Secure and Scalable Blockchain-Based Framework for IoT-Supply Chain Management Systems

Omimah Alsaedi[1†], Omar Batarfi[2††], Mohammed Dahab[1†],
oalsaedi0003@kau.edu.sa obatafri@kau.edu.sa mdahab@kau.edu.sa
[†]Department of Computers Science, King Abdulaziz University, Saudi Arabia
[††]Department of Information Technology, King Abdulaziz University, Saudi Arabia

Abstract

Modern supply chains include multiple activities from collecting raw materials to transferring final products. These activities involve many parties who share a huge amount of valuable data, which makes managing supply chain systems a challenging task. Current supply chain management (SCM) systems adopt digital technologies such as the Internet of Things (IoT) and blockchain for optimization purposes. Although these technologies can significantly enhance SCM systems, they have their own limitations that directly affect SCM systems. Security, performance, and scalability are essential components of SCM systems. Yet, confidentiality and scalability are one of blockchain's main limitations. Moreover, IoT devices are lightweight and have limited power and storage. These limitations should be considered when developing blockchain-based IoT-SCM systems. In this paper, the requirements of efficient supply chain systems are analyzed and the role of both IoT and blockchain technologies in providing each requirement are discussed. The limitations of blockchain and the challenges of IoT integration are investigated. The limitations of current literature in the same field are identified, and a secure and scalable blockchain-based IoT-SCM system is proposed. The proposed solution employs a Hyperledger fabric blockchain platform and tackles confidentiality by implementing private data collection to achieve confidentiality without decreasing performance. Moreover, the proposed framework integrates IoT data to stream live data without consuming its limited resources and implements a dual-storge model to support supply chain scalability. The proposed framework is evaluated in terms of security, throughput, and latency. The results demonstrate that the proposed framework maintains confidentiality, integrity, and availability of on-chain and off-chain supply chain data. It achieved better performance through 31.2% and 18% increases in read operation throughput and write operation throughput, respectively. Furthermore, it decreased the write operation latency by 83.3%.

Keywords:
Blockchain, Supply Chain, SCM, IoT, Confidentiality

1. Introduction

Nowadays, modern business involves multiple individuals and organizations who share large amounts of information over different geographical areas, which means supply chains operate in a growing ever-changing environment [1]. Supply chain systems are defined as a network of all participating entities in product manufacturing, transportation, storage, and sale, including individuals, organizations, resources, and activities [2]. Thus, managing supply chain systems is considered one of the most important and complicated tasks in the industrial domain.

Data is the most valuable component of supply chain management (SCM) systems. Therefore, the most common security threats target SCM systems by stealing, manipulating, or disclosing confidential data, with the aim of interrupting a service or gaining a financial advantage [3,4].

Digital technologies and information technology (IT) infrastructure are integrated to create efficient SCM systems that simplify and streamline supply chain operations and provide a better business outcome. However, these technologies bring their own security risk and challenges that negatively affect SCM security as it is connected to an unstable internet environment.

The Internet of Things (IoT) is a key technology that effectively enhances the collaboration, visibility, and traceability of SCM systems by allowing different objects to sense and monitor data in a timely manner, to provide accurate information and support better decision-making [5] across different supply chain stages without any human interaction. However, security remains a challenge since IoT-based systems are centralized and rely on third parties to store data [6]. Moreover, IoT devices are lightweight and heterogeneous in nature with limited storage and power which must be mostly devoted to executing core application functionalities. This means that current security paradigms do not perfectly fit IoT-based systems.

Blockchain is an emerging technology that has been introduced as the missing piece of the puzzle to solve security, reliability, and visibility challenges that IoT-SCM systems face due to the unique characteristics that lead both the academia and industry to use it beyond cryptocurrencies and financial services, including in applications such as healthcare, real estate, and logistics [7]. Although the decentralization and immutability features of blockchain enhance the availability and integrity of IoT-SCM systems, the high transparency of blockchain reduces data confidentiality which is considered one of the key security requirements to maintain supply chain sustainability [8].

Manuscript received December 5, 2022
Manuscript revised December 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.12.5

Moreover, scalability is a major issue in blockchain technology that needs to be addressed and taken into consideration while developing such complex systems. Scalability concerns the number of transactions being processed, and the required performance and storage needed to handle these transactions [9].

Most of the existing blockchain-based IoT-SCM systems do not take these challenges into account. This paper aims to propose a blockchain-based IoT-SCM system that tackles the data confidently issue by taking into consideration blockchain scalability and IoT devices limitations without negatively affecting efficiency. The main contributions of this paper are that it:

- Investigates the requirements of IoT-SCM systems.
- Investigates the role of blockchain in enhancing the security of IoT-SCM systems.
- Identifies the security approaches used by current blockchain-based SCM systems and their limitations.
- Proposes a secure blockchain-based framework that fits IoT-SCM security requirements.

The remaining paper is organized as follows: Section 2 provides theoretical background on blockchain and IoT systems requirements; Section 3 reviews the literature on blockchain-based SCM systems; Section 4 describes the proposed framework; Section 5 gives the application results of the proposed framework and analyzes its performance as compared to systems identified in the literature review; the conclusions and suggestions for future work are presented in Section 6.

2. Theoretical Background

In this section, the theoretical basis used in this study is discussed, including IoT-SCM systems requirements, and the use of blockchain for IoT-SCM systems security.

2.1. IoT-SCM Systems Requirements

SCM efficiency is described in terms of several performance objectives. According to the Supply Chain Operations Reference Model (SCOR) that was developed by the Supply Chain Council (SCC) [10], the key performance attributes of SCM are cost, responsiveness, reliability, flexibility, and asset management.

Recently, after the well-known terrorist attack in 2001 and the negative impact that had on trades and supply chains across the world, researchers added sustainability and security as critical performance attributes of SCM [11].

To meet the aforementioned attributes and achieve SCM goals, SCM systems should be developed according to the following requirements:

- Collaboration: describes the ability of a system to enable multiple organizations to communicate across organizational boundaries to manage shared assets effectively [12].
- Visibility: describes the level at which an organization can precisely see all the activities happening within its supply chain [8].
- Tractability: refers to the ability to track and follow the path of each supply chain input from origin to destination [8].
- Scalability: refers to the ability of a system to scale up with additional resources to cope with demand and the changing needs of supply chains without being affected negatively
- Security: concerns about protecting supply chain resources and data from any unauthorized access, modification, disclosure, or destruction to guarantee integrity, confidentiality, and availability.

Figure 1 maps each SCM objective with the corresponding SCM systems requirements to achieve it.

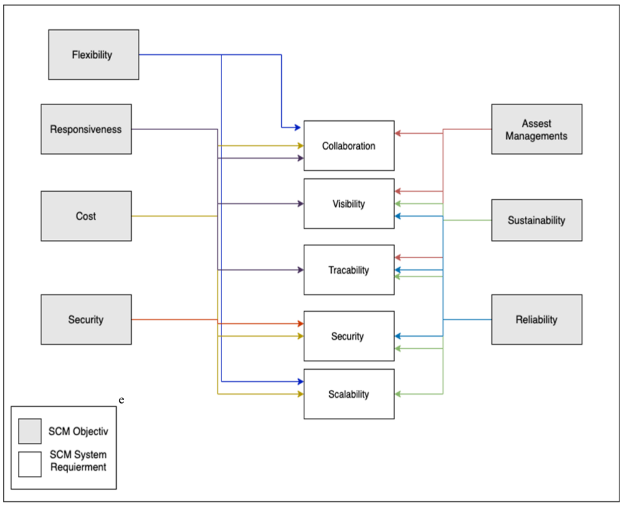


Figure 1 The relationship between SCM objectives and requirements

The integration of IoT devices in SCM systems allows data to be allocated and transmitted automatically in a real time manner among different objects, which enhances supply chain collaboration, visibility, and traceability requirements of SCM systems. However, the centralized architecture of IoT systems, the heterogeneous nature of IoT devices, the different characteristics they have, and the large amount of data they generate, make them susceptible to different attacks that threaten the security of SCM systems.

2.2. Blockchain for IoT-SCM Systems

Blockchain is defined as a distributed, appended-only, and immutable ledger of transactions that is added to a network in the form of digital blocks [7]. Blocks are chronologically ordered and each block is connected to the preceding block through a cryptographic hash function. To further illustrate this concept, a blockchain network consists of numerous nodes sharing the same administrative role and storing the same copy of data. There is no centralized authority over the network and no single point of failure. Once a transaction is validated and added to the blockchain, the ledger is updated with the new transaction at each participating node and cannot be altered or changed in the future [7]. The technology on which the blockchain is built has empowered it with many distinctive features, including:

- Decentralization: blockchain eliminates the need for the involvement of a centralized authority or a third party to control and store the data as all nodes within the network can validate the transactions independently and store the same copy of the ledger. In case of a node failure, no data will be lost. This feature keeps the network up and running and the network will remain functional.
- Reliability: although there is no centralized authority over the blockchain network, it remains reliable as each generated transaction should be digitally signed using the hashing algorithm with the sender's private/secret key to be considered as a valid transaction [13], which ensures authenticity. In addition, every transaction within the network is validated from each node using some consensus protocol to be added to the ledger [7]. This mechanism provides trust and ensures ledger consistency among the distributed networks.
- Immutability: hashing is the backbone of blockchain technology. Every block is connected to the previous block through its hash except the first block in the chain which is formally called the genesis block and it points to itself [14]. The add-only structure of the blockchain makes the blocks chronologically connected. Such orientation of connection means changing one block will require changing all preceding blocks on all participating nodes. As a result, changing or modifying the ledger is almost impossible, which enhances the integrity of the data.
- Transparency: all performed transactions are stored on each node with the related data and hash value, which makes the ledger visible and audible to all the nodes within the network [13].
- Supply chain security concerns about protecting SCM systems from any threat that results from incorrect, incomplete, or illegal access to information, and negatively affects organization assets and the confidentiality, integrity, or availability of information, which leads to loss of functionality, connectivity, performance, or capacity 3,15].

The unique characteristics of blockchain have influenced researchers to adopt IoT-SCM systems to create a reliable and secure architecture that fits the requirement of these systems [6].
Table 1 illustrates the role of blockchain in archiving the requirements of each IoT-SCM system.

3. Related Work

In this section, recently published research on the use of blockchain in SCM systems and the role of blockchain in IoT-SCM security are discussed. Studies published in the past five years are considered in this section.

Most of the existing works discuss the integration of blockchain with SCM systems as a method to provide better supply chain visibility and traceability. Hasan et al. [16] proposed a blockchain-based solution for managing single-echelon shipments through sensor-enabled containers. The proposed solution implements a smart contracts feature in the Ethereum blockchain to control and manage the information flow between a sender and a receiver and allow them to track the shipment status as well as receive alerts in case of any violation.

Arena et al. [17] proposed a private blockchain-based solution to certify the extra virgin olive oil supply chain in order to ensure oil provenance by allowing the final customer to access a tamper-proof history of the oil, from farming to transportation processes. In their solution, data is collected automatically using IoT devices and all records are stored directly in blockchain, accessible by sellers and end-users through a website. The simulation results using OMNeT++ showed that the system performance is not feasible in a realistic situation as the transaction arrival rate may vary according to the number of transactions and the amount of data stored on-chain over time, hence, a dynamic auto-tuning mechanism of blockchain was proposed to enhance the performance of the network. However, implementing such a mechanism adds more complexity to the design a of practical SCM system.

Botcha et al. [18] designed a conceptual approach to enhance IoT-based pharmaceutical supply chain traceability and reliability through blockchain. In their approach, each part of the supply chain (supplier, production, distribution, consumption) has a separate block that records current part data collected by the linked IoT device as well as its transaction data. The whole blockchain system is maintained on cloud storage where the records are transmitted and stored. Hence, the approach allows full data transparency and audibility which enhances end to end data traceability and security, and provides value-added services to pharmaceutical companies in areas such as production scheduling, inventory optimization, and early warning. On the other hand, because of the security architecture of blockchain technology, the development of a secure IoT-SCM system was the key goal that led some researchers to implement blockchain-based SCM systems.

Table 1 Blockchain's role in IoT-SCM systems

SCM Requirement	Blockchain Feature	Mechanism
Collaboration	Decentralization	- Blockchain offers a shared ledger among all SC stakeholders and IoT devices in which each party can access one unified data resource for all supply chain phases
Visibility and Traceability	Decentralization Transparency	- Transparency feature of blockchain allows each transaction to be registered in the shared ledger in a reliable way in which the transaction is visible to all stakeholders - Transactions are chronologically ordered in which each product can be traced from origin to destination in addition to providing provenance
Security	Decentralization Immutability	- Blockchain decentralized architecture eliminates the need for a single authority or intermediate storage, allow the network to operate across SC stakeholders which enhance the availability and eliminates the single point of failure risk. - Immutability provides a tamper-proof ledger which increases integrity

Rathee et al. [19] proposed a secure industrial IoT framework that uses blockchain technology to track industry activities and shipments. Their aim was to have reliable data sharing with full visibility over the shared IoT data in a way that any malicious node or activity would be detected. Simulation experiments using a customized testbed were performed on both the proposed blockchain-based framework and the conventional client-server approach. Compared to the conventional approach, the results showed that the proposed framework decreases the possibility of falsification attack, black hole attack, and product loss ratio by 91.5%, 97.5%, and 88.9%, respectively. Despite this, the framework architecture involves IoT devices in the blockchain network as a node that keeps a copy of the blocks in them, which may be quite challenging in terms of performance and scalability since IoT devices are limited in their power and capacity.

Kuo and Su [20] discussed the problem of scalability and data integrity in supply chain systems, although 18blockchain can maintain data integrity and prevent data from being maliciously tampered with, it could not be used directly as secure storage for IoT supply chain data since IoT devices are limited in size and power and generate an enormous amount of data. Moreover, commonly used blockchains have limited throughput. Hence, involving IoT devices as blockchain nodes would increase the transaction arrival rate (TAR), and, consequently, increase latency and cost. In order to overcome this problem, the authors proposed blockchain-indexed storage (BIS). In BIS, data is collected from IoT devices and stored in suitable off-chain storage according to its type (i.e., high-frequency or low-frequency). A fingerprint associated with each data bundle is generated. The data-id in the off-chain storage and fingerprint is stored in the blockchain that generates a transaction ID linked to them. As a result, data integrity is maintained since fingerprints are stored in a blockchain and cannot be tampered with. In the case of supply chain data retrieving, BIS uses data fingerprints and transaction IDs to retrieve full data.

Älvebrink and Jansson [21] proposed a conceptual blockchain-based framework to solve access control, data integrity, and data tampering security issues in IoT-SCM systems. The proposed framework takes into consideration supply chain data scalability and IoT device limitations by gathering and uploading IoT data into a local network for filtering. As a result, only important data will be uploaded to a private blockchain located within a cloud service. This reduces the size of IoT data stored on blockchain nodes and IoT devices. Confidentiality is a key security requirement for SCM systems to maintain business competitiveness [22]. Blockchain architecture does not maintain confidentiality by nature. Researchers tend to implement different algorithms that provide confidential data sharing for blockchain-based SCM systems.

The authors in [23] and [24] proposed a multi-channel blockchain-based framework to implement a secure and confidential supply chain system. In their studies, they created multiple channels. Each channel is completely isolated from the other channels, and they all have their own transactions and nodes. Thus, two supply chain stakeholders could share exclusive data that is not visible to other members outside the channel. However, having multiple-channel architecture could increase system complexity and requires additional storage as a single node could have more than one ledger. Furthermore, having no channel combines all supply chain stakeholders and leads to inefficient supply chain traceability as each participant is allowed to track part of the product history. This decreases transparency and increases the rate of fraud. In contrast, encryption is another diffuse method that is used by other researchers to provide data confidentiality in blockchain-based SCM systems.

In [25], the author proposed an approach for hiding confidential data in blockchain-based SCM systems. In his approach, Elliptic Curve Integrated Encryption Scheme (ECIES) encryption and decryption procedures are applied to each part of the transaction that holds sensitive data. The encryption technique takes the sensitive data along with a set of authorized recipients with public keys as input to perform the encryption process. To obtain the encrypted data, each actor in the network attempts to decrypt the cipher text, and, if the actor is an authorized user, the data is retrieved.

Jianfeng et al. [26] proposed a consortium blockchain framework that combines both symmetric encryption algorithms and asymmetric encryption algorithms through blockchains with smart contracts in order to encrypt both the sensitive data of the IoT-SCM system itself and the private key of the encrypted data as an additional step to protect confidential data. Yang et al. [27] proposed a multi-layer blockchain-based IoT framework that includes an application layer, an interface layer, a service layer, and a storage layer. In order to provide an efficient storing solution, the proposed system adopts the dual storage model in which public data is stored using an off-chain MySQL database, while the hashes of the public data along with encrypted private data are stored on-chain. However, private data is encrypted using the cipher block chaining (CBC) mode of the Advanced Encryption Standard (AES) algorithm in which the key generation and the encryption processes are performed using the smart contract. Furthermore, the generated key is encrypted using elliptic curve cryptography and stored on-chain along with the public key of the authorized node. Consequently, to view the private data, the authorized node needs to perform a two-step decryption process to decrypt the key and private data.

Flapper [28] proposed a blockchain framework to support supply chain visibility and data privacy that uses an XML-standard version to manage and perform Attribute-Based Access Control. The proposed architecture implements access control logic in a distributed manner using an Ethereum smart contract and integrates BigchainDB [29] to serve as a database to store asset data and access control policies. However, the proposed framework is not implemented as the author faced many obstacles in integrating two types of blockchain in one solution. Table 2 summarizes the main differences between the reviewed literature.

4. Proposed Framework

This section describes the proposed framework in detail including the design criteria, the proposed architecture, and the implementation details.

4.1. Framework Design Criteria

When mapping the requirements of SCM systems with the limitations found in the reviewed literature, several criteria are deduced to design the proposed framework as follows:

- Implement decentralized architecture rather than traditional centralized architecture to enhance availability.
- Use a permissioned consortium blockchain network to enhance privacy, integrity, and reliability.
- Implement a confidentiality mechanism considering the limitations of the mechanisms examined in the literature review.
- Integrate IoT devices in an efficient manner taking into consideration the limited storage and computational power of an IoT device, and the security issues associated with IoT devices.
- Integrate an off-chain database that allows the SCM system to be scalable without affecting the confidentiality and integrity of both on-chain and off-chain data.

4.2. Framework Architecture

This paper proposes a multi-layer framework based on the stated criteria above. Figure 2 illustrates the architecture of the proposed framework. The framework consists of the following three layers:

1. **Data Acquisition Layer**: this layer is designed to collect product data through a user interface (UI) and IoT device. Basically, the product life cycle in the proposed framework is initiated when the supplier inserts product data through the UI, consequently, each participant in the supply chain updates the product data through the UI. Furthermore, at the transportation stage, IoT sensors automatically detect and stream live product data such as temperature and location. Both UI and IoT devices are linked with the blockchain network through REST API. The UI is implemented using React JS, and IoT devices are simulated using JavaScript programming language and Redis in-memory database.

2. **Blockchain Layer**: this layer is used to store, process, and track product data in a secure manner. Basically, each data inserted through the REST API is transmitted to the blockchain network as a signed transaction that is verified and added to a tamper-proof ledger. This framework implements Hyperledger fabric blockchain [30] and uses

JavaScript programming language for writing blockchain with smart contracts. We use the open-source Hyperledger fabric blockchain platform for two reasons: first, Hyperledger fabric allows different components to be plug-and-play and has a modular design that fits a broad range of industry use cases; secondly, it is considered a consortium blockchain that allows pre-selected, equally privileged participants to access the network [31].

Table 2 literature review comparison

Reference	Blockchain Platform	IoT	Off-Chain Storage	Data Security			Performance	
				Integrity	Availability	Confidentiality	Throughput	Latency
[16]	Ethereum	✓	-	✓	✓	-	N/A	
[17]	Hyperledger fabric	✓	-	✓	✓	-	N/A	
[18]	N/A	✓	-	✓	✓	-	N/A	
[19]	N/A	✓	-	✓	Low	-	N/A	
[20]	Ethereum	✓	✓	✓	✓		N/A	
[21]	N/A	✓	✓	✓	✓	-	N/A	
[23]	Hyperledger fabric	-	-	✓	✓	Implemented using multiple-channel architecture	N/A	
24]	Hyperledger fabric	-	-	✓	✓	Implemented using multiple-channel architecture	Read: 32.0-49.0 TPS Write:15.8-21.7 TPS	N/A
[27]	Hyperledger-fabric	✓	✓	✓	Low	Implemented by encrypting private data using CBC	Read: 250 TPS Write:125-250 TPS	Read: 0.02s Write: 0.12s
[25]	N/A	-	-	✓	✓	Implemented by encryption	Write: 0.51 TPS	N/A
[26]	Hyperledger fabric	✓	-	✓	✓	Implemented by encrypting data using multiple encryption algorithms	N/A	
[28]	Ethereum	-	✓	✓	Low	Implemented using access control policy	N/A	

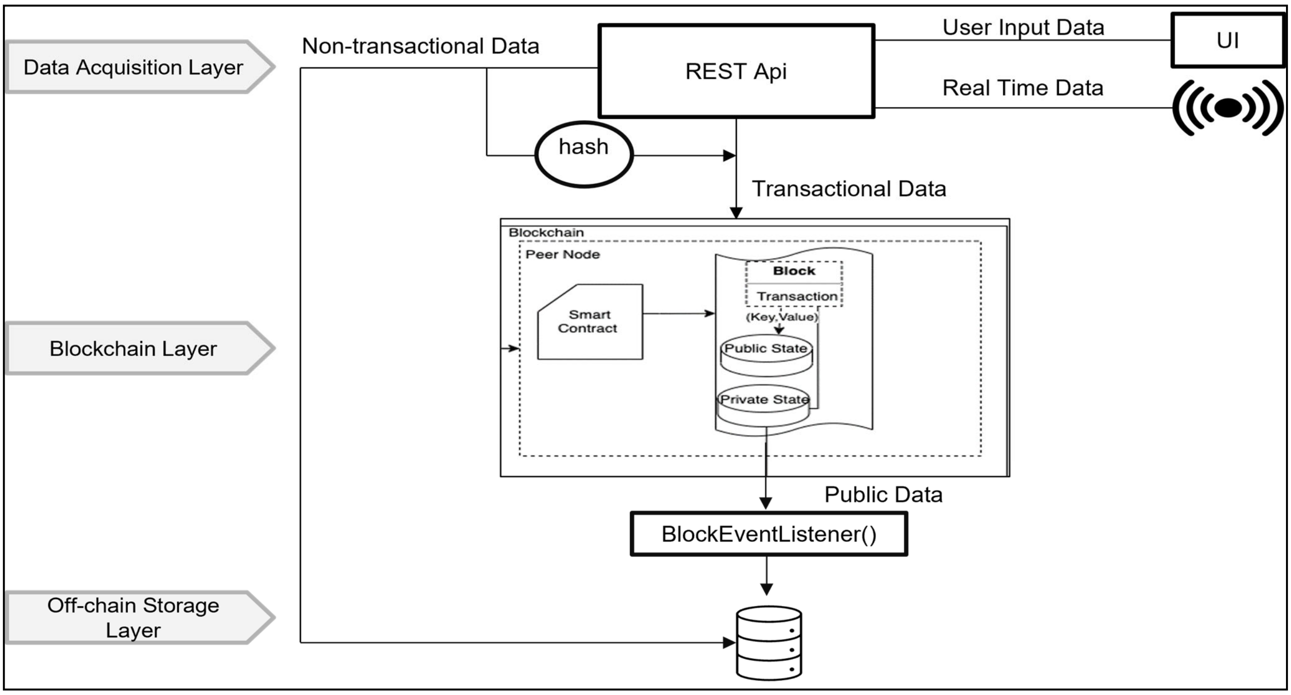
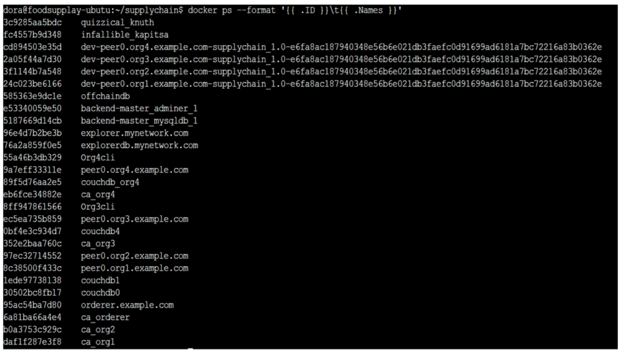


Figure 2 Proposed framework's architecture

The official Hyperledger fabric test network is used as a development foundation of the blockchain network. This test network is mainly created to aid developers and researchers in learning and testing purposes. It is built based on docker compose with two organization peers and an ordering service node. The test network was extended and modified to meet the requirements of the proposed framework.
Figure 3 shows the components of the implemented blockchain network.



```
dora@foodsupplay-ubutu:~/supplychain$ docker ps --format '{{ .ID }}\t{{ .Names }}'
3c9285aa5bdc    quizzical_knuth
fc4557b9d348    infallible_kapitsa
cd894503e35d    dev-peer0.org4.example.com-supplychain_1.0-e6fa8ac187940348e56b6e021db3faefc0d91699ad6181a7bc72216a83b0362e
2a05f44a7d30    dev-peer0.org3.example.com-supplychain_1.0-e6fa8ac187940348e56b6e021db3faefc0d91699ad6181a7bc72216a83b0362e
3f1144b7a548    dev-peer0.org2.example.com-supplychain_1.0-e6fa8ac187940348e56b6e021db3faefc0d91699ad6181a7bc72216a83b0362e
24c023be6166    dev-peer0.org1.example.com-supplychain_1.0-e6fa8ac187940348e56b6e021db3faefc0d91699ad6181a7bc72216a83b0362e
585363e9dc1e    offchaindb
e53340059e50    backend-master_adminer_1
5187669d14cb    backend-master_mysqldb_1
96e4d7b2be3b    explorer.mynetwork.com
76a2a859f0e5    explorerdb.mynetwork.com
55a46b3db329    Org4cli
9a7eff33311e    peer0.org4.example.com
89f5d76aa2e5    couchdb_org4
eb6fce34882e    ca_org4
8ff947861566    Org3cli
ec5ea735b859    peer0.org3.example.com
0bf4e3c934d7    couchdb4
352e2baa760c    ca_org3
97ec32714552    peer0.org2.example.com
8c38500f433c    peer0.org1.example.com
1ede97738138    couchdb1
30502bc8fb17    couchdb0
95ac54ba7d80    orderer.example.com
6a81ba66a4e4    ca_orderer
b0a3753c929c    ca_org2
daf1f287e3f8    ca_org1
```

Figure 3 Blockchain network docker containers

Figure 3 shows the following components:

- One Orderer: orderer.example.com
- 4 Peers, one for each organization, peer0.org1.example.com peer0.org2.example.com peer0.org3.example.com peer0.org4.example.com
- 5 Certificate Authorities: ca_org1, ca_org2, ca_ org3, ca_org4, and ca_orderer
- 4 CouchDB instances for each organization which form the local storage for each node

Supply chain users are registered through the certificate authorities (CAs) according to their organizations: (org1 for supplier, org2 for producer, org3 for retailer, and org4 for transporter). CAs are responsible for issuing users certificates that form the identities that are used for interacting with the system.

3. **Off-chain Storage Layer:** this layer is used as secondary storage to store both public data on-chain and off chain or non-transactional data, which is inserted directly through the REST API. The purpose of this storage is to support blockchain scalability as some large-size data such as files, pictures, and additional text data that are not needed in the product track and trace process could be stored off-chain. Moreover, it supports the performance of data queries since blockchain by nature has a limited performance compared to traditional database systems.

In fact, non-transactional data is not transmitted or stored on the tamper-proof ledger. Hence, the integrity of such data is not preserved. The proposed framework solves this issue by hashing the non-transactional data first on the application level, and then sending and storing the hash along with other product data as a transaction on the ledger, while the actual values are sent to the off-chain storage. This layer uses a NoSQL database: CouchDB.

To ensure the integrity of the off-chain data, we provide a comparison mechanism that compares the data stored on the hash of the off-chain data with the stored hash on-chain.

Table 3 shows the other tools and technologies used in the implementation of the proposed framework.

Table 3 Framework implementation tools

Tool	Description
Ubuntu 20.04 LTS	Open-source Linux distribution based on Debian [32]. It is used to host the framework.
Node.js	A JavaScript server-side runtime environment [33]. It is used to create an API that allows the interaction between different framework layers.
Visual Studio Code	A standalone source-code editor that is used to edit, debug, build, and deploy different applications [34]
Docker	A containerization platform that enables programmers to bundle applications into standardized executable containers that simplify delivery of distributed applications [35]. It is used to deploy the different blockchain nodes as well as another framework component.

4.3. Private Data Collection

Hyperledger fabric supports the feature of private data collection, which can be used as an additional security layer that maintains some data confidential to a subset of organization peers within a network. Basically, private data uses the GossipProtocol which is a peer-to-peer communication protocol that is usually used in distributed systems to disseminate data to a group member [36].

Private data is implemented as a collection, in which each collection is created through a JSON format definition file and managed through a customized policy that contains the authorized peer to access the private data. In fact, the private data values are stored only on the private local database of the authorized peer, in which only the hashes of the data are stored on the ledger and visible to all other peers as evidence of the existence of the data, which also preserves its integrity. **오류! 참조 원본을 찾을 수 없습니다.** shows the ledger component without and with private data collection [36].

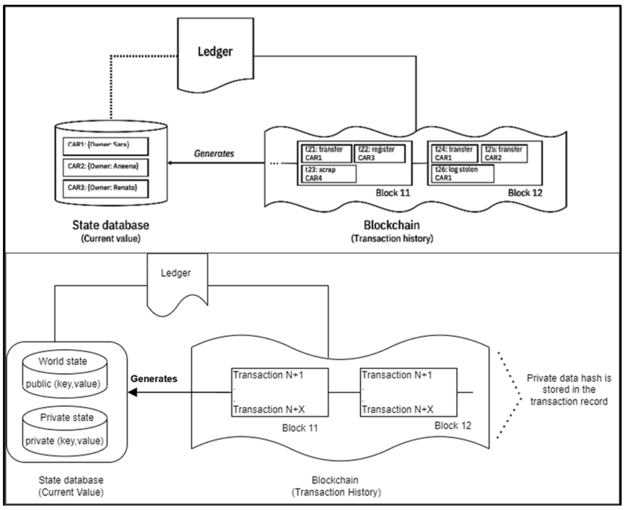
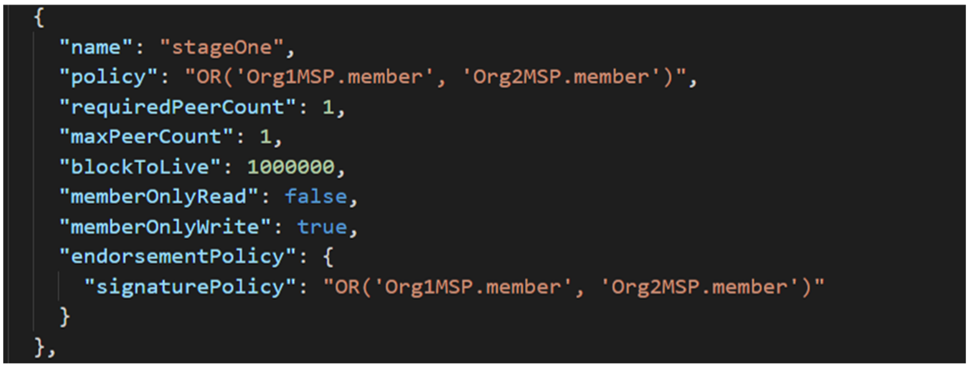


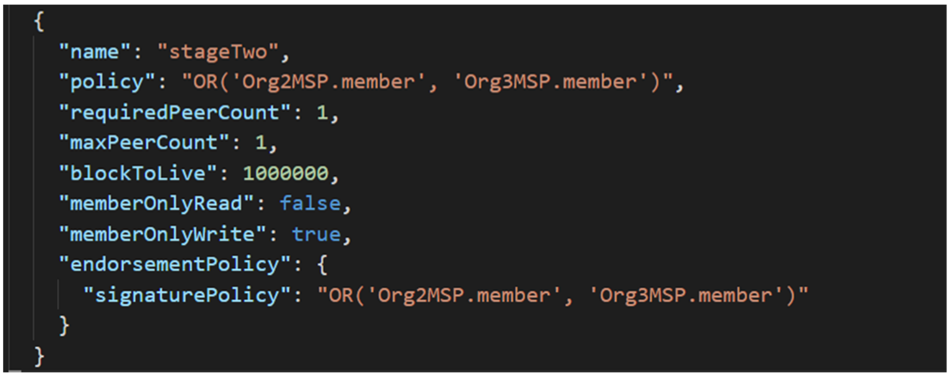
Figure 4 Ledger without private data collection (top) vs ledger with private data collection (bottom)

In this framework, product price is considered an example of confidential supply chain data that must be kept private between each stakeholder. There are two private data collections that have been implemented: the first collection is called “stageOne” and is shared between the first organization which represents the supplier and the second organization which represents the producer. The second collection is called “stageTwo” and is shared between the second organization which represents the producer and the third organization which represents the retailer. Figures 5 and 6 show the collection definitions and policy.



```
{
  "name": "stageOne",
  "policy": "OR('Org1MSP.member', 'Org2MSP.member')",
  "requiredPeerCount": 1,
  "maxPeerCount": 1,
  "blockToLive": 1000000,
  "memberOnlyRead": false,
  "memberOnlyWrite": true,
  "endorsementPolicy": {
    "signaturePolicy": "OR('Org1MSP.member', 'Org2MSP.member')"
  }
},
```

Figure 5 stageOne private data collection definition



```
{
  "name": "stageTwo",
  "policy": "OR('Org2MSP.member', 'Org3MSP.member')",
  "requiredPeerCount": 1,
  "maxPeerCount": 1,
  "blockToLive": 1000000,
  "memberOnlyRead": false,
  "memberOnlyWrite": true,
  "endorsementPolicy": {
    "signaturePolicy": "OR('Org2MSP.member', 'Org3MSP.member')"
  }
}
```

Figure 6 stageTwo private data collection definition

According to blockchain context and transaction flow in Hyperledger fabric, data is sent from the client application and invokes some chaincode functions as arguments. These arguments are sent to the endorsing and orderer peers along with other transaction data, which later will be stored permanently on the ledger. However, this behavior reveals private data during the endorsement stage. This requires the implementation of a secure mechanism to keep this private data totally confidential at every transaction stage from all unauthorized organizations.

Transient data is a data input mechanism in which data can be sent and used by the chaincode without being stored within the transaction record. Using transient data along with private data achieves the confidentiality of data on both processing and storage levels. In this prototype, getTransient() chaincode API [37] is used to send private data as a transient field which is saved during the endorsement in a peer's temporary local storage. Consequently, once the transaction that holds the hash of this private data is committed and verified, the value in the transient storage will be copied and stored in the local peer's database and then deleted from the transient storage.

4.4. IoT Device Integration

IoT devices are implemented as client nodes in which these devices are used only for transmitting data without storing a copy of the blockchain ledger or chaincode. Moreover, we implement an access control list in order to prevent these devices from reading the ledger.

The implementation is done using Nodejs SDK to send IoT sensor data through REST API to the blockchain network using the identity of the linked blockchain node. Each IoT device should have an owner from the blockchain users. In our supply chain model, since the devices used are temperature and location sensors, they belong to the transporter, and all data is submitted to the blockchain using the transporter's identity. In fact, since blockchain data will not be stored in IoT devices, this will suit the constrained storage and power of IoT devices. Moreover, it will eliminate some IoT-related issues such as information disclosure, tampering, and data leaks. Furthermore, linking IoT devices with a specific blockchain organization allows better monitoring and auditing of IoT device behavior which will speed up the detection of any faults and help mitigate the associated risk in a timely manner.

The flow chart in Figure 7 represents the read/write function of the proposed framework.

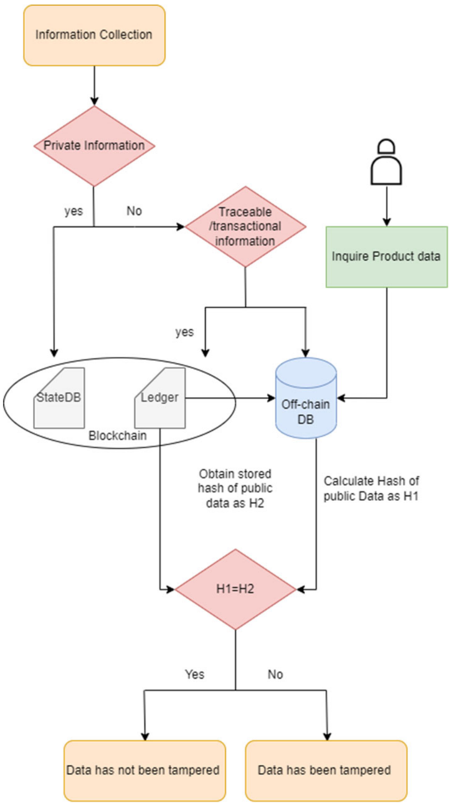

Information Collection
Private Information
yes
No
Traceable /transactional information
Inquire Product data
yes
StateDB
Ledger
Blockchain
Off-chain DB
Obtain stored hash of public data as H2
Calculate Hash of public Data as H1
H1=H2
Yes
No
Data has not been tampered
Data has been tampered

Figure 7 Read and write function flow chart

5. Results and Evaluation

This section presents the testing and evaluation results of the proposed framework. Basically, the framework is tested according to predefined test cases based on supply chain data flow to test its functionality, security, and performance.

5.1. Test Results

The results according to each supply chain organization are discussed below.

Supplier: only the supplier is allowed to initiate the product life cycle by creating a product and adding the associated on-chain and off-chain data through the UI. On-chain data along with the hash of off-chain data are stored on the blockchain.

Producer: the producer can only request a specific product from a supplier. The producer is allowed to input the required quantity and price. The proposed price is private data that is only visible and stored on the local private database of the authorized nodes (supplier and producer). When the supplier accepts the request, product ownership is transferred to the producer.

Retailer: the retailer can only request producer-owned products. The proposed price is only visible and stored in the local private databases of the producer and retailer. At this stage, the producer accepts the request and assigns a transporter to deliver the product to the retailer.

Transporter: the transporter can update the product delivery status . The transported is only allowed to view a product's public data.

IoT Data: IoT-simulated sensors are set in this prototype to automatically transmit temperature and location data periodically when the product assigns the request to the transporter until the transporter changes the request status to delivered. IoT data is sent using the transporter's identity. IoT devices cannot access and read any blockchain data.

5.2. Evaluation

Two aspects are considered to evaluate the proposed framework: security and performance.

5.2.1 Security

Security is evaluated based on the confidentiality, integrity, and availability triad model.

Confidentiality: data confidentiality is preserved on the proposed framework at different levels. ***First***, the proposed framework is developed using a permissioned blockchain in which only a predefined set of users are allowed to join the network and perform read and write operations on the ledger. ***Second***, the proposed framework allows different network participants to share part of the data privately using private data collection in which data is transmitted and stored in the authorized peer's network only without performing chaincode level encryption because it decreases blockchain performance. ***Third***, the proposed framework implements an access control mechanism to restrict IoT devices from performing any read operation on ledger data. This prevents any data leakage in case an IoT device becomes compromised.

Integrity: due to the nature of blockchain, the integrity of data on the chain is preserved through the block hashing feature. Thus, the proposed framework works in such a way that every transaction is written permanently on the ledger, and transaction data cannot be tampered with or changed and will be written on the chain forever.

Availability: the proposed framework is a decentralized system that operates globally across several computers and allows access to all nodes. The ledger data is stored on every network node without any modification allowed, so even if some of them go down, product tracing data will still be available.

Off-chain Data: off-chain storage is mainly created to support blockchain storage scalability and enhance the performance of blockchain data retrieval. The off-chain storage is not protected against alteration. Hence, the proposed framework supports the integrity of the off-chain data by hashing it at the application level and submitting the hash along with other product data as blockchain transactions.

IoT Devices and Security: IoT devices are implemented in this framework as client nodes that are connected to the blockchain network through the transporter peer's identity provided by Hyperledger fabric CA. This provides some critical security requirements for IoT devices. ***First***, IoT device ownership, in which all devices are connected as part of a trusted user identity which is responsible for the IoT devices' data and activities. In the case of any abnormal behavior or security violations, the attack is easier to identify and mitigate. ***Second***, due to the nature of the blockchain, all IoT data and activities are logged in the blockchain which enhances auditing and detection of IoT device behaviors. ***Third***, no ledger data is stored on the IoT devices' storage and access to the ledger is restricted as mentioned in the confidentiality section above, which reduces the possibility of information disclosure.

5.2.2. Performance

Performance is a major factor that determines the feasibility of any solution. To verify the feasibility of the proposed framework, the research activities included a comprehensive test of system performance using a Hyperledger caliper [38] which is a blockchain performance benchmark tool developed by the Hyperledger foundation to be used with Hyperledger projects to test and generate network performance indicators using predefined use cases. To evaluate the performance of the proposed framework, two metrics are considered, throughput and latency of both create and query product functions.

Tables 4 and 5 show the testing result using different send rates for create product and query product, respectively.

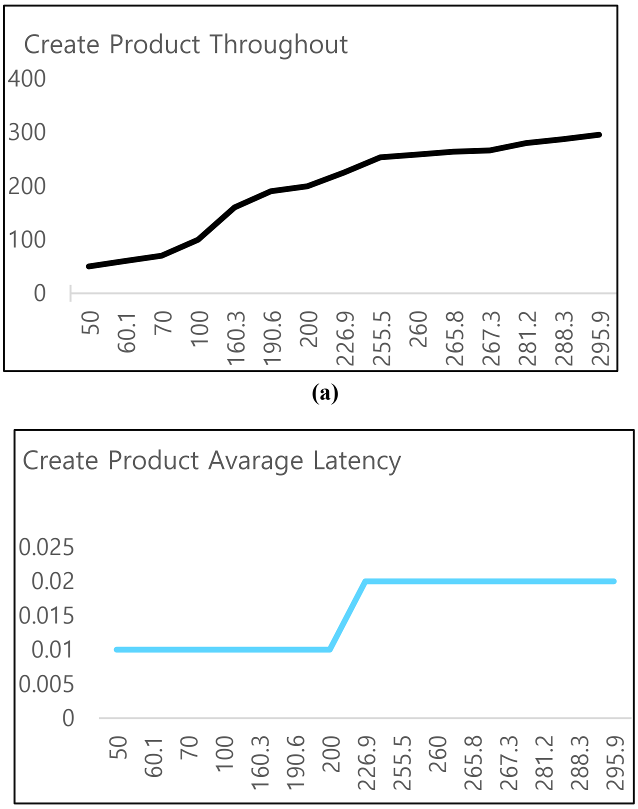
Table 4 Create functions performance

Function	**Send Rate (TPS)**	**Avg Latency (S)**	**Throughput (TPS)**
Create Product ()	50.0	0.01	50.0
	60.1	0.01	60.1
	70.0	0.01	70.0
	100.0	0.01	100.0
	160.3	0.01	160.0
	190.6	0.01	190.0
	200.0	0.01	199.3
	255.5	0.02	253.8
	260.0	0.02	258.3
	265.8	0.02	263.7
	277.5	0.02	276.6
	281.2	0.02	279.7
	288.3	0.02	286.8
	295.9	0.02	295.1

Table 5 Query function performance

Function	**Send Rate (TPS)**	**Avg Latency (S)**	**Throughput (TPS)**
Query Product ()	50.0	0.01	50.0
	57.0	0.01	57.0
	70.2	0.01	70.1
	100.1	0.01	100.0
	150.3	0.01	150.0
	200.7	0.01	200
	247.6	0.01	246.3
	250.8	0.02	250.0
	281.2	0.02	279.7
	287.2	0.02	285.1
	290.4	0.02	288.2
	297.3	0.02	295.2
	302.3	0.02	301.9
	314.1	0.02	311.5
	328.0	0.02	327.6

It is noted that the throughput is linearly related to the send rate for writing operation which is 295 transactions per second (TPS), while the average latency is 0.01s when the send rate is under 100 TPS and 0.02s for higher send rates as shown in Figure 8.

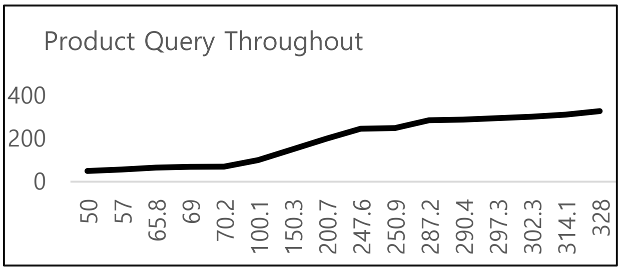


(a)

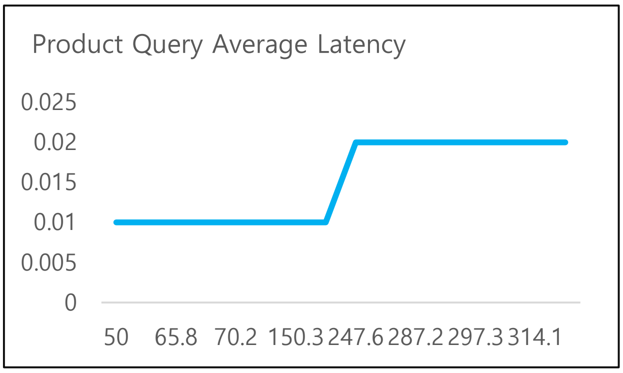
(b)

Figure 8 (a) Create function throughput, (b) create product latency

Furthermore, the throughput is linearly related to the send rate for the read function. The maximum throughput for the writing operation is 295 TPS, while the average latency is 0.01s when the send rate is under 100 TPS and 0.02s for higher send rates as shown in Figure 9.
Compared to the literature reviewed in Section 2, the proposed framework achieved better performance by 31.2% and 18%, which represents increases in read operation throughput and write operation throughput, respectively.



(a)



(b)

Figure 9 (a) Query function throughput, (b) query product latency

Furthermore, it decreased the write operation latency by 83.3%.

6. Conclusions and Future Work

Supply chain complexity has increased because of globalization and modern business, which require additional management efforts. IT infrastructure and technologies enhance SCM systems which in turn provides a better business outcome and supports better decision-making. However, integrating digital technologies increases the many challenges related to supply chain security and functionality. This paper aims to enhance the security of IoT-based SCM systems by proposing a blockchain-based system that fits both IoT and supply chain requirements. To achieve this, the requirements of supply chain systems are gathered and analyzed. The limitations of IoT are identified, and the role of blockchain technology in these systems is discussed. Based on that, a set of design criteria are identified and implemented in order to propose a secure and scalable blockchain-based IoT-SCM system. Testing results show that the proposed system maintains confidentiality, integrity, and availability of on-chain and off-chain supply chain data. Moreover, it allows the integration of IoT devices without consuming their limited resources. Performance analysis indicated that the proposed system has a better performance compared to those in the reviewed literature with increases of 31.2% and 18% in read operation throughput and write operation throughput, respectively. There was also a decrease of 83.3% in write operation latency.

The future direction of the proposed framework includes testing the performance of the integrated IoT devices to ensure the efficiency of the proposed framework on each component and limiting the uploaded IoT data on-chain. This might be done by applying some policy on the data that has to be stored permanently such as storing only the last transmitted data or data values that violate the regulations. Furthermore, the security of the proposed framework might be enhanced in the future by applying data obfuscation techniques such as salting on private data hashes, since short data might be predictable or vulnerable to some attacks such as dictionary attack.

.

References

1. M. Ben-Daya, E. Hassini, and Z. Bahroun, "Internet of things and supply chain management: a literature review," *International Journal of Production Research*, vol. 57, no. 15–16, pp. 4719–4742, Nov. 2017, doi: 10.1080/00207543.2017.1402140. [Online]. Available: http://dx.doi.org/10.1080/00207543.2017.1402140
2. J. Zhang, H. Chen, L. Gong, J. Cao, and Z. Gu, "The current research of IoT security," *2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC)*, 2019, doi: 10.1109/DSC.2019.00059.
3. H. Boyes, "Cybersecurity and cyber-resilient supply chains," *Technology Innovation Management Review*, vol. 5, no. 4, pp. 28–34, Apr. 2015, doi: 10.22215/timreview/888. [Online]. Available: http://dx.doi.org/10.22215/timreview/888
4. M. Stevenson and J. Busby, "An exploratory analysis of counterfeiting strategies," *International Journal of Operations & Production Management*, vol. 35, no. 1, pp. 110–144, Jan. 2015, doi: 10.1108/ijopm-04-2012-0174. [Online]. Available: http://dx.doi.org/10.1108/ijopm-04-2012-0174
5. P. Urien, "Blockchain IoT (BIoT): a new direction for solving internet of things security and trust issues," *2018 3rd Cloudification of the Internet of Things (CIoT)*, pp. 1–4, 2018, doi: 10.1109/CIOT.2018.8627112.
6. A. Panarello, N. Tapas, G. Merlino, F. Longo, and A. Puliafito, "Blockchain and IoT integration: a systematic survey," *Sensors*, vol. 18, no. 8, p. 2575, Aug. 2018, doi: 10.3390/s18082575. [Online]. Available: http://dx.doi.org/10.3390/s18082575
7. Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: architecture, consensus, and future trend," *2017 IEEE International Congress on Big Data (BigData Congress)*, pp. 557–564, 2017, doi: 10.1109/BigDataCongress.2017.85.
8. F. Betti and I. Cronin, "Visibility and traceability: the twin engines of sustainable supply chains," *World Economic Forum*, 2020.
9. Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: a survey," *International Journal of Web and Grid Services*, 2018, doi: 10.1504/IJWGS.2018.095647.
10. SCOR Model, *SCOR*. [Online]. Available: https://scor.ascm.org/. [Accessed: Mar. 2022].
11. M. Windelberg, "Objectives for managing cyber supply chain risk," *International Journal of Critical Infrastructure Protection*, vol. 12, pp. 4–11, Mar. 2016, doi: 10.1016/j.ijcip.2015.11.003. [Online]. Available: http://dx.doi.org/10.1016/j.ijcip.2015.11.003
12. M. Hudnurkar, S. Jakhar, and U. Rathod, "Factors affecting collaboration in supply chain: a literature review," *Procedia - Social and Behavioral Sciences*, vol. 133, pp. 189–202, May 2014, doi: 10.1016/j.sbspro.2014.04.184. [Online]. Available: http://dx.doi.org/10.1016/j.sbspro.2014.04.184
13. R. Gupta, *Hands-On Cybersecurity with Blockchain*. Packt Publishing Ltd, 2018.
14. N. Vyas, A. Beije, and B. Krishnamachari, *Blockchain and the Supply Chain*. Kogan Page Publishers, 2019.
15. J. K. Deane, C. T. Ragsdale, T. R. Rakes, and L. P. Rees, "Managing supply chain risk and disruption from IT security incidents," *Operations Management Research*, vol. 2, no. 1–4, pp. 4–12, Apr. 2009, doi: 10.1007/s12063-009-0018-2. [Online]. Available: http://dx.doi.org/10.1007/s12063-009-0018-2.
16. H. Hasan, E. AlHadhrami, A. AlDhaheri, K. Salah, and R. Jayaraman, "Smart contract-based approach for efficient shipment management," *Computers & Industrial Engineering*, vol. 136, pp. 149–159, Oct. 2019, doi: 10.1016/j.cie.2019.07.022. [Online]. Available: http://dx.doi.org/10.1016/j.cie.2019.07.022
17. A. Arena, A. Bianchini, P. Perazzo, C. Vallati, and G. Dini, "Ruschetta: an IoT blockchain-based framework for certifying extra virgin olive oil supply chain," *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, Jun. 2019, doi: 10.1109/SMARTCOMP.2019.00049.
18. K. Botcha, V. Chakravarthy, and Anurag, "Enhancing traceability in pharmaceutical supply chain using internet of things (IoT) and blockchain," *2019 IEEE International Conference on Intelligent Systems and Green Technology (ICISGT)*, Jun. 2019, doi: 10.1109/ICISGT44072.2019.00025.
19. G. Rathee, A. Sharma, Rajiv Kumar, and R. Iqbal, "A secure communicating things network framework for industrial IoT using blockchain technology," *Ad Hoc Networks*, Jun. 2019, doi: 10.1016/j.adhoc.2019.101933.
20. S.-S. Kuo and Wei-Tsung Su, "A blockchain-indexed storage supporting scalable data integrity in supply chain traceability," *2020 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pp. 348–349, Aug. 2020, doi: 10.1109/SmartIoT49966.2020.00064.
21. Johan Älvebrink and Maria Jansson, "Investigation of blockchain applicability to Internet of Things within supply chains," *(Dissertation)*, 2018. [Online]. Available:

http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-357258. [Accessed: Nov. 19, 2021]

22. A. Akram and P. Bross, "Trust, privacy and transparency with blockchain technology in logistics," *MCIS 2018 Proceedings*, vol. 17, 2018, [Online]. Available: https://aisel.aisnet.org/mcis2018/17.
23. K. Winata, "Blockchain based data sharing system for supply chain," *International Journal Of Engineering Research & Technology (IJERT)*, vol. 09, no. 11, Nov. 2020, doi: 10.17577/IJERTV9IS110272.
24. I. Surjandari, H. Yusuf, E. Laoh, and R. Maulida, "Designing a Permissioned Blockchain Network for the Halal Industry using Hyperledger Fabric with multiple channels and the raft consensus mechanism," *Journal of Big Data*, vol. 8, no. 1, Jan. 2021, doi: 10.1186/s40537-020-00405-7. [Online]. Available: http://dx.doi.org/10.1186/s40537-020-00405-7
25. M. El Maouchi , O. Ersoy , and Z. Erkin, "Decouples: a decentralized, unlinkable and privacy-preserving traceability system for the supply chain," *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pp. 364–373, Apr. 2019, doi: https://doi.org/10.1145/3297280.3297318.
26. J. Shi, D. Yi, and Jian Kuang, "Pharmaceutical supply chain management system with integration of IoT and blockchain technology," *International Conference on Smart Blockchain*, vol. 11911, pp. 97–108, 2019, doi: 10.1007/978-3-030-34083-4_10.
27. X. Yang, M. Li, H. Yu, M. Wang, D. Xu, and C. Sun, "A trusted blockchain-based traceability system for fruit and vegetable agricultural products," *IEEE Access*, vol. 9, pp. 36282–36293, 2021, doi: 10.1109/access.2021.3062845. [Online]. Available: http://dx.doi.org/10.1109/access.2021.3062845
28. J. Flapper, "User access control on the blockchain for supply chain visibility," *Thesis, University of Twente*, 2019, doi: http://essay.utwente.nl/78800/.
29. "BigchainDB • • The blockchain database.," *BigchainDB*. [Online]. Available: https://www.bigchaindb.com/. [Accessed: Nov. 19, 2022]
30. Hyperledger Fabric, *Hyperledger Foundation*. [Online]. Available: https://www.hyperledger.org/use/fabric. [Accessed: Nov. 2022]
31. S. Zafar, S. F. U. Hassan, A. Mohammad, A. A. Al-Ahmadi, and N. Ullah, "Implementation of a distributed framework for permissioned blockchain-based secure automotive supply chain management," *Sensors*, vol. 22, no. 19, p. 7367, Sep. 2022, doi: 10.3390/s22197367. [Online]. Available: http://dx.doi.org/10.3390/s22197367
32. Enterprise Open Source and Linux | Ubuntu, *Ubuntu*. [Online]. Available: https://ubuntu.com/. [Accessed: Nov. 19, 2021]
33. Node.js, *Node.js*. [Online]. Available: https://nodejs.org. [Accessed: Mar. 19, 2021]
34. Visual Studio: IDE and Code Editor for Software Developers and Teams, *Visual Studio*, Nov. 18, 2022. [Online]. Available: https://visualstudio.microsoft.com. [Accessed: Feb. 2022]
35. Docker: Accelerated, Containerized Application Development, *Docker*, May 10, 2022. [Online]. Available: https://www.docker.com/. [Accessed: Aug. 19, 2022].
36. N. Gaur, A. O'Dowd, P. Novotny, L. Desrosiers, V. Ramakrishna, and S. A. Baset, *Blockchain with Hyperledger Fabric*. 2020.
37. Private Data — *hyperledger-fabricdocs main documentation*. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/latest/private-data-arch.html. [Accessed: Feb. 19, 2022]
38. Hyperledger Caliper, *Hyperledger Caliper | Caliper blockchain performance benchmark framework*. [Online]. Available: https://hyperledger.github.io/caliper/. [Accessed: Sep. 2022].