked closely with drug manufacturers and wholesalers in China to develop a blockchain-based supply chain system for real-world usage. We realize that the common assumption in existing solutions may not apply to drug manufacturing. Specifically, many existing solutions assume the existence of a UID for each product, which is often unavailable in practice. The Chinese government has strict regulations on the production of packing materials of drug products [12]. Each modification of the packing contents, including description text, batch number, serial number and expiry date, requires an explicit approval by relevant authorities. Indeed, the smallest unit of "tracking" is often a product batch (a product batch is a group of identical items produced together; each batch goes through the same stages in the production process). In case of fault, it is often the smallest unit of recall. Besides the pharmaceutical industry, we note that batch production is also common in the production of electrical goods, clothing, and fast food, etc. Furthermore, there are numerous procedures and regulations [13], [14] for the practitioners to follow during the production and logistic processes. A robust system for such settings is required to record all essential activities, timestamps and participants for auditing purposes. However, there is no existing solution considering this scenario.

In this paper, to address the above-mentioned issues, we propose a generic blockchain-based supply chain management system for traceability, regulation and anti-counterfeiting. Specifically, the system provides the following features:

- **Traceability**: We identify the real-world requirements from drug industries in China and propose a generic design for supply chain systems based on blockchain technology. Our system supports four common logistic behavioral patterns and supports efficient multi-hop traceability which is new to existing solutions. As an IOS which is hard to consolidate participants during adoption, we define common algorithms and standard interfaces of smart contracts for data interchange.
- **Accountable**: All logistic activities can be tracked in the system. We further include identity management to the system, where the authenticity of transactions can be ensured. Procedure logs of production is recorded for auditing purpose. Unlike existing solutions, message authentication in our design is not bound to company level, but it is down to employee level, which is more realistic in the auditing processes.
- **Integrity**: We deploy our system on blockchain, which ensures data immutability and integrity. It can prevent data tampering on product manufacturing logs and logistic information.
- **Verifiability and Transparency**: All logistic data and manufacturing procedure logs are recorded in blockchain. The transaction data and signatures in blockchain are publicly verifiable, which also increases the trust among the stakeholders in the supply chain and reduces the efforts on regulation enforcement.
- **Efficient**: We employ a multi-layer blockchain architecture, which reduces the complexity of consensus among nodes, and to improve the overall efficiency of the system. Also, we develop tracing algorithms with logarithmic complexity, which allow customers to perform efficient origin tracing.

II. Related Work

The history of supply chain management system can be traced back to 1985 when Cash [15] reviewed on how inter-organizational systems (IOSs) influenced the manufacturing industry. In its early stage, supply chain management systems are designed for reducing transaction cost in the value-added chain of manufacturing [1]. In 1997, Choudhury [2] proposed the three common patterns of IOSs, namely, electronic monopolies, multilateral IOSs and electronic dyads. The electronic dyads pattern is a type of peer-to-peer (P2P) network, where each organization directly connects with each of its business partner through independent electronic links. It is common that Electronic Data Interchange (EDI) is adopted as a message standard in electronic dyads.

The word traceability is added into ISO8420:1994 [16] for quality management and quality assurance. In 2002, van Dorp [17] summarized the development of existing traceability systems. According to van Dorp, traceability can be defined into the ability of $tracking$ and $tracing$. In 2008, Benchini et al. [3] proposed a new supply chain system and modeled the behavioral patterns in logistics. Since then, many works [18], [19] focused on the food and drug supply chain system. Recently, due to the improvement in wireless communication, radio frequency identification (RFID) and the Internet of Things (IoT) are introduced to supply chain systems [4], [19], [20] as well. Although there are more data to assist the quality management in supply chain system, there is a trust issue on these centralized systems. Data tampering can be easily conducted if a malicious organization wants to circumvent government regulations.

Recently, blockchain-based supply chain systems were proposed to address this issue. In 2017, Chen et al. [11] proposed a blockchain-based supply chain quality management framework. They described different modules required in the system, including digital identity management, quality monitoring and control, logistics planning and demand analysis. In 2018, Hua et al. [6] proposed a system for agricultural products. However, these systems remain conceptual and no actual implementation is conducted. Drugledger [7] is a blockchain-based system, extended from Bitcoin [21]. The system provides traceability and identity management. However, its design relies on the

unspent transaction output (UTXO) model and the proposed data structure does not cater for tracking the internal process of the organization down to the employee level. Since the personnel in the manufacturing process may change, it is hard to define the owner of a transaction output. In 2019, Aniello et al. [22] proposed to adopt physically unclonable function (PUF) in blockchain-based supply chain system for anti-counterfeiting purpose. This technique requires special hardware, which may hinder its deployment in practice. More recently, a blockchain token based solution was proposed [9]. It used an Ethereum [23] token standard (ERC721 [24]), known as Non-Fungible Token (NFT), to keep track of various logistic activities. Nevertheless, ERC721 standard is originally intended to record asset ownership only, the design added quantity to it, which violates the original intention, non-fungibility, of the token. PharmaCrypt [8] is another recent blockchain-based drug supply chain system. It requires the manufacturer to scan barcode for each product and upload the data of the shipping box to blockchain. This process is inherently inefficient and not practical when the batch size is large. In 2021, Musamih et al. [10] proposed another Ethereum-based solution with implementation details, with testing and validation. However, both [8] and [10] did not consider all logistic behavioral patterns in their design, hereby could not handle integration and division lots where it is very common in reality. Currently, most of the existing blockchain-based solutions are still in prototyping phase and most of them focus on traceability only, with less efforts spent on other essential features, like identity management and quality monitoring, for supply chain systems.

III. Preliminary

A. Supply Chain System

In a supply chain management system, traceability of products and its activities are the essential requirements [3]. The tracing unit is commonly called *lot*, which can be a single unit of product or a batch of products. Traceability can be further divided into the abilities of lot tracking and lot tracing. *Tracking* is the ability of keeping track of the flows of lots transporting from upstream to downstream in a supply chain. It is especially important during products recall when a fault in the manufacturing process is discovered. On the other hand, *tracing* refers to the ability to follow the supply chain upward and determine the source of a lot. This enables customers to have capability to distinguish counterfeit products. The flows of tracking and tracing are shown in Fig. 2.

The basic lot behaviors can be modeled by four patterns [25], namely, *integration*, *division*, *alteration* and *movement*. They are illustrated in Fig. 3 and described below:

- **Integration**: A number of lots are combined into a new lot. It is a general representation of mixing, assembling and packing during manufacturing. The relationship is instantiated by $sentTo$ and $madeBy$ fields of Lot record in our system.
- **Division**: A lot is split into multiple new lots. It represents the splitting of a batch and distributing them to different parties. The fields $receivedFrom$ and $sentTo$ are used to record this pattern in our system.
- **Alteration**: A lot is altered and processed. Most of the manufacturing processes belong to this pattern. In our system, we do not create a new batch. Instead, it is recorded in the procedure logs of a lot due to performance concern.
- **Movement**: A lot is moved from a source site to a destination site. It represents the internal transfer within a company between warehouses or the shipment between buyer to seller. The relationship is also recorded by $receivedFrom$ and $sentTo$ in our system.

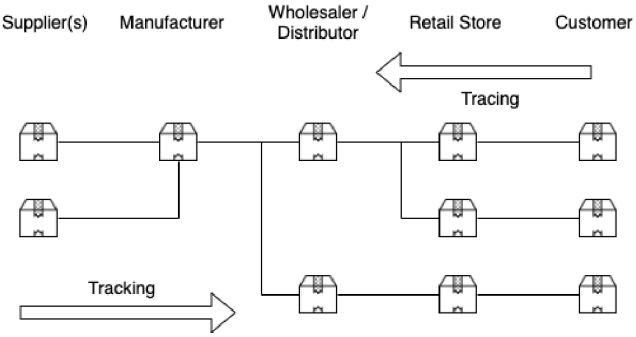

Fig. 2 Tracing and tracking in a supply chain

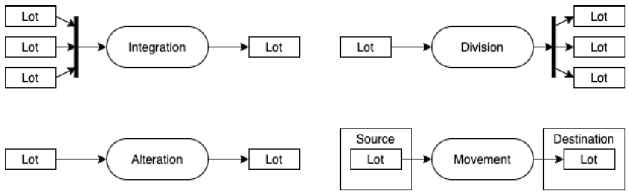

Fig. 3 Basic behavioral patterns of a lot

To use supply chain management system for assisting regulation enforcement, accountability is the key concern. Companies need to record all necessary information, such as responsible party in each process and laboratory test results of products, into the system for periodic auditing and sudden inspection from the government agency. However, a malicious company can modify the data in a centralized server to avoid legal responsibilities. The details of how our system leverages blockchain technology to provide authenticity will be described in section III-B.

For anti-counterfeiting, it highly relies on product tracing. Supply chain system enables customers to trace the source and all the materials used for the manufacturing of a product. Due to division and integration process of lots, a cyclic graph of supply chain may be formed. In addition, as discussed before, not all products have a unique serial number. Nonetheless, the system must provide multi-hop routing during lot tracing. Lastly, digital identity of a trusted company, which owns relevant certificates issued by government, is important for customers to distinguish between genuine or counterfeit products.

B. Blockchain and Smart Contract

A blockchain is a distributed and append-only ledger which maintains a growing list of data *blocks*. Each block contains an ordered set of *transactions* (TXs) data, and typically links to its predecessor through a cryptographic hash pointer [26]. The authenticity of transactions are ensured by digital signatures. Public keys of the users are commonly encoded to *addresses*, which are also used as user identifiers. Blockchain data is distributed and synchronised among all full nodes through a *consensus protocol* such as proof-of-work (PoW), proof-of-authorities (PoA) and proof-of-stake (PoS). Each block and each transaction is validated by all others, before being added to the chain. Based on the distributed architecture and cryptographic techniques adopted, blockchain can provide five properties [26] to supply chain system, namely, immutability, non-repudiation, integrity, transparency and equal rights. It helps the implementation of a secure tamper-proof supply chain system.

Besides the basic assets transfer functionality, blockchain can have *smart contract* for building generic applications. It is a programming script deployed to blockchain and executed among all network nodes. A smart contract can express conditions, iterations and complex business logic [26] which enables developers to implement programmable transactions and develop decentralized applications on top of blockchain.

Based on access control model, blockchain can be classified to *public blockchain* and *private blockchain*; the former is an open network that allows any nodes to join and participate in the consensus, and the latter is a close network where a node requires permission to be granted before joining the network. Blockchain efficiency is highly dependent on the consensus protocol employed, and it is common that permissioned blockchain can achieve higher transaction throughput. For instance, Bitcoin [21] can only support a maximum of 7 transactions per second (tps) [27]. Ethereum [23] supports around 15 tps [27] in public network, while it can reach over 100 tps in private network setting [28]. In our implementation, we adopt a hybrid blockchain approach, which mixes public blockchain and private blockchain.

IV. Design of Our System

A. Entities Definitions

Our supply chain system contains the following six key entities, namely, *Regulation Authority*, *Company*, *Product*, *Lot*, *Procedure* and *Actor*. An overview of the entities' relationship is shown in Fig. 4. Descriptions of the entities are given below.

- **Regulation Authority**: An organization with regulatory authority. Responsible for certifying companies and products, monitoring the quality of lots and manufacturing procedures.
- **Company**: An organization that interacts with the product lots in a supply chain. It is governed by a regulation authority.
- **Product**: A type of goods for sale. Need to follow compliance during the manufacturing and logistic processes.

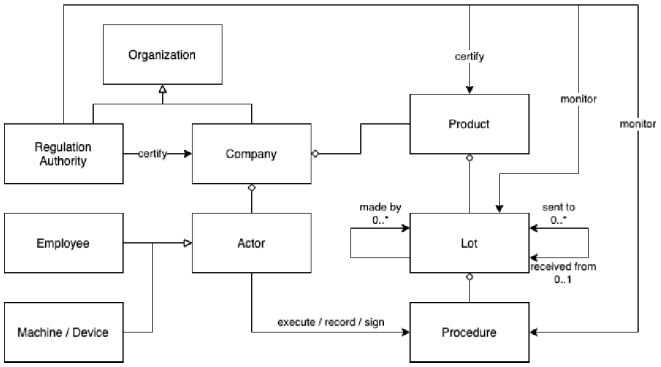

Fig. 4 An overview of the relationship of entities

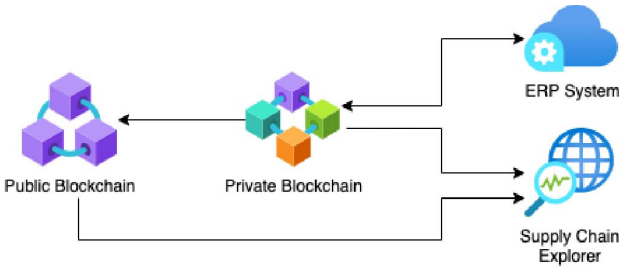

Fig. 5 Architecture of the system

- **Lot**: A tracing unit of product in supply chain.
- **Procedure**: An activity associated to a production lot, which is recorded by an actor and monitored by the regulation authority.
- **Actor**: A person or device of a company who executes, records and signs procedure logs for a lot.

B. Our Construction

1) System Architecture: While blockchain provides trust and data integrity, its lacks the performance and query handling capability to act as a centralised database replacement. Therefore, our system adopts a hybrid blockchain architecture as shown in Fig. 5. The system can be divided into four components, namely, a *public blockchain*, a *private blockchain*, an *Enterprise Resource Planning (ERP) system* and a *supply chain explorer*.

As mentioned in section III-B, private blockchain has a better performance. Our data and smart contracts are mainly stored in a private blockchain, which is hosted by the participating organizations. Companies interact with the private blockchain through a centralized ERP system. All public data are uploaded to the blockchain, while private data are stored in the database of the ERP system. To prevent collusion among blockchain node participants in a private blockchain network, hash of block data are uploaded to a public blockchain periodically.

Supply Chain Explorer is a client providing traceability of product lots and showing the certifications of companies. It is a light-weighted client which only read data from the blockchains and smart contracts.

2) Digital Identities: Each organization and actor must have a digital identity in the system. Each of them must generate its blockchain key pairs and record its public identifier (i.e., account address) in the blockchain. All TXs it makes are signed by the corresponding private key before submitting to the blockchain network. In order to bind the real identity, an organization is responsible to manage its actors' digital identities and records their roles in the blockchain. Also, an organization publishes its blockchain identifier publicly, e.g., by uploading it to its official website address. An organization can also attach a digital certificate, issued by a trusted CA, which is used to generate a signature to its digital identity and uploaded to the blockchain.

3) Data Model and Smart Contracts: To represent the six entities defined in section IV-A, we define two smart contracts, namely, *SCOrganization* and *SCLot*. The functionality of each smart contract are described below:

- **SCOrganization**: (Refer to Script 1) Each deployed *SCOrganization* represents an individual organization. The contract extends $Owned$ and $Certifiable$, which allows a declaration of the owner blockchain address and certification address. There are three mapping fields in the contract, namely, $products$, $employees$ and $devices$, which are used to store and allow a quick look-up for the relevant entities. When a product only has a batch ID on its packing, the system uses Algorithm 3 $TraceByBatch$ to trace the origin of a lot. The field $batchTrace$ is used to serve this algorithm. When a company receives a lot $SCLot_{from}$ and creates a contract $SCLot_{new}$, it sets $SCLot_{new}.batch = SCLot_{from}.batch$ and updates the mapping by $batchTrace[SCLot_{new}.batch].push(SCLot_{new}.address)$. The $lots$ fields in the $Product$ structure stores all the logs of that product. The id field of the $Actor$ structure is the blockchain account address of the actor.
- **SCLot**: (Refer to Script 2) Each deployed *SCLot* represents an individual lot. This smart contract manages and represents the lot behavioral patterns. It uses the fields $madeBy$, $receivedFrom$ and $sentTo$ to link other lots and construct a graph of a supply chain. To represent integration, we use $madeBy$ and $sentTo$. To represent division and movement, we use $receivedFrom$ and $sentTo$. To represent alteration we append $logs$ and change the $status$ based on the work flow. The $procedures$ field stores the regulation procedures and responsible actors.

As mentioned in section IV-B2, all TXs are signed, and thus, the sender of each smart contract's function execution is recorded in the blockchain. We use the account address of the organization and the actor for permission checking.

```
contract SCOrganization is Owned,
                           Certifiable {
  enum ActorType { Employee, Device }
  struct Product {
    string id;
    address[] lots;
    // ...
  }
  struct Actor {
    address id;
    ActorType type;
    // ...
  }
  map<string, Product> products;
  map<string, Actor> employees;
  map<string, Actor> devices;
  map<address, address[]> batchTrace;
  // ...
}
```

Script 1: Smart contract of organization SCOrganization

```
contract SCLot is Owned {
  struct Procedure {
    string id;
    address primary_actor;
    address secondary_actor;
    // ...
  }
  struct Log {
    address actor;
    string content;
    unit256 timestamp;
  }
  address company;
  string product;
  address batch;
  address[] madeBy;
  address receivedFrom;
  address[] sendTo;
  uint8 status;
  unit256 quantity;
  Procedure[] procedures;
  Log[] logs;
  // ...
}
```

Script 2: Smart contract of lot SCLot

4) Tracking and Tracing Algorithms: The system provides three algorithms, namely, $Track$, $TraceByLot$ and $TraceByBatch$. All these algorithms only depend on data retrieval from smart contract and details are listed below:

- **Algorithm 1** $Track : ID_{lot} \rightarrow t$ takes a lot identifier ID_{lot} as input and returns a tree t which represents all downstream lots. It recursively calls the $Track$ function and crafts the returned results as a sub-tree of current root node. The algorithm will only traverse each node once. Thus, the complexity is linear to the size of the result set.
- **Algorithm 2** $TraceByLot : ID_{lot} \rightarrow l$ takes ID_{lot} as input and returns a list l of lot identifiers, which represents a path of its upstream lots. This algorithm

follows the $receivedFrom$ fields of $SCLot$, recursively appends the result list. The complexity is $O(log(n))$ where n is the size of the supply chain graph.

- **Algorithm 3** $TraceByBatch : (ID_{company}, ID_{batch}) \rightarrow m$ takes a company identifier $ID_{company}$ and a lot identifier ID_{batch} as input. It returns a matrix m, which represents multiple paths that the specific lot takes to reach the company. This algorithm is commonly used when a product only print its batch identifier ID_{batch} on its packing and customer wants to trace a product which is bought from $ID_{company}$. This algorithm uses the assisting mapping $SCOrganization\#traceBatch$ to query the lots IDs of targeting batch in current company. It runs $TraceByLot$ on each item in IDs to obtain the result. Let $b = |IDs|$ be the branching factor, the complexity of $TraceByBatch$ is $O(b \cdot log(n))$

Algorithm 1 Track(ID_{lot})

```
Create a tree t and denote the root node as r
c ← SearchContract(ID_lot)
r ← ID_lot
if c ≠ φ then
    for all ID_to in c.sentTo do
        t_to ← Track(ID_to)
        Graft t_to as a child of r
    end for
end if
return t
```

Algorithm 2 TraceByLot(ID_{lot})

```
l ← [ID_lot]
c ← SearchContract(ID_lot)
if c = φ then
    return l
else
    return Concat(l, TraceByLot(c.receivedFrom))
end if
```

Algorithm 3 TraceByBatch($ID_{company}, ID_{batch}$)

```
m ← new matrix
c ← SearchContract(ID_company)
if c = φ then
    IDs ← c.batchTrace[ID_batch]
    size ← |IDs|
    for i ← 0 to size − 1 do
        m[i] ← TraceByLot(IDs[i])
    end for
end if
return m
```

5) External Storage: Currently, most blockchain frameworks do not support TX with large amount of data, meaning that storing of multimedia files on blockchain is infeasible. In our system, the multimedia files, like the certificate of company and laboratory test results of product samples, are uploaded to a separate decentralized storage and the link and a hash of the file(s) are stored in blockchain.

V. Implementation

Our system is implemented on a private Ethereum blockchain network hosted on Alibaba Cloud. We use geth version 1.9.14 as Ethereum node client and solidity version 0.4.24 for smart contract implementation. We use swarm version 0.5.7 for distributed storage on top of Ethereum. The ERP system and supply chain explorer are built by Ruby on Rails with Ruby version 2.5.1 and Rails version 5.1.6 and PostgreSQL database version 12.2. The periodic backup process of the hybrid blockchain is implemented by schedule job and a Node.js script with version 10.16.3 which submits TXs to public blockchain network.

The system is deployed to five Elastic Compute Service (ECS) instances. Each of them has a 40 GB system disk in Ubuntu 16.04 64-bit. Each server has an Elastic IP address with a 5 MB/s bandwidth. The mining speed of Ethereum network is configured to 3 s/block. The five ECS instances are located in different physical zone and have the specifications shown in Table I. The responsibility of each server is described as the follows:

- **Server 1**: Host an ERP server, a supply chain explorer server, a PostgreSQL database, a swarm node, a primary Ethereum boot node and an Ethereum Remote Procedure Call (RPC) node.
- **Server 2**: Host a primary Ethereum mining node.
- **Server 3**: Host a secondary Ethereum boot node.
- **Server 4**: Host a secondary Ethereum mining node.
- **Server 5**: Host a secondary Ethereum mining node.

Our system's performance is evaluated by the test cases jointly designed with our collaborating industrial partners, based on a drug manufacturing scenario involving manufacturer, distributors and retailers. Each company has multiple work flow stages and processes for a product lot. For example, a manufacturer requires employees to perform eight rounds of quality assurance and quality control testing during production. Since our system works on lot (a product batch) instead of a single product unit, the underlying requirement of the

[1]The design applies to blockchain architectures that support smart contract. Our implementation is based on Ethereum.

Name	Zone	Instance Type	CUP	Memory
Server 1	Shenzhen Region Zone A	ecs.sn1ne.2xlarge	8 vCPU	16 GB
Server 2	Shenzhen Region Zone A	ecs.sn1ne.2xlarge	8 vCPU	16 GB
Server 3	Hangzhou Region Zone G	ecs.n4.small	1 vCPU	2 GB
Server 4	Shenzhen Region Zone E	ecs.hfc5.large	2 vCPU	4 GB
Server 5	Shenzhen Region Zone C	ecs.hfc5.large	2 vCPU	4 GB

TABLE I: Specification of Alibaba Cloud ECS instances

	[7]	[8]	[9]	[10]	Ours
Blockchain Platform	Extended Bitcoin	Ethereum	Ethereum	Ethereum	Ethereum[1]
Data Structure	UTXO	Smart Contract	ERC721	Smart Contract	Smart Contract
Tracing and Tracking Media	TX	Event Logs	Event Logs	Event Logs	Smart Contract
Authentication Level	Company	Company	Company	Company	Actor (Employee or Device)
Supports Quality Management	No	No	No	No	Yes
Supports Multi-hop Routing	Yes	No	Yes	No	Yes
Supports Tracing without UID	No	No	No	No	Yes

TABLE II: Comparison between existing solutions and our proposed design

blockchain is light. A typical scenario of drug selling in Shanwei City, a manufacturer sells its products to a sole distributor. The distributor sells the drugs to 103 hospitals and clinics. The monthly production of a drug can reach 20 milion product units in peak. It is commonly set to 100,000 product units per batch. Thus, our system needs to handle 7 batches per day. The number of trace and track queries, on the other hand, depends on the number of product units (e.g., each customer may issue a track query when they receive a product unit). Nonetheless, the corresponding query is a read-only transaction, and typically no consensus is needed[2].

Our measurement on the performance of the system is based on the average time of common operations. For lot creation, the average time is 17.98s, while for recording a stage in the work flow, the average time is 1.91s. The lot creation time is linear to the number of stages in the work flow. The average time for $TraceByLot$ and $TraceByBatch$ is 716.33ms and 236.01ms respectively. The tracing time is linear to the size of result set. The time for a tracking operation is also linear to the size of the result set, which is 8.35s for our system.

VI. Evaluation

A. System Analysis

Our system is able to achieve the five features, namely, *traceability*, *accountable*, *immutable*, *verifiability and transparency* and *efficient*.

- **Traceability**: In our system, all logistic information is stored in the smart contract in blockchain. A user can use UID to perform lot tracking and tracing by running $Track$ and $TraceByLot$ algorithms respectively. Also, the $TraceByBatch$ algorithm supports multi-hop routing for finding multiple possible paths generated due to lot division and lot integration in the practical settings.
- **Accountable**: Every interaction with the product lot is recorded in blockchain by executing the log function in *SCProcedure*. Moreover, all TXs are signed by the private key of actors, entity authentication is guaranteed by the digital signatures. Thus, the responsible party(s) of a particular stage in the manufacturing and logistic work flow can be traced easily, which is typically important for regulated industries.
- **Immutable**: Data immutability and integrity is achieved by the immutability property of blockchain. The hybrid blockchain architecture further reduces the risk of data tampering even there is any collusion among the private blockchain nodes, since hashes of private blockchain blocks are backup to a public blockchain periodically.
- **Verifiability and Transparency**: Our system allows validator nodes to join as parts of the private blockchain network, all blockchain TXs are publicly verifiable. Besides, all the certifications of products and companies, the procedure logs of production and the logistic information can be retrieved through the smart contract functions, any user can query and verify by itself.
- **Efficient**: We adopt a hybrid blockchain architecture in our system to maintain the efficiency. Typically, most smart contract query and TXs are submitted to the private blockchain layer, which has significant improvement when comparing to directly submit TXs to a public blockchain. Besides, the proposed tracing algorithms ($TraceByLot$ and $TraceByBatch$) have logarithmic complexity that allows customer to trace the product origin efficiently for anti-counterfeiting purpose. Moreover, our implementation and testing shows that the system is efficient and practical for real-world settings.

[2]A read-only transaction can be handled locally by a full node.

B. Comparison

Table II summarizes the comparison between the existing blockchain-based supply chain systems [7]–[10] and our proposed design. Most of the existing solutions [8]–[10] were built on Ethereum blockchain platform and [7] was built on Bitcoin network with extended capabilities. While our proposed system is also implemented on Ethereum, but the design itself is possible to deploy on any blockchain platforms that supports smart contract. The proposed systems in [8], [10] and ours adopt smart contracts to store the traceability records, while [7] and [9] use UTXO and ERC721 in their design respectively. However, the design of [9] violated the original intention of ERC721 token, since a "quantity" field is added. In order to perform tracing or tracking in the system, [7] would query TXs from the blockchain. Then, trace through the transaction inputs and outputs. In our design, we direct query the smart contract instead. For [8]–[10], they use event logs in Ethereum as a tracing media. However, searching in event logs is slower than directly interact with smart contract. Besides, this may affect the availability of systems, since indexing event logs consumes additional computational and storage resources. In a default setting of Ethereum node, it only indexes one year of event logs, which may not suitable for products with long expiry day. In additional, unlike the existing solutions which authenticate users through validating the signatures of companies, our system authenticates each actor, employee or

device. This increases the accountability and transparency of the system, since every action and procedure are logged in a more fine-grained manner. In [8] and [10], the proposed systems do not support multi-hop routing and do not consider all logistics behavioral patterns, while [7], [9] and our solution do. For customers, it is common that they are not in the possession of a unique identifier (UID) for tracing the origin of a product. Our proposed system includes an algorithm (Algorithm 3) in the smart contract to allow efficient tracing with without UID. Therefore, our system is able to provide a more complete solution to supply chain management.

VII. Conclusion

In this article, we propose a blockchain-based supply chain management system for traceability, regulation and anti-counterfeiting and show that it achieves the common functionality requirements for logistic and supply chain systems. We analyzed existing blockchain-based supply chain management solutions and point out their insufficiencies in practical settings. We provide an implementation and evaluation of our proposed system. Specifically, our system supports quality management, multi-hop routing and tracing without unique identifier. The performance of our system is also satisfactory in a practical environment. Our proposed design is generic. On one hand, it can be adopted in industries with the same logistic and supply chain requirement. On the other hand, it can be implemented in various blockchain architectures that support smart contract.

Acknowledgements. Part of this research is supported by the Innovation and Technology Fund University-Industry Collaboration Programme and UBI BlockChain Internet Limited (Ref: UIM/331)

References

[1] T. W. Malone, J. Yates, and R. I. Benjamin, "Electronic markets and electronic hierarchies," *Communications of the ACM*, vol. 30, no. 6, pp. 484–497, 1987.

[2] V. Choudhury, "Strategic choices in the development of interorganizational information systems," *Information Systems Research*, vol. 8, no. 1, pp. 1–24, 1997. [Online]. Available: https://doi.org/10.1287/isre.8.1.1

[3] A. Bechini, M. G. C. A. Cimino, F. Marcelloni, and A. Tomasi, "Patterns and technologies for enabling supply chain traceability through collaborative e-business," *Information and Software Technology*, vol. 50, no. 4, pp. 342–359, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950584907000213

[4] L. Liu and W. Jia, "Business model for drug supply chain based on the internet of things," in *2010 2nd IEEE InternationalConference on Network Infrastructure and Digital Content*, 2010, pp. 982–986.

[5] M. Garaus and H. Treiblmaier, "The influence of blockchain-based food traceability on retailer choice: The mediating role of trust," *Food Control*, vol. 129, p. 108082, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0956713521002206

[6] J. Hua, X. Wang, M. Kang, H. Wang, and F. Wang, "Blockchain based provenance for agricultural products: A distributed platform with duplicated and shared bookkeeping," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, Conference Proceedings, pp. 97–101.

[7] Y. Huang, J. Wu, and C. Long, "Drugledger: A practical blockchain system for drug traceability and regulation," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, Conference Proceedings, pp. 1137–1144.

[8] N. Saxena, I. Thomas, P. Gope, P. Burnap, and N. Kumar, "Pharmacrypt: Blockchain for critical pharmaceutical industry to counterfeit drugs," *Computer*, vol. 53, no. 7, pp. 29–44, 2020.

[9] M. Westerkamp, F. Victor, and A. Küpper, "Tracing manufacturing processes using blockchain-based token compositions," *Digital Communications and Networks*, vol. 6, no. 2, pp. 167–176, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S235286481830244X

[10] A. Musamih, K. Salah, R. Jayaraman, J. Arshad, M. Debe, Y. Al-Hammadi, and S. Ellahham, "A blockchain-based approach for drug traceability in healthcare supply chain," *IEEE Access*, vol. 9, pp. 9728–9743, 2021.

[11] S. Chen, R. Shi, Z. Ren, J. Yan, Y. Shi, and J. Zhang, "A blockchain-based supply chain quality management framework," in *2017 IEEE 14th International Conference on e-Business Engineering (ICEBE)*, 2017, Conference Proceedings, pp. 172–176.

[12] National Medical Products Administration, China, "Drug administration law chapter 6 drug packaging management," 2002, [Online]. Available: https://www.nmpa.gov.cn/xxgk/zhcjd/20020120134401569.html.

[13] State Administration for Market Regulation, "Drug registration and management law," 2020, [Online]. Available: http://gkml.samr.gov.cn/nsjg/fgs/202003/t20200330_313670.html.

[14] National Medical Products Administration, China, "Drug production management law," 2020, [Online]. Available: https://www.nmpa.gov.cn/yaopin/ypfgwj/ypfgbmgzh/20200330182901110.html.

[15] J. I. Cash and B. R. Konsynski, "Is redraws competitive boundaries," *Harvard business review*, vol. 63, no. 2, pp. 134–142, 1985.

[16] ISO, "Iso 8402:1994 quality management and quality assurance - vocabulary," 1994.

[17] K. van Dorp, "Tracking and tracing:a structure for development and contemporary practices," *Logistics Information Management*, vol. 15, no. 1, pp. 24–33, 2002. [Online]. Available: https://doi.org/10.1108/09576050210412648

[18] M. Thakur and C. R. Hurburgh, "Framework for implementing traceability system in the bulk grain supply chain," *Journal of Food Engineering*, vol. 95, no. 4, pp. 617–626, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0260877409003264

[19] I.-H. Hong, J.-F. Dang, Y.-H. Tsai, C.-S. Liu, W.-T. Lee, M.-L. Wang, and P.-C. Chen, "An rfid application in the food supply chain: A case study of convenience stores in taiwan," *Journal of Food Engineering*, vol. 106, no. 2, pp. 119–126, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S026087741100210X

[20] T. Kelepouris, K. Pramatari, and G. Doukidis, "Rfid-enabled traceability in the food supply chain," *Industrial Management & data systems*, 2007.

[21] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Bitcoin*, vol. 4, 2008, [Online]. Available: https://bitcoin.org/bitcoin.pdf. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[22] L. Aniello, B. Halak, P. Chai, R. Dhall, M. Mihalea, and A. Wilczynski, "Towards a supply chain management system for counterfeit mitigation using blockchain and puf," *arXiv preprint arXiv:1908.09585*, 2019.

[23] V. Buterin, "A next-generation smart contract and decentralized application platform," *white paper*, vol. 3, no. 37, 2014.

[24] W. Entriken, D. Shirley, J. Evans, and N. Sachs, "Eip-721: Erc-721 non-fungible token standard," *Ethereum Improvement Proposals*, 2018, [Online]. Available: https://eips.ethereum.org/EIPS/eip-721.

[25] Committee for Developing Educational Materials for Food Traceability, *Handbook for Introduction of Food Traceability Systems (Guidelines for Food Traceability)*. Food Marketing Research and Information Center (FMRIC), Japan, 2003, [Online]. Available: https://www.maff.go.jp/j/syouan/seisaku/trace/pdf/handbook_en.pdf. [Online]. Available: https://www.maff.go.jp/j/syouan/seisaku/trace/pdf/handbook_en.pdf

[26] X. Xu, I. Weber, and M. Staples, *Architecture for Blockchain Applications*, 2019.

[27] I. A. Seres, D. A. Nagy, C. Buckland, and P. Burcsi, "Mixeth: efficient, trustless coin mixing service for ethereum," in *International Conference on Blockchain Economics, Security and Protocols (Tokenomics 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2019.

[28] M. Schäffer, M. di Angelo, and G. Salzer, "Performance and scalability of private ethereum blockchains," in *Business Process Management: Blockchain and Central and Eastern Europe Forum*, C. Di Ciccio, R. Gabryelczyk, L. García-Bañuelos, T. Hernaus, R. Hull, M. Indihar Štemberger, A. Kő, and M. Staples, Eds. Cham: Springer International Publishing, 2019, pp. 103–118.