symmetry

MDPI

Article

Towards Convergence of IoT and Blockchain for Secure Supply Chain Transaction

A S M Touhidul Hasan [1,2,3], Shabnam Sabah [1], Rakib Ul Haque [3,4], Apubra Daria [3], Abdur Rasool [2,5] and Qingshan Jiang [2,*]

1 Department of Computer Science and Engineering, University of Asia Pacific, Dhaka 1205, Bangladesh; touhid@uap-bd.edu (A.S.M.T.H.); tarannum.cse@gmail.com (S.S.)
2 Shenzhen Key Laboratory for High Performance Data Mining, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; rasool@siat.ac.cn
3 Institute of Automation Research and Engineering, Dhaka 1205, Bangladesh; rakibulhaqueraj@mails.ucas.ac.cn (R.U.H.); apubra@iar-e.com (A.D.)
4 School of Computer Science & Technology, University of Chinese Academy of Sciences, Shijingshan District, Beijing 100049, China
5 Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Shenzhen 518055, China
* Correspondence: qs.jiang@siat.ac.cn; Tel.: +86-186-6532-6469

Abstract: Supply chain management (SCM) is essential for a company's faster, efficient, and effective product life cycle. However, the current SCM systems are insufficient to provide product legitimacy, transaction privacy, and security. Therefore, this research proposes a secure SCM system for the authenticity of the products based on the Internet of Things (IoT) and blockchain technology. The IoT-enabled Quick Response (QR) scanner and the blockchain-integrated distributed system will allow all the SCM stakeholders to begin secure and private transactions for their products or services. Resulting, the consumer will receive an authentic and genuine product from the original producer. A lightweight asymmetric key encryption technique, i.e., elliptic curve cryptography (ECC) and Hyperledger Fabric-based blockchain technology with on-chain smart contracts are applied for distributed IoT devices to make the authentication process faster and lighter. Each SCM stakeholder is registered by the service provider and receives corresponding public and private keys, which will be used for the authentication process of the participants and IoT devices. The authenticated QR scanner records all transactions on the blockchain. Consequently, there will be no human intervention for the SCM transactions. The security and scalability analysis demonstrates that the proposed system is more secure and robust than other state-of-the-art techniques.

Keywords: supply chain; internet of things; blockchain; asymmetric encryption; authentication and security

Citation: Hasan, T.A.S.M.; Sabah, S.; Haque, R.U.; Daria, A.; Rasool, A.; Jiang, Q. Towards Convergence of IoT and Blockchain for Secure Supply Chain Transaction. *Symmetry* **2022**, *14*, 64. https://doi.org/10.3390/sym14010064

Academic Editors: Chin-Ling Chen, Zi-Yi Lim, Xingsi Xue and Chi-Hua Chen

Received: 30 November 2021
Accepted: 16 December 2021
Published: 3 January 2022

1. Introduction

The internet and technology have been developed so rapidly that the whole world is experiencing the fourth industrial revolution (Industry 4.0) [1] in all aspects of humankind, where the Internet of Things (IoT) [2] plays a significant role for its diverse adoption. IoT is a network of interlinked physical objects (e.g., devices, machines, and appliances) installed with sensors, software, and electronics, provided with unique identifiers. IoT sensors also possess the capacity to exchange data over the internet without human intervention. It can create information about the associated objects, examine them and make decisions. It has enormous potential to give various elating services across numerous spaces from industry, healthcare [3], smart home [4], smart cities, social media, and supply chain. IoT devices have revolutionized the supply chain management (SCM) system [5].

SCM is the management of the movement of goods through various parties like manufacturers, distributors, retailers, and customers [3]. It helps to check the traversal

of products and information without any complexities. A supply chain involves a series of steps to get a product or service to the customer. The steps include moving and transforming raw materials into finished products, transporting and distributing them to the end-user.

IoT devices can be connected to a product to confirm the product's authenticity, investigate the origin and quality. Moreover, IoT devices can ensure real-time tracking, traceability, and visibility of a product in the supply chain. A recent survey reveals that Australian retailers have integrated IoT devices into their supply chain. It includes internet-based barcode technology, sensors and scanners, palm-held tablets/smart devices, smartphones, mobile apps, GPS-based location awareness, and Internet-based security and surveillance system [6].

There is no doubt regarding the advantages of the IoT in the supply chain. Despite the benefits, some concerns are related to the IoT integrated supply chain. The IoT devices generate a large amount of data stored in a centralized server, i.e., in a cloud as a plaintext. As a result, there is a chance that the centralized server might act dishonestly and make fallacious use of users' sensitive data. There is a severe threat related to the privacy and security of user data in the centralized IoT infrastructure [7]. Even most of the existing supply chains are not IoT integrated, and because of human intervention [8], there is a high risk in the privacy and security of product and user's data.

Besides the above-discussed articles, there are some other investigations where IoT and blockchain [9] are integrated into the supply chain, whereas there are no studies that focus on the incorporation of asymmetric key encryption technique elliptic curve cryptography (ECC), IoT, and supply chain. Moreover, none of the earlier studies which are discussed in Section 2 focuses on key distributions and key agreements for authenticating IoT devices. Blockchain is a decentralized and distributed network of peers that shares the same ledger of transactions connected with the system without any central server. The transaction records in the blockchain ledger are immutable, and therefore, it assures authenticity, transparency, traceability, security, and visibility among supply chain entities. The immutable nature of the blockchain platform ensures the SCM transactions data authenticity and security, but it does not ensure data privacy. Therefore, users' sensitive data needs to be protected from disclosure. Due to the resource limitations (i.e., small memory, limited battery power, and insufficient processing capability) of the IoT device, conventional PC-based cryptographic solutions are not appropriate for most IoT devices [10]. Therefore, a lightweight cryptographic protocol is required for the system.

This research converges IoT, lightweight asymmetric key cryptography, i.e., ECC, and Hyperledger fabric for secure and trusted supply chain transactions to mitigate the existing supply chain problems. A lightweight key agreement scheme based on ECC has been introduced to ensure the authenticity of IoT devices. Hyperledger fabric assures faster and private supply chain transactions between participating entities. All products or services carry a quick response (QR) code from their production. The proposed system will scan QR codes with an IoT-enabled QR scanner, whereas the transaction data will be stored into the blockchain automatically and securely. Every participant's (e.g., manufacturer, distributor, and retailer) QR scanner will be registered through the lightweight authentication process in the blockchain network. After the registration and successful mutual authentication between the IoT device of two entities, the product information scanned by the QR scanner is stored in the blockchain. The proposed approach serves as a peer-to-peer, trusted distributed supply chain that introduces the product's real-time tracking and traceability and guarantees product information authenticity and confidentiality with an authenticated IoT device. Integration of IoT in the blockchain-based supply chain will enhance the supply chain's flexibility, traceability, transparency, real-time audibility, autonomy, and transaction privacy.

The main contributions of this paper are as follows:

- IoT and Blockchain are used to reduce human intervention at the time of recording the supply chain transaction;

- Asymmetric key encryption technique ECC based Key distribution and key agreement are developed in SCM. ECC is used for managing the cryptographic operations and also for lightweight authentication of entities;
- Hyperleadger fabric based blockchain technology will ensure the transaction data privacy and security;
- Security and Privacy analysis illustrate the efficiency of the proposed method.

The rest of the article is structured as follows. Related works are analyzed in Section 2. Preliminaries, System Overview, and Model Construction are delineated in Sections 3–5, respectively. Section 6 illustrates the Performance Evaluation. Finally, Section 7 concludes this article.

2. Related Work

This section briefly reviews previous works and also discusses their limitations and the novelty of these works.

2.1. Privacy by Design

Security of information with the help of technology design is called privacy by design. This concept can merge privacy at the development and production level. It is better to employ a proactive method for data security before they occur, instead of lingering till the breach happens [11,12]. End-to-end security for the entire lifecycle protection can be achieved by this concept. All data are processed securely and also being destroyed securely when needs are over. Specification of privacy context is necessary to defend user privacy. Recent studies [13–17] determined some privacy terms necessary for cyberspace. They are intruders, receivers, senders. and so on. Pfitzmann and Hansen [15–17] illustrate a setting related to privacy, which specifies the affinity among privacy terms. Moreover, privacy by design is important for information security.

2.2. IoT and Blockchain in Supply Chain

Malik [18] proposed TrustChain, which is a three-layered trust management framework for SCM integrated with blockchain. Tsang [19] presented a blockchain and IoT-enabled food traceability system called BIFTS where incorporates IoT, fuzzy logic, and blockchain for complete traceability of perishable food. Shi [20] designed and developed an IoT and blockchain-integrated pharmaceutical supply chain management system to mitigate the concerns of belief, safety, traceability, and inefficiency. Caro [21] proposed a system for the agricultural food supply chain management, which is a comprehensively decentralized traceability system. It incorporates different IoT sensor devices with the supply chain. Abdel-Basset [22] proposed a framework based on RFID technologies for supply chain management that automate the identification process of products, trace and track products globally.

Cui [23] proposed a Hyperledger Fabric-based blockchain framework to trace and track every electronic chip in the supply chain. All the supply chain entities could benefit from this framework since it helps to preserve the supply chain from forged devices. Cocco [24] proposed a blockchain and IoT-based system for Carasau bread's supply chain management to ensure the product's transparent and auditable traceability. In their suggested system, every supply chain party can check the condition of the products and the agreement to the prescriptive about the hygienic-sanitary circumstances on the chain. Matteo [25] presented a DL-Tags solution based on IoT and blockchain that allows privacy-preserving, decentralized, and verifiable management of commodities labeled with Smart Tags. All the product consumers and stakeholders can check its authenticity without disclosing their identity. Their recommended solution proves the product's source and journey throughout the supply chain while preventing label replication and manipulation. Bhutta [26] proposed a supply chain management framework for agricultural food supply that ensures secure traceability, identification, and real-time tracking of transportation using IoT and Blockchain.

Grida (2020) [27] discusses the uncertainty of evaluating the outcomes of the supply chain based on IoT by blending pathogenic set with Vlse Kriterijumska Optimizacija Kompromisno Resenje and Best-Worst schemes in a judgment-making framework employed for this field. Yadav (2020) [28,29] employs a framework for regulating the performance of SCM for agriculture based on IoT and to develop an IoT-based effective system following natural outbreaks for advancing the coordination mechanism in agriculture supply chain management. Zhang (2020) [30] presents a thorough review of existing SCM-related studies.

Table 1 illustrates the summary of the state-of-the-art techniques with the proposed studies. Most of the investigations utilized IoT and blockchain in SCM, and some of them used cryptographic technologies which are not lightweight. None of them showed the authentication of the entities in terms of privacy and security, and only a few of them focused on transaction data confidentiality. These studies utilized IoT devices to track the products' real-time information, such as product quality and location, without considering security and privacy issues. Some studies employed the transaction privacy module, but it lacks security proof. On the other hand, the proposed framework addresses all the limitations of the studies mentioned earlier, and it is lighter, secure, and faster for supply chain transactions.

Table 1. An overview of existing research on privacy-preserving SCM adopting IoT, cryptography, and Blockchain technologies.

Study	Year	State of the Art Technologies Adoption			Security Parameters Covered	
		Cryptography		Blockchain	Authentication	Confidentiality
		Light Weight	Heavy Weight			
Caro [21]	2018	-	-	✓	-	-
Abdel-Basset [22]	2018	-	-	✓	✓	-
Malik [18]	2019	-	-	✓	-	-
Tsang [19]	2019	-	-	✓	-	-
Shi [20]	2019	-	✓	✓	✓	✓
Cui [23]	2019	-	-	✓	-	-
Matteo [25]	2019	-	-	✓	✓	-
Cocco [24]	2021	-	-	✓	-	-
Bhutta [26]	2021	-	-	✓	✓	-
Proposed	2021	✓	-	✓	✓	✓

3. Preliminaries

This section describes all the notations, which are shown in Table 2 and technologies related to the system.

Table 2. Notations.

Sign	Meanings	Sign	Meanings
$\mathcal{M}$	manufacturer	$\mathcal{D}$	distributor
$\mathcal{R}$	retailer	$\mathcal{C}$	customer
$\mathcal{SP}$	service provider	$\mathcal{ID}$	identity
$\mathcal{PK}$	public key	$\mathcal{SK}$	private Key
$\mathcal{A}$	adversaries	p, q	two large primes
$\mathcal{E}(\,)$	elliptic curve	$\mathcal{F}_p$	finite field
$\mathcal{Z}_q^*$	multiplicative group	$\mathcal{P}$	a generator
n	security parameter	$h()$	hash function
iot	IoT device	$\mathcal{DE}()$	decryption function
r, u, v	nonce	$\mathcal{EN}()$	encryption function
Φ	registration protocol	Γ	authentication protocol
$\mathcal{SEK}$	session key	$\mathcal{MA}$	message
$\mathcal{DS}$	digital signature	$\mathcal{SN}$	$\mathcal{SP}$ sign

3.1. Asymmetric-Key Encryption

Asymmetric encryption technique is known as public-key cryptography. This cryptographic system uses key pairs, i.e., public and private keys. Here, the public keys are declared openly, and private keys are kept secret by the key owners. The formation of the before-mentioned keys depends on cryptographic algorithms based on large prime numbers to build one-way cryptographic algorithms [31]. There are different types of asymmetric-key cryptography such as Diffie Hellman, Rivest–Shamir–Adleman (RSA), Elliptic Curve Cryptography (ECC), ElGamal, and so on. However, ECC is the lightweight asymmetric-key cryptography for data encryption and decryption [32].

3.2. Blockchain and Smart Contract

Blockchain is an immutable distributed ledger technology where the transactions are open to every node of the network associated with a peer-to-peer (P2P) design. It permits untrusted participants to interact and broadcast transactions among each other in a secure way and no trusted third party is needed. Figure 1 represents the smart contract and blockchain system. Blockchain is an ordered list and cryptographic hashes are used to identify each one of the blocks. A chain of blocks is created, where each block references the block that came before it. Every block has a group of transactions [9]. Again, an executable code, which operates on the blockchain in order to aid, execute and dictate the terms of an agreement is known as a smart contract. Its goal is to execute the terms of an agreement automatically if the specific requirements are fulfilled. Its capability fully depends on the programming language, which is utilized for expressing the contract but not on the technology. It has private storage, executable code, and account balance. This study used Practical Byzantine Fault Tolerance (PBFT) [33] for consensus protocol. PBFT is a way for a distributed network to reach the consensus set for the blockchain even if some nodes are malicious. It is used in Hyperledger, in the transaction approval process to avoid malicious decisions. When a Hyperledger transaction is made, the transaction details are sent to the nodes in the network. There are might some nodes that will approve the transactions and some nodes that will not. The majority of nodes have to approve the transaction for the transaction to be completed. To keep the system secure, PBFT requires $3f + 1$ nodes in the system, where f is the maximum number of faulty nodes that the system can tolerate. Therefore, for the group of nodes to make any decision, approval from $2f + 1$ nodes is required.

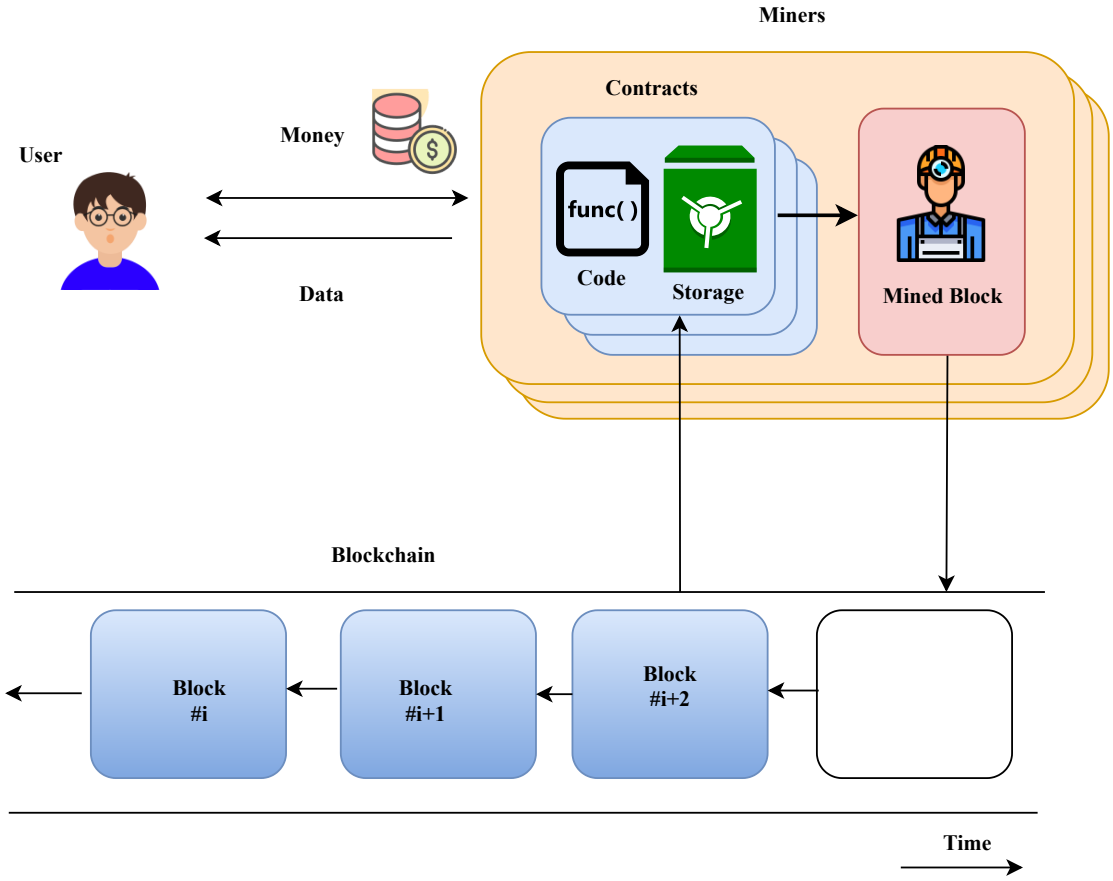


Figure 1. The internal structure of a smart contract for on chain transaction.

3.3. Elliptic Curve Cryptography

Elliptic curve discrete logarithm problem (ECDLP) [34]: Nowadays, 160 bit ECDLP is often used in cryptosystem where $\mathcal{A}$ failed to calculate u, when $Q = uP$ for $P, Q \in E(F_p)$ and $u \in Z_q^*$.

Elliptic curve computational Diffie–Hellmen problem [34]: The length 160 bit ECDLP is secure [34] for that reason $\mathcal{A}$ failed to calculate uvP, where $uP, vP \in E(F_p)$ and $u, v \in Z_q^*$.

4. System Overview

This section discusses the system model, threat model, and security goals.

4.1. System Model

This study envisage blockchain and IoT based data-driven supply chain ecosystem, which is showed in Figure 2. In this system, the registration protocol, consensus mechanism, and authentication protocol are studied in detail. Entities involved in this systems are Manufacturer ($\mathcal{M}$), Distributor ($\mathcal{D}$), Retailer ($\mathcal{R}$), Customer ($\mathcal{C}$), and Service Provider ($\mathcal{SP}$). Their roles are described in Table 3.

Table 3. Individual entities and their roles.

Entities	Roles
Manufacturer	produces the product and sells it to the $\mathcal{D}$
Distributor	purchase the product from $\mathcal{M}$ and sells it to the $\mathcal{R}$
Retailer	buys the the product from $\mathcal{D}$ and sells it to the $\mathcal{C}$
Customer	are the end user who purchase the product from the $\mathcal{R}$
Service Provider	are responsible for registering $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ into the system

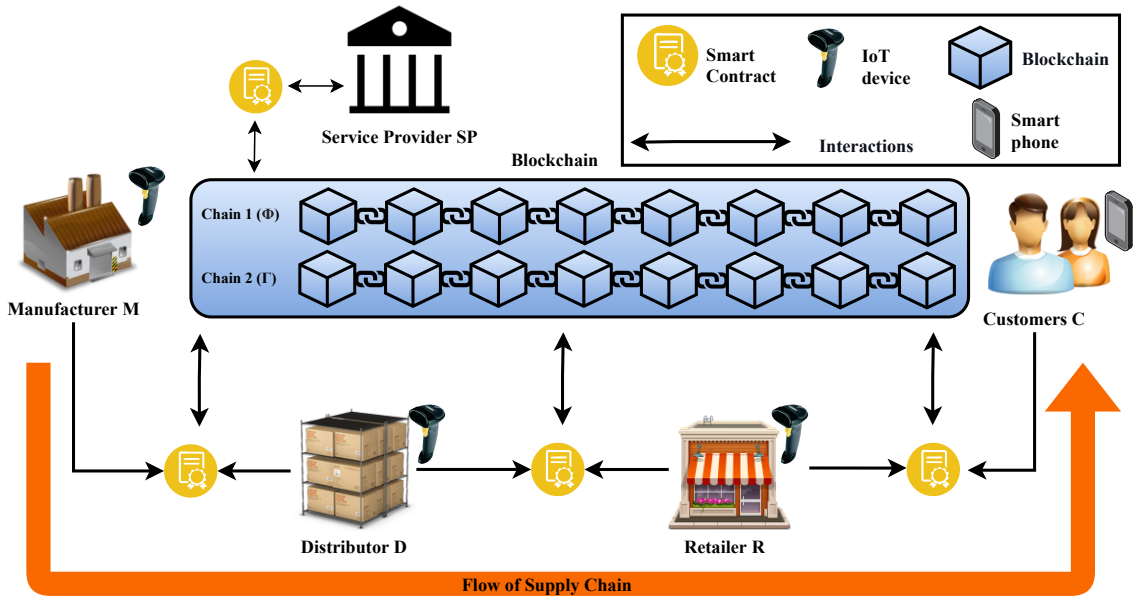


Figure 2. IoT-enabled P2P ecosystem of the proposed supply chain management.

Formally, the proposed system in Figure 2 consist of n number of Manufacturer $\mathcal{M}_{i=1}^{n}$, Distributor $\mathcal{D}_{i=1}^{n}$, Retailer $\mathcal{R}_{i=1}^{n}$, Customer $\mathcal{C}_{i=1}^{n}$, and Service Provider $\mathcal{SP}_{i=1}^{n}$. Each $\mathcal{M}$, $\mathcal{D}$ and $\mathcal{R}$ will have n number of IoT devices, which is $\mathcal{M}_{IoT_{i=1}^{n}}$, $\mathcal{D}_{IoT_{i=1}^{n}}$ and $\mathcal{R}_{IoT_{i=1}^{n}}$, respectively. Participants $\mathcal{M}$, $\mathcal{D}$, $\mathcal{R}$, and $\mathcal{SP}$ perform their task by executing protocols Φ and Γ. Protocol Φ and Γ are used for registration and authentication process, respectively. These protocols can make sure privacy, security, and authenticity of the participants, which is described in Section 5.

The entire system is divided into two parts. They are:

- Registration
- Authentication

In the registration process, the $\mathcal{M}$ requests for registration to $\mathcal{SP}$ and $\mathcal{SP}$ approves the request and completes the registration process of $\mathcal{M}$. Similarly, $\mathcal{D}$ and $\mathcal{R}$ complete the registration process. Each of $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ follow Protocol Φ during the registration process and also receive their public key and private key from $\mathcal{SP}$. All interactions are handled by smart contract and transactions are recorded in the chain 1. Any participant within the network can have the public key of other participants. The registration process and Protocol Φ are elaborately discussed in Section 5. In the authentication process, $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ authenticate each other and their IoT devices by following the protocol Γ. Consider a scenario, where $\mathcal{M}$ and $\mathcal{D}$ want to participate in the authentication. Based on asymmetric encryption ECC they authenticate each other. Their smart contract handles all the interactions and transactions are recorded in the chain 2. All participants in the authentication portion will also authenticate the public key from chain 1 by their smart contract. The detailed authentication process and Protocol Γ are discussed in Section 5. Lastly, $\mathcal{C}$ can buy its product from the $\mathcal{R}$ by scanning the QR code of the product by $\mathcal{C}$'s smartphone but $\mathcal{C}$ will not participate in any of the above-mentioned protocols.

4.2. Threat Model

Participants in protocol Φ and Γ do not trust each other except $\mathcal{SP}$. Others are semi-honest adversaries ($\mathcal{A}$), who are honest in following the protocol but also interested in the private data of other participants [35]. $\mathcal{A}$ can also be a middle man adversary. It can dominate the public channel by intercepting, modifying, and forging messages. Yet $\mathcal{A}$ failed to infer information from the private channel. In case of forwarding secrecy, $\mathcal{A}$'s attack has a minute probability of success against participants.

- $\mathcal{A}$ might get all messages between two entities by initiating a passive attack.
- $\mathcal{A}$ might execute any operation by initiating an active attack.
- $\mathcal{A}$ might forge any message in a key agreement stage.
- $\mathcal{A}$ might retrieve the session key of the entity.

4.3. Security Goals

The privacy-preserving protocol Φ and Γ satisfy the following security requirements of the supply chain. $\mathcal{SP}$ is the only trusted entity in the entire system.

- None of the participants can infer other participants' privacy.
- None of the participants can breach other participants' security.
- $\mathcal{A}$ cannot forge any message in a key agreement stage.
- $\mathcal{A}$ cannot retrieve the session key of the entity.
- $\mathcal{A}$ cannot be successful with an impersonate attack.
- $\mathcal{A}$ cannot be successful in forwarding secrecy.
- $\mathcal{A}$ cannot be successful in a replay attack.

Moreover, $\mathcal{A}$ cannot be successful after a passive or an active attack.

5. Model Construction

The section describes the entire system in detail. This scheme mainly consists of two parts, i.e., registration and authentication.

5.1. System Setup

This section only focuses on the system setup. Here, $\mathcal{SP}$ selects an elliptic curve $\mathcal{E}(\mathcal{F}_p)$, where $\mathcal{F}_p$ is a finite field, which is decided by prime p. It also selects a generator $\mathcal{P}$ on the curve with order q and a master or secret key $\mathcal{SK}_{\mathcal{SP}}$. It publishes the public key $\mathcal{PK}_{\mathcal{SP}} = (\mathcal{SK}_{\mathcal{SP}}\mathcal{P})$, $\mathcal{P}$, p, q, $h_i(.)(i = 1,2,3)$ where $h_i : \{0,1\}^* \rightarrow \mathcal{Z}_q^*, i = 1,2$ and $h_3 : \{0,1\}^* \rightarrow \{0,1\}^n$. Here, $\mathcal{Z}_q^*$ is a multiplicative group of integers modulo q.

5.2. Registration

This section describes the registration process and protocol Φ in detail, which illustrates the registration process of $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ with $\mathcal{SP}$. All these participants follow protocol Φ at the time of interaction. The registration process of $\mathcal{M}$ with $\mathcal{SP}$ is described below and $\mathcal{D}$ and $\mathcal{R}$'s registration follow the same protocol. $\mathcal{M}$ submits its identity $\mathcal{ID}_{\mathcal{M}}$ to the $\mathcal{SP}$. The $\mathcal{SP}$ generates a nonce $r_{\mathcal{M}} \in \mathcal{Z}_q^*$, and works out $\mathcal{PK}_{\mathcal{M}} = r_{\mathcal{M}}\,\mathcal{P}$, $\mathcal{X}_{\mathcal{M}} = h_1\,(\mathcal{ID}_{\mathcal{M}} \parallel \mathcal{PK}_{\mathcal{M}})$, and $\mathcal{SK}_{\mathcal{M}} = r_{\mathcal{M}} + \mathcal{SK}_{\mathcal{SP}}\,\mathcal{X}_{\mathcal{M}}$. Then, the $\mathcal{SP}$ sends $\{\mathcal{PK}_{\mathcal{M}}, \mathcal{SK}_{\mathcal{M}}\}$ to $\mathcal{M}$ secretly. Figure 3 shows the entire registration process of $\mathcal{M}$.

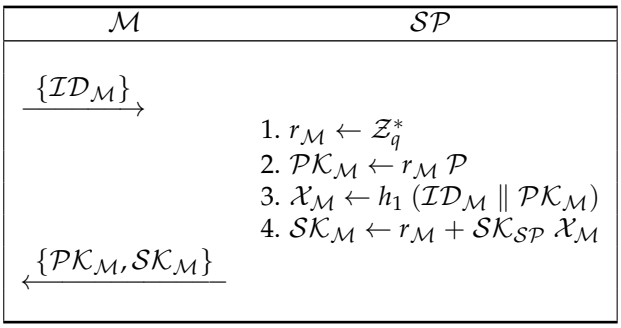


Figure 3. Registration process of $\mathcal{M}$ through protocol Φ.

5.2.1. Blockchain-Based Data Sharing (via Chain 1)

During the registration stage through protocol Φ, the $\mathcal{SP}$ generates the hash of the $\mathcal{PK}$ of $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ and encrypt them with the $\mathcal{SK}_{\mathcal{SP}}$ in order to generate a digital signature ($\mathcal{DS}$). Now, the $\mathcal{SP}$ concats $\mathcal{PK}$s' of $\mathcal{M}$, $\mathcal{D}$ and $\mathcal{R}$, $\mathcal{DS}$ and its sign $\mathcal{SN}_{\mathcal{SP}}$ which are publicly available. The $\mathcal{SP}$ commits the concated information in the blockchain by

calling the smart contract. Algorithm 1 shows the working process of smart contract for registration, where functions $gen()$ and $reg()$ stand for generation of keys and register for writing data into the chain 1. The procedure is described in detail below:

$\mathcal{SP}$ utilize Equation (1) for generating the $\mathcal{DS}_{\mathcal{M}}$ and then ($\mathcal{PK}_{\mathcal{M}} \parallel \mathcal{DS}_{\mathcal{M}} \parallel \mathcal{SN}_{\mathcal{SP}}$).

$$\mathcal{DS}_{\mathcal{M}} = \mathcal{EN}(h(\mathcal{PK}_{\mathcal{M}}), \mathcal{SK}_{\mathcal{SP}}) \tag{1}$$

Similarly, $\mathcal{SP}$ generates $\mathcal{DS}_{\mathcal{D}}$, $\mathcal{DS}_{\mathcal{R}}$ and then ($\mathcal{PK}_{\mathcal{D}} \parallel \mathcal{DS}_{\mathcal{D}} \parallel \mathcal{SN}_{\mathcal{SP}}$), ($\mathcal{PK}_{\mathcal{R}} \parallel \mathcal{DS}_{\mathcal{R}} \parallel \mathcal{SN}_{\mathcal{SP}}$), respectively. Publicly available information from chain 1 are as follows:

- Public key of the entities;
- Verifiable digital signatures of the entities;
- Sign of the service provider.

Algorithm 1: Working process of smart contract for registration.

1 $\mathcal{M}$**'s Input:** request req, $\mathcal{ID}_{\mathcal{M}}$.
2 $\mathcal{SP}$**'s Input:** $r_{\mathcal{M}}$, $\mathcal{P}$, $\mathcal{SK}_{\mathcal{SP}}$ $\mathcal{SN}_{\mathcal{SP}}$.
3 $\mathcal{M}$**'s Output:** $\mathcal{PK}_{\mathcal{M}}$, $\mathcal{SK}_{\mathcal{M}}$.
4 **if** $req == 1$ **then**
5 $\quad \{\mathcal{DS}, \mathcal{PK}, \mathcal{SK}\}_M \leftarrow \text{gen}(r_{\mathcal{M}}, \mathcal{P}, \mathcal{SK}_{\mathcal{SP}}, \mathcal{ID}_{\mathcal{M}})$;
6 $\quad \text{reg}(\mathcal{DS}_{\mathcal{M}}, \mathcal{PK}_{\mathcal{M}}, \mathcal{SN}_{\mathcal{SP}})$;
7 **end**

5.2.2. Security Analysis of Protocol Φ

Proposition 1. *(Security of Protocol Φ). Protocol Φ in Figure 3 is secured in case of adversaries $\mathcal{A}$.*

Proof of Proposition 1. In Protocol Φ: $\mathcal{M}, \mathcal{D}, \mathcal{R}$, and $\mathcal{SP}$, four entities are involved in three scenario. The actions and processes of all of them are the same. Therefore, one scenario is secured means all of them are secured. This section considers the scenario of Figure 3. The function is $\mathcal{F}$:

$$\mathcal{F} : \mathcal{F}(\mathcal{ID}_{\mathcal{M}}, r_{\mathcal{M}}, \mathcal{P}, h_1(), \mathcal{SK}_{\mathcal{SP}}) = (\mathcal{PK}_{\mathcal{M}}, \mathcal{SK}_{\mathcal{M}})$$

The view of each $\mathcal{M}$ is

$$view_{\mathcal{M}}^{\Phi} = (\mathcal{ID}_{\mathcal{M}}, \mathcal{PK}_{\mathcal{M}}, \mathcal{SK}_{\mathcal{M}}, \mathcal{P}, p, q, h_i(.)(i = 1, 2, 3))$$

Clearly, none of this information can be used to infer any private data of other participants. Therefore, in case $\mathcal{A}$ is a semi-honest adversary, he would not able to infer any private information of other participants from these data. Again, if $\mathcal{A}$ is an outsider dishonest adversaries, he might try to take control over the network and try to infer data but that's not possible as the interactions are happening under the Blockchain network. On the other hand, $\mathcal{SP}$ is a trusted entity. Lastly, it is important to discuss the security and privacy issues related to the public ledger of chain 1. Therefore, public view, which also can be seen by $\mathcal{A}$:

$$view_{\mathcal{A}}^{\Phi} = (\mathcal{PK}_{\mathcal{M}}, \mathcal{PK}_{\mathcal{SP}}, \mathcal{DS}_{\mathcal{M}}, \mathcal{SN}_{\mathcal{SP}})$$

Now, $\mathcal{PK}_{\mathcal{M}}, \mathcal{PK}_{\mathcal{SP}}$, $\mathcal{DS}_{\mathcal{M}}$ and $\mathcal{SN}_{\mathcal{SP}}$ has no security concerns as they are just addresses. Thus, protocol Φ is secured in presence of semi-honest and dishonest adversaries for Figure 3. □

5.3. Authentication

This section describes the authentication process and protocol Γ in detail, which illustrates the authentication process of $\mathcal{M}$ with $\mathcal{D}$, and $\mathcal{D}$ with $\mathcal{R}$. All these participants

follow protocol Γ at the time of interaction. The authentication process of $\mathcal{M}$ with $\mathcal{D}$ is illustrated in this section and others follow the same protocol.

5.3.1. Verification of $\mathcal{PK}$ and Corresponding $\mathcal{SP}$

This section describes the verification of participants' ($\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$) $\mathcal{PK}$, where any participant can identify the corresponding $\mathcal{SP}$ for any $\mathcal{PK}$. Let us consider a scenario where a $\mathcal{D}$ attempts to verify the $\mathcal{PK}$ of an $\mathcal{M}$ and identify its corresponding $\mathcal{SP}$. Figure 4 illustrates the entire process. $\mathcal{D}$ retrieves $\mathcal{M}$'s $\mathcal{PK}_{\mathcal{M}}$ along with $\mathcal{DS}_{\mathcal{M}}$ and $\mathcal{SN}_{\mathcal{SP}}$ from chain 1. It recognizes $\mathcal{PK}_{\mathcal{SP}}$ from $\mathcal{SN}_{\mathcal{SP}}$. It decrypts $\mathcal{DS}_{\mathcal{M}}$ with $\mathcal{PK}_{\mathcal{SP}}$ and gets $h(\mathcal{PK}_{\mathcal{M}})_{\mathcal{SP}}$, which is generated by $\mathcal{SP}$. It generates $h(\mathcal{PK}_{\mathcal{M}})^{\mathcal{D}}$ as $\mathcal{H}$. It compares $\mathcal{H}$ and $h(\mathcal{PK}_{\mathcal{M}})_{\mathcal{SP}}$, if matches then $\mathcal{PK}_{\mathcal{M}}$ is verified with $\mathcal{SP}$. All participants use this process to verify the $\mathcal{PK}$ of other participants in the same process and follow the protocol Γ.

$\mathcal{D}$
1. $\mathcal{PK}_{\mathcal{SP}} \leftarrow$ *recognizes* $\mathcal{SN}_{\mathcal{SP}}$ 2. $h(\mathcal{PK}_{\mathcal{M}})_{\mathcal{SP}} \leftarrow (\mathcal{DE}(\mathcal{DS}_{\mathcal{M}}), \mathcal{PK}_{\mathcal{SP}})$ 3. $\mathcal{H} \leftarrow h(\mathcal{PK}_{\mathcal{M}})^{\mathcal{D}}$ 4. $\mathcal{H}\ ? = h(\mathcal{PK}_{\mathcal{M}})^{\mathcal{SP}}$

Figure 4. Verification of $\mathcal{PK}$ and corresponding $\mathcal{SP}$ through protocol Γ.

5.3.2. Authentication between $\mathcal{M}$ and $\mathcal{D}$

This section is described in three phases and shown in Figure 5. $\mathcal{D}$ sends its IoT device ID to $\mathcal{M}$ using asymmetric encryption.

1. **Phase 1:** $\mathcal{M}$ chooses a nonce $u \in \mathcal{Z}_q^*$, $\mathcal{B}_1 = u\mathcal{P}$, $\mathcal{B}_2 = h_2(u\mathcal{PK}_{\mathcal{D}} + uh_1(\mathcal{ID}_{iot_{\mathcal{D}}} \parallel \mathcal{PK}_{\mathcal{D}})\mathcal{PK}_{\mathcal{SP}}) \oplus \mathcal{ID}_{iot_{\mathcal{M}}}$, $\mathcal{B}_3 = h_2(\mathcal{B}_1 \parallel \mathcal{PK}_{\mathcal{M}} \parallel \mathcal{PK}_{\mathcal{D}} \parallel \mathcal{ID}_{iot_{\mathcal{M}}} \parallel \mathcal{ID}_{iot_{\mathcal{D}}})$. Then the message $\mathcal{MA}_1 = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{PK}_{\mathcal{M}}\}$ is sent to $\mathcal{D}$.
2. **Phase 2:** $\mathcal{D}$ calculates $\mathcal{ID}_{iot_{\mathcal{M}}} = \mathcal{B}_2 \oplus h_2(\mathcal{SK}_{\mathcal{D}}\mathcal{B}_1)$ and checks $\mathcal{B}_3? = h_2(\mathcal{B}_1 \parallel \mathcal{PK}_{\mathcal{M}} \parallel \mathcal{PK}_{\mathcal{D}} \parallel \mathcal{ID}_{iot_{\mathcal{M}}} \parallel \mathcal{ID}_{iot_{\mathcal{D}}})$. If true, $\mathcal{D}$ continues to select $v \in \mathcal{Z}_q^*$ and calculates $\mathcal{B}_4 = v\mathcal{P}$, $\mathcal{SEK}_{\mathcal{D}} = h_3(\mathcal{B}_1 \parallel \mathcal{B}_4 \parallel v\mathcal{B}_1)$, $\mathcal{B}_5 = h_2(v\mathcal{PK}_{\mathcal{M}} + vh_1(\mathcal{ID}_{iot_{\mathcal{M}}} \parallel \mathcal{PK}_{\mathcal{M}})\mathcal{PK}_{\mathcal{SP}}) \oplus \mathcal{ID}_{iot_{\mathcal{D}}}$, and $\mathcal{B}_6 = h_2(\mathcal{ID}_{iot_{\mathcal{M}}} \parallel \mathcal{ID}_{iot_{\mathcal{D}}} \parallel \mathcal{B}_1 \parallel \mathcal{B}_4 \parallel \mathcal{SEK}_{\mathcal{D}})$. Then the message $\mathcal{MA}_2 = \{\mathcal{B}_4, \mathcal{B}_5, \mathcal{B}_6\}$ is sent to $\mathcal{M}$.
3. **Phase 3:** $\mathcal{M}$ calculates $\mathcal{SEK}_{\mathcal{M}} = h_3(\mathcal{B}_1 \parallel \mathcal{B}_4 \parallel u\mathcal{B}_4)$, $\mathcal{ID}_{iot_{\mathcal{D}}} = h_2(\mathcal{SK}_{\mathcal{M}}\mathcal{B}_4) \oplus \mathcal{B}_5$, and checks $\mathcal{ID}_{iot_{\mathcal{D}}}$ and $\mathcal{B}_6? = h_2(\mathcal{ID}_{iot_{\mathcal{M}}} \parallel \mathcal{ID}_{iot_{\mathcal{D}}} \parallel \mathcal{B}_1 \parallel \mathcal{B}_4 \parallel \mathcal{SEK}_{\mathcal{M}})$. If true, then the two IoT devices of $\mathcal{M}$ and $\mathcal{D}$ are the authenticated on the both side.

5.3.3. Blockchain Based Data Sharing (via Chain 2)

During the authentication stage through protocol Γ, all participants verify the authenticity of other participants' $\mathcal{PK}$. In the case of Figure 5, $\mathcal{M}$ generates the hash of the $\mathcal{MA}_1$ and commits it in the blockchain by calling the smart contract along with its $\mathcal{PK}_{\mathcal{M}}$. On the other hand, $\mathcal{D}$ generates the hash of the $\mathcal{MA}_2$, $[[\mathcal{ID}_{iot_{\mathcal{D}}}]]_{\mathcal{PK}_{\mathcal{M}}}$ and commits it in the blockchain by calling the smart contract along with its $\mathcal{PK}_{\mathcal{D}}$. Algorithm 2 shows the working process of smart contract for authentication, where functions $auth()$ and $reg()$ stand for authentication and register for writing data into the chain 2. The procedure is described in detail below:

- $\mathcal{M}$ generates $(\mathcal{PK}_{\mathcal{M}} \parallel \mathcal{H}_{\mathcal{M}})$ using (2)

$$\mathcal{H}_{\mathcal{M}} = h(\mathcal{MA}_1) \tag{2}$$

- $\mathcal{D}$ generates $(\mathcal{PK}_{\mathcal{D}} \parallel \mathcal{H}_{\mathcal{D}})$ using (3)

$$\mathcal{H}_{\mathcal{D}} = h(\mathcal{MA}_2 \parallel [[\mathcal{ID}_{iot_{\mathcal{D}}}]]_{\mathcal{PK}_{\mathcal{M}}}) \tag{3}$$

$\mathcal{M}$	$\mathcal{D}$
	1. $[[\mathcal{ID}_{iot_\mathcal{D}}]]_{\mathcal{PK}_\mathcal{M}} \leftarrow \mathcal{EN}(\mathcal{ID}_{iot_\mathcal{D}}, \mathcal{PK}_\mathcal{M})$
$\xleftarrow{\{[[\mathcal{ID}_{iot_\mathcal{D}}]]_{\mathcal{PK}_\mathcal{M}}\}}$	
2. $\mathcal{ID}_{iot_\mathcal{D}} \leftarrow \mathcal{DE}([[\mathcal{ID}_{iot_\mathcal{D}}]]_{\mathcal{PK}_\mathcal{M}}, \mathcal{SK}_\mathcal{M})$ 3. $u \leftarrow \mathcal{Z}_q^*$ 4. $\mathcal{B}_1 \leftarrow u\mathcal{P}$ 5. $\mathcal{H}_2 \leftarrow h_2(u\mathcal{PK}_\mathcal{D} + uh_1(\mathcal{ID}_{iot_\mathcal{D}} \parallel \mathcal{PK}_\mathcal{D})\mathcal{PK}_{\mathcal{SP}})$ 6. $\mathcal{B}_2 \leftarrow \mathcal{H}_2 \oplus \mathcal{ID}_{iot_\mathcal{M}}$ 7. $\mathcal{B}_3 \leftarrow h_2(\mathcal{B}_1 \parallel \mathcal{PK}_\mathcal{M} \parallel \mathcal{PK}_\mathcal{D} \parallel \mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{ID}_{iot_\mathcal{D}})$ 8. $\mathcal{MA}_1 \leftarrow \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{PK}_\mathcal{M}\}$	
$\xrightarrow{\{\mathcal{MA}_1\}}$	
	9. $\mathcal{ID}_{iot_\mathcal{M}} \leftarrow \mathcal{B}_2 \oplus h_2(\mathcal{SK}_\mathcal{D}\mathcal{B}_1)$ 10. $\mathcal{H}_2^{'} \leftarrow h_2(\mathcal{B}_1 \parallel \mathcal{PK}_\mathcal{M} \parallel \mathcal{PK}_\mathcal{D} \parallel \mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{ID}_{iot_\mathcal{D}})$ 11. check $\mathcal{B}_3? = \mathcal{H}_2^{'}$ 12. $v \leftarrow \mathcal{Z}_q^*$ 13. $\mathcal{B}_4 \leftarrow v\mathcal{P}$ 14. $\mathcal{SEK}_\mathcal{D} \leftarrow h_3(\mathcal{B}_1 \parallel \mathcal{B}_4 \parallel v\mathcal{B}_1)$ 15. $\mathcal{H}_2^{''} \leftarrow h_2(v\mathcal{PK}_\mathcal{M} + vh_1(\mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{PK}_\mathcal{M})\mathcal{PK}_{\mathcal{SP}})$ 16. $\mathcal{B}_5 \leftarrow \mathcal{H}_2^{''} \oplus \mathcal{ID}_{iot_\mathcal{D}}$ 17. $\mathcal{B}_6 \leftarrow h_2(\mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{ID}_{iot_\mathcal{D}} \parallel \mathcal{B}_1 \parallel \mathcal{B}_4 \parallel \mathcal{SEK}_\mathcal{D})$ 18. $\mathcal{MA}_2 \leftarrow \{\mathcal{B}_4, \mathcal{B}_5, \mathcal{B}_6\}$
$\xleftarrow{\{\mathcal{MA}_2\}}$	
19. $\mathcal{SEK}_\mathcal{M} \leftarrow h_3(\mathcal{B}_1 \parallel \mathcal{B}_4 \parallel u\mathcal{B}_4)$ 20. $\mathcal{ID}_{iot_\mathcal{D}} \leftarrow h_2(\mathcal{SK}_\mathcal{M}\mathcal{B}_4) \oplus \mathcal{B}_5$ 21. $\mathcal{H}_2^{'''} \leftarrow h_2(\mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{ID}_{iot_\mathcal{D}} \parallel \mathcal{B}_1 \parallel \mathcal{B}_4 \parallel \mathcal{SEK}_\mathcal{M})$ 22. check $\mathcal{ID}_{iot_\mathcal{D}}$ and $\mathcal{B}_6? = \mathcal{H}_2^{'''}$	

Figure 5. Authentication process of $\mathcal{M}$ and $\mathcal{D}$ through protocol Γ.

Again, in the case of the registration process of $\mathcal{D}$ generates the hash of the $\mathcal{MA}_1$ and commits it in the blockchain by calling the smart contract along with its $\mathcal{PK}_\mathcal{D}$. On the other hand, $\mathcal{R}$ generates the hash of the $\mathcal{MA}_2$, $[[\mathcal{ID}_{iot_\mathcal{R}}]]_{\mathcal{PK}_\mathcal{D}}$ and commits it in the blockchain by calling the smart contract along with its $\mathcal{PK}_\mathcal{R}$. The procedure is described in detail below:

- $\mathcal{D}$ generates $(\mathcal{PK}_\mathcal{D} \parallel \mathcal{H}_\mathcal{D})$ using (4)

$$\mathcal{H}_\mathcal{D} = h(\mathcal{MA}_1) \tag{4}$$

- $\mathcal{R}$ generates $(\mathcal{PK}_\mathcal{R} \parallel \mathcal{H}_\mathcal{R})$ using (5)

$$\mathcal{H}_\mathcal{R} = h(\mathcal{MA}_2 \parallel [[\mathcal{ID}_{iot_\mathcal{R}}]]_{\mathcal{PK}_\mathcal{D}}) \tag{5}$$

Publicly available information from chain 2 are as follows:

- Public key of the entities
- Hash of the shared messages

Algorithm 2: Working process of smart contract for authentication.

$\mathcal{M}$**'s Input:** PK_M, SK_M, u, $\mathcal{P}$, $\mathcal{ID}_{iot_\mathcal{M}}$.
$\mathcal{D}$**'s Input:** $\mathcal{ID}_{iot_\mathcal{D}}$, PK_D, SK_D, v, $\mathcal{P}$.
if PK_M *is in chain 1* **then**
  if $auth(u, v, \mathcal{P}, \{\mathcal{SK}, \mathcal{SM}, \mathcal{ID}_{iot}\}_{\mathcal{M}|\mathcal{D}}) == 1$ **then**
    $\text{reg}(\mathcal{PK}_\mathcal{M}, \mathcal{H}_\mathcal{M}, \mathcal{PK}_\mathcal{D}, \mathcal{H}_\mathcal{D})$;
  end
end

5.3.4. Security Analysis of Protocol Γ

Proposition 2. *(Security of Protocol* Γ*). Protocol* Γ *in Figure 5 is secured in case of adversaries* $\mathcal{A}$*.*

Proof of Proposition 2. In Protocol Γ: mainly $\mathcal{M}, \mathcal{D}$, and $\mathcal{R}$, three entities are involved in two scenario. The actions and processes for both of them are the same. Therefore, one scenario is secured means another one is also secured. This section considers the scenario of Figure 5. The function is $\mathcal{F}$:

$$\mathcal{F} : \mathcal{F}(\ \{\mathcal{ID}_{iot}, \mathcal{PK}, \mathcal{SK}\}_{\{\mathcal{M},\mathcal{D}\}}, \mathcal{EN}(), \mathcal{DE}(), u, v, \mathcal{P},$$

$$\mathcal{PK}_\mathcal{SP}, h_i()(i = 1, 2, 3))\ = (\mathcal{MA}_1, \mathcal{MA}_2)$$

The view of each $\mathcal{M}$ is:

$$view^\Gamma_\mathcal{M} = (\mathcal{ID}_{iot_\mathcal{D}}, u, \mathcal{P}, \mathcal{PK}_\mathcal{SP}, \mathcal{PK}_\mathcal{D}, \mathcal{SEK}_\mathcal{M})$$

Here, $\mathcal{M}$ can authenticate $\mathcal{D}$ by checking $\mathcal{B}_6$ and there are no other available data visible to $\mathcal{M}$ from where it can infer further private information. Again, the view of each $\mathcal{D}$ is:

$$view^\Gamma_\mathcal{D} = (\mathcal{ID}_{iot_\mathcal{M}}, v, \mathcal{P}, \mathcal{PK}_\mathcal{SP}, \mathcal{PK}_\mathcal{M}, \mathcal{SEK}_\mathcal{D})$$

$\mathcal{D}$ can authenticate $\mathcal{M}$ by checking $\mathcal{B}_3$ and there are no other available data visible to $\mathcal{D}$ from where it can infer further private information. On the other hand, it is important to discuss the view outsider dishonest adversaries $\mathcal{A}$. In ideal case its view is:

$$view^\Gamma_\mathcal{A} = (\mathcal{PK}_\mathcal{M}, \mathcal{PK}_\mathcal{SP}, \mathcal{PK}_\mathcal{D}, \mathcal{H}_\mathcal{M}, \mathcal{H}_\mathcal{D})$$

In the ideal case $\mathcal{A}$ can not infer any information from $\mathcal{PK}_\mathcal{M}, \mathcal{PK}_\mathcal{SP}, \mathcal{PK}_\mathcal{D}, \mathcal{H}_\mathcal{M}$ and $\mathcal{H}_\mathcal{D}$ as $\mathcal{PK}$s' are addresses and hash values has no backward operations. Considering the threat from the threat model, $\mathcal{A}$ has far more ability and visibility than the publicly available data. It is also important to analyze the security of those threats. It is clear that the $\mathcal{ID}_{iot_{\{\mathcal{M},\mathcal{D}\}}}$ are secured by the hash values $h_2(u\mathcal{PK}_\mathcal{M} + uh_1(\mathcal{ID}_{iot_\mathcal{M}} \parallel \mathcal{PK}_\mathcal{M})\mathcal{PK}_\mathcal{SP})$ and $h_2(v\mathcal{PK}_\mathcal{D} + vh_1(\mathcal{ID}_{iot_\mathcal{D}} \parallel \mathcal{PK}_\mathcal{D})\mathcal{PK}_\mathcal{SP})$, respectively. The outcomes needs $\mathcal{SK}_\mathcal{SP}$ or $\mathcal{SK}_\mathcal{M}$ and $\mathcal{SK}_\mathcal{SP}$ or $\mathcal{SK}_\mathcal{M}$ to directly or indirectly forge those hash values. These keys' are private to their respective owners. Again, in the case of Forward Secrecy $\mathcal{A}$ breaks and obtains all of the secret keys from $\mathcal{M}$ and $\mathcal{D}$ such as $\mathcal{SK}_\mathcal{M}$ and $\mathcal{SK}_\mathcal{D}$. However, $\mathcal{A}$ failed to infer past session keys as all of them are generated based on the ECDH issue. Since $u, v, \mathcal{P}$ are not precisely calculable, the forward secrecy is preserved. Again for impersonation attack, if $\mathcal{A}$ intends to infer any message at the time of key agreement, it requires $\mathcal{SK}_\mathcal{SP}, \mathcal{SK}_\mathcal{M}$ or $\mathcal{SK}_\mathcal{D}$. Yet according to the premise of $\mathcal{A}$, it cannot get any of them. Therefore, it will fail to build the entire message. Therefore, this invasion will fail. Lastly, in case of a reply attack, all individuals utilize unexplored random numerals v and u every time. $\mathcal{A}$ will not be able to crack the ECDH issue depending on ($u\ \mathcal{P}$, $v^{old}\ \mathcal{P}$) or ($u^{old}\ \mathcal{P}$, $v\ \mathcal{P}$), despite any message is being replayed. Thus, protocol Γ is secured in presence of semi-honest and dishonest adversaries for Figure 5. □

6. Experimental Analysis

This section describes the test apparatuses and analyzes the performance evaluation of the suggested schema.

6.1. Testbed

Hyperledger Fabric network deployment machine configuration: Memory 66 GB. Processor: Xeon(R) Intel(R), 3.70GHz W-2135 CPU (6 Core). GPU: Attached GPUs: 4. Product Name: NVIDIA GeForce RTX 2080 Ti. Blockchain transaction performance test machine: Memory, 16 GB 2667 MHz DDR4. Processor: 8-Core 2.3 GHz Core i9 Intel. GPU: Graphics 630 1536 MB Intel UHD.

6.2. Score and Scalability Evaluation Metric

This subsection depicts the measures used to analyze the outcomes.

Evaluation Metrics

The outcomes of the suggested framework is evaluated based on execution time ($\mathcal{ET}$), average latency ($\mathcal{AL}$), and average throughput ($\mathcal{AT}$).

- $\mathcal{ET}$: The total amount of time (in seconds) consumed by a system to perform all transactions for a certain corpus, which is showed in Equation (6) shows the where $\mathcal{N}$ is the total number of transactions.

$$\mathcal{ET} = \sum_{i=1}^{\mathcal{N}} (\mathcal{T}_2 - \mathcal{T}_1) \tag{6}$$

  $\mathcal{T}_1$ and $\mathcal{T}_2$ represent the time when the transaction was made and the blockchain verified the transaction, respectively.
- $\mathcal{AL}$: The average latency is the norm of the difference between $\mathcal{T}_2$ and $\mathcal{T}_1$ in a dataset for a bunch of transactions, which is shown in Equation (7).

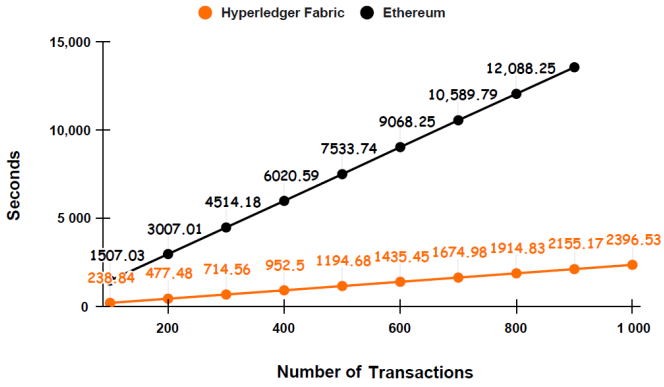
$$\mathcal{AL} = \frac{\sum_{i=1}^{\mathcal{N}} (\mathcal{T}_2 - \mathcal{T}_1)}{\mathcal{N}} \tag{7}$$

- $\mathcal{AT}$: The average throughput is the norm of successful transaction's number per second over the execution time, which is shown in Equation (8).

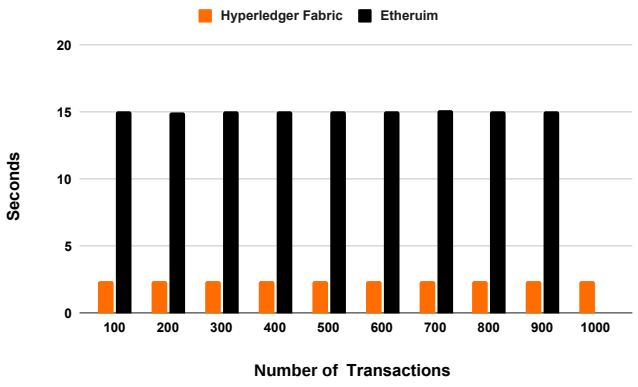
$$\mathcal{AT} = \frac{\frac{\mathcal{N}}{\sum_{i=1}^{\mathcal{N}} (\mathcal{T}_2 - \mathcal{T}_1)}}{\mathcal{N}} \tag{8}$$

6.3. Result Evaluation

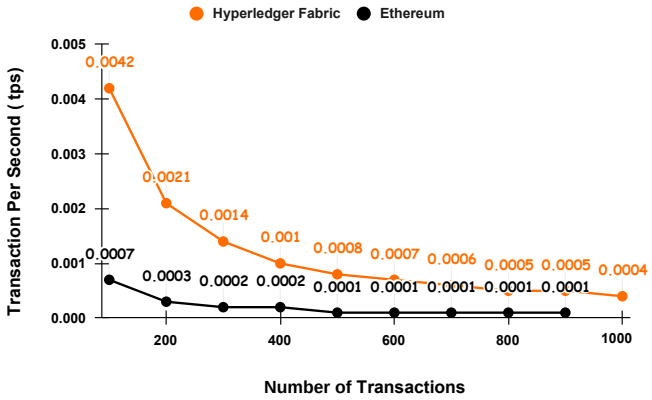
This section demonstrates the result analysis of the system and also detail analysis of protocol Φ and Γ. The proposed system is evaluated in three ways: execution time, average latency, and average throughput. Figure 6 illustrates the performance analysis of Hyperledger Fabric and Ethereum.



(**a**)



(**b**)



(**c**)

Figure 6. Performance analysis of Hyperledger Fabric and Ethereum. (**a**) represents the execution time where the x-axis shows the number of transactions and the y-axis shows the time in seconds for each group of transactions. (**b**) represents the average latency where the x-axis presents the number of transactions, and the y-axis presents the time in seconds for each set of transactions, and (**c**) represents the average throughput where the x-axis holds the number of transactions, and the y-axis holds the number of transaction per second (tps)) for each set of transactions. (**a**) Execution time; (**b**) Average latency; (**c**) Average throughput.

This study examines the diversity in performance time consumption by altering the numeral of transactions in Figure 6a with two types of blockchain technology such as Etherium and Hyperledger Fabric. The x-axis exhibits the transactions counts running from 1 to 1000 and the y-axis presents the total time consumption for various groups of transactions in seconds. The graph is represented in linear scale. The execution time is proportional to the number of transactions. In the scenario, Etherium hardly completes 980 transactions. Analysis shows that the Hyperledger Fabric constantly consumes less time than Ethereum. The difference between Ethereum and Hyperledger Fabric in execution time grows larger as the transactions amount increases. In Figure 6b,c, we assessed the latency and throughput, respectively by deviating the count of transactions with Etherium and Hyperledger Fabric. The x-axis of both figures shows the transactions number, which varies from 1 to 1000. The y-axis of Figure 6b shows the average latency in seconds for every set of transactions but on that same axis, Figure 6c shows the average throughput in transaction per second (tps) for individual transaction sets. Analysis of the performance reveals that latency of Hyperledger Fabric is constantly lower and throughput is constantly higher in comparison to Ethereum. Therefore, it proves Hyperledger Fabric

is faster in comparison to Ethereum. In summary, the proposed system provides more reliable performance in Hyperledger Fabric than Ethereum in terms of scalability. Another important feature of Hyperledger Fabric is that it is a private network but Ethereum is public. Therefore the transaction privacy can also be achieved by Hyperledger Fabric. Figure 7 illustrates the performance analysis of protocol Φ and Γ on Hyperledger Fabric.

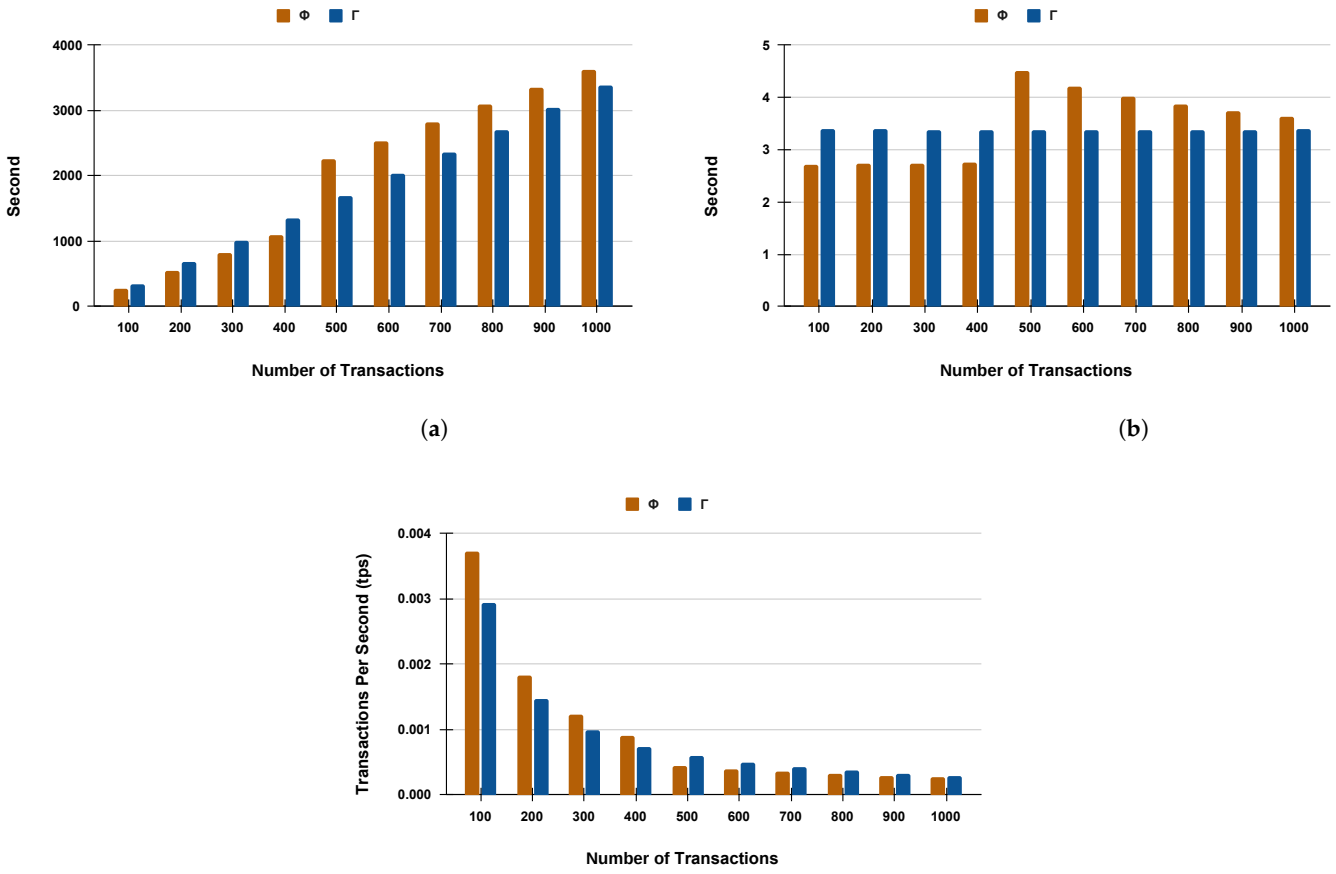


Figure 7. Performance analysis of protocol Φ and Γ on Hyperledger Fabric. (**a**) represents the execution time where x-axis shows the number of transactions and the y-axis shows the time in seconds for each group of transactions. (**b**) represents the average latency where x-axis presents the number of transactions, and the y-axis presents the time in seconds for each set of transactions, and (**c**) represents the average throughput where x-axis holds the number of transactions, and the y-axis holds the number of transaction per second (tps)) for each set of transactions. (**a**) Execution time; (**b**) Average latency; (**c**) Average throughput.

We investigate the execution time of two protocols (i.e., Φ and Γ) by the number of transactions in Figure 7a with Hyperledger Fabric. The x-axis illustrates transactions counts (varying from 1 to 1000) and the y-axis indicates an individual group's transaction time. The execution time is proportional to the transaction count on a linear scale. The result analysis of this study shows that Hyperledger Fabric's execution time is quite practical. When the number of transactions is 100, then the protocol Φ takes 268.55811 s and the protocol Γ takes 339.351912 s. When the number of transactions is 500, then the protocol Φ takes 2258.245325 s and the protocol Γ takes 1687.087315 s. When the number of transactions is 1000, then the protocol Φ takes 3626.718443 s and the protocol Γ takes 3386.043616 s. It is clear from the above Figure 7a that protocol Φ consumes much more time than protocol Γ as the transaction number increases.

Figure 7b assessed the average latency by transaction count with Hyperledger. The x-axis and y-axis show the same parameters as Figure 7a. The result analysis of this study shows that the execution time of Hyperledger Fabric is pretty practical. When the number of transactions is 100, then the protocol Φ consumes 2.71235371 s and the protocol Γ consumes 3.39351912 s. When the number of transactions is 500, then the protocol Φ consumes 4.51649065 s and the protocol Γ consumes 3.37417463 s. When the number of transactions is 1000, then the protocol Φ consumes 3.626718443 s and the protocol Γ consumes 3.386043616 s.

We again assessed the average throughput by altering the transaction counts in Figure 7c with Hyperledger. The x-axis and y-axis show the same parameters as Figure 6c. When the number of transactions is 100, then the protocol Φ executes 0.00372358891 tps and the protocol Γ executes 0.0029467935 tps. When the number of transactions is 500, then the protocol Φ executes 0.000442821685 tps and the protocol Γ executes 0.0005927375 tps. When the number of transactions is 1000, then the protocol Φ executes 0.0002757313576 tps and the protocol Γ executes 0.0002953299 tps.

After the analysis of transaction time, it is important to have a look at execution time. Table 4 shows the execution time analysis. It focuses on each entity's time consumption. Precisely, there is no previous work whose result can be directly comparable with this proposed system. In the proposed system, entities $\mathcal{SP}$, $\mathcal{M}$, $\mathcal{D}$, and $\mathcal{R}$ consume 2.049688 ms, 4.534202 ms, 4.011596 ms, and 4.373648 ms, respectively. ECC's time consumption of the proposed system shows better performance but the total execution time of the proposed system is a bit higher due to the time expenditure of blockchain.

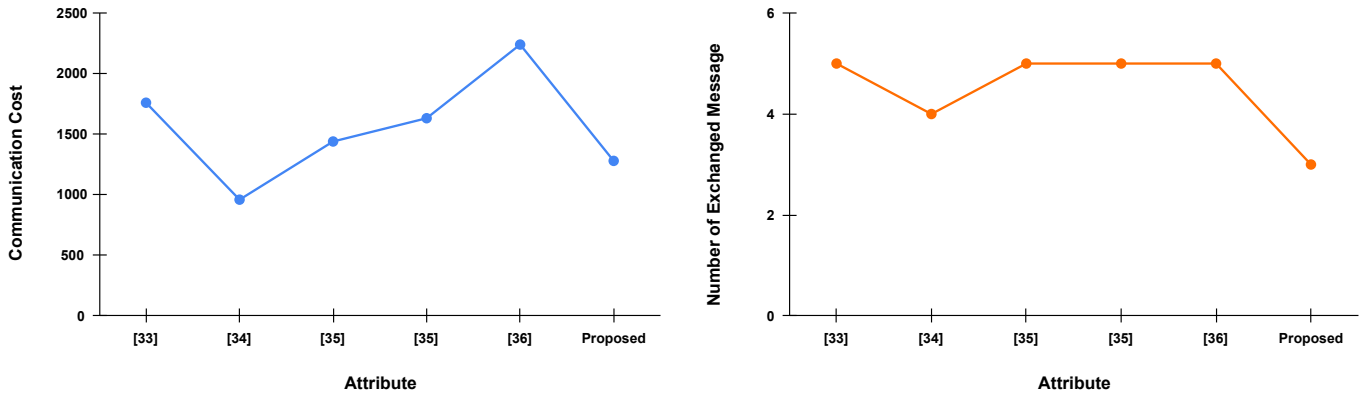
Table 4. Execution time analysis.

Attribute	Entities	Time (millisecond, ms)
Proposed method	$\mathcal{SP}$	2.049688
	$\mathcal{M}$	4.534202
	$\mathcal{D}$	4.011596
	$\mathcal{R}$	4.373648

The performance of the proposed method can be compared with the method of other domains in terms of computational costs and the number of exchanged message. Analysis is showed in Table 5 and Figure 8. The proposed outperforms the methods of other domains also, where it takes 1260 bits for communication costs and only 3 exchanges of messages.

Table 5. Comparison of communication cost (Co-co) and number of exchanged message (Ex-me).

Attribute	Co-co	Ex-me
Sidorov [36]	1760	5
Mujahid [37]	960	4
LRMAPC [38]	1440	5
ULRMAPC [38]	1632	5
LBRAPS [39]	2240	5
Proposed	1280	3



(**a**) (**b**)

Figure 8. Performance analysis of the proposed scheme with the method of other domains in terms of communication cost and number of the exchanged message. (**a**) Comparison for communication cost; (**b**) Comparison for number of exchanged message.

7. Conclusions

Integration of IoT devices in a centralized nature increases the issue of transaction data privacy and security of the supply chain management system. Therefore, this paper proposed a unified solution with the distributed ledger technology, i.e., Hyperledger fabric, IoT, and elliptic curve cryptography, to protect the transaction data from privacy and security breaches. ECC ensured the lightweight cryptographic operations and authentication of IoT devices. Authenticated IoT scanner guarantees an error-free supply chain transaction enabling the trusted immutable ledger among all participants. Rigorous implementation of the proposed system on the Hyperledger fabric network confirmed that the system works smoothly in a multi-party setup. The result and security analysis prove that the proposed system is robust and secure for real-life applications.

In future research, we want to integrate self-sovereign identity (SSI) with the distributed ledger technology for faster and more reliable peer-to-peer authentication processes for all supply chain entities. The decentralized SSI module will guarantee frictionless supply chain transactions where data privacy and security can also be ensured.

Author Contributions: Conceptualization, A.S.M.T.H. and R.U.H.; methodology, A.S.M.T.H., S.S., and R.U.H.; software, S.S. and A.D.; validation, A.S.M.T.H. and Q.J.; formal analysis, A.S.M.T.H. and R.U.H.; investigation, S.S.; resources, S.S.; data curation, S.S.; writing—original draft preparation, S.S. and R.U.H.; writing—review and editing, A.S.M.T.H., R.U.H., A.R., and Q.J.; visualization, S.S.; supervision, R.U.H.; project administration, A.S.M.T.H.; funding acquisition, Q.J. All authors have read and agreed to the published version of the manuscript.

Funding: The National Key Research and Development Program of China: 2021YFF1200104. Additionally, thanks to the Institute of Automation Research and Engineering for their technical and infrastructure support.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. da Silva, F.S.T.; da Costa, C.A.; Crovato, C.D.P.; da Rosa Righi, R. Looking at energy through the lens of Industry 4.0: A systematic literature review of concerns and challenges. *Comput. Ind. Eng.* **2020**, *143*, 106426. [CrossRef]
2. Granell, C.; Kamilaris, A.; Kotsev, A.; Ostermann, F.O.; Trilles, S. Internet of things. In *Manual of Digital Earth*; Springer: Singapore, 2020; pp. 387–423.
3. Akter, S.; Habib, A.; Islam, M.A.; Hossen, M.S.; Fahim, W.A.; Sarkar, P.R.; Ahmed, M. Comprehensive Performance Assessment of Deep Learning Models in Early Prediction and Risk Identification of Chronic Kidney Disease. *IEEE Access* **2021**, *9*, 165184–165206. [CrossRef]
4. Bing, K.; Fu, L.; Zhuo, Y.; Yanlei, L. Design of an Internet of Things-based smart home system. In Proceedings of the 2011 2nd International Conference on Intelligent Control and Information Processing, Harbin, China, 25–28 July 2011; Volume 2, pp. 921–924.
5. Langley, C.J.; Novack, R.A.; Gibson, B.; Coyle, J.J. *Supply Chain Management: A Logistics Perspective*; Cengage Learning: Boston, MA, USA, 2020.
6. De Vass, T.; Shee, H.; Miah, S.J. The effect of "Internet of Things" on supply chain integration and performance: An organisational capability perspective. *Australas. J. Inf. Syst.* **2018**, *22*. [CrossRef]
7. Khujamatov, K.; Reypnazarov, E.; Akhmedov, N.; Khasanov, D. IoT based centralized double stage education. In Proceedings of the 2020 International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, Uzbekistan, 4–6 November 2020; pp. 1–5.
8. Sabah, S.; Hasan, A.S.M.T.; Daria, A. A Blockchain-based Approach to Detect Counterfeit Drugs in Medical Supply Chain. In Proceedings of the International Conference on Big Data, IoT and Machine Learning (BIM 2021), Cox's Bazar, Bangladesh, 23–25 September 2021.
9. Li, X.; Jiang, P.; Chen, T.; Luo, X.; Wen, Q. A survey on the security of blockchain systems. *Future Gener. Comput. Syst.* **2020**, *107*, 841–853. [CrossRef]
10. Shahroz, M.; Mushtaq, M.F.; Ahmad, M.; Ullah, S.; Mehmood, A.; Choi, G.S. IoT-based smart shopping cart using radio frequency identification. *IEEE Access* **2020**, *8*, 68426–68438. [CrossRef]
11. Kawano, Y.; Cao, M. Design of privacy-preserving dynamic controllers. *IEEE Trans. Autom. Control* **2020**, *65*, 3863–3878. [CrossRef]
12. Li, C.; Palanisamy, B.; Xu, R. Scalable and privacy-preserving design of on/off-chain smart contracts. In Proceedings of the 2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW), Macao, Macao, 8–12 April 2019; pp. 7–12.
13. Sweeney, L. K-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **2002**, *10*, 557–570. [CrossRef]
14. Kambourakis, G. Anonymity and closely related terms in the cyberspace: An analysis by example. *J. Inf. Secur. Appl.* **2014**, *19*, 2–17. [CrossRef]
15. Pfitzmann, A.; Köhntopp, M. Anonymity, unobservability, and pseudonymity—A proposal for terminology. In *Designing Privacy Enhancing Technologies*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 1–9.
16. Pfitzmann, A.; Hansen, M. A Terminology for Talking About Privacy by Data Minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management. 2010. Available online: http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf (accessed on 20 October 2021).
17. Hansen, M.; Smith, R.; Tschofenig, H. CA Privacy terminology and concepts. In *Internet Draft, March 2012, Technical Report*; Network Working Group; IETF: Fremont, CA, USA, 2011.
18. Malik, S.; Dedeoglu, V.; Kanhere, S.S.; Jurdak, R. Trustchain: Trust management in blockchain and iot supported supply chains. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 184–193.
19. Tsang, Y.P.; Choy, K.L.; Wu, C.H.; Ho, G.T.S.; Lam, H.Y. Blockchain-driven iot for food traceability with an integrated consensus mechanism. *IEEE Access* **2019**, *7*, 129000–129017. [CrossRef]
20. Shi, J.; Yi, D.; Kuang, J. Pharmaceutical supply chain management system with integration of iot and blockchain technology. In *International Conference on Smart Blockchain*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 97–108.
21. Caro, M.P.; Ali, M.S.; Vecchio, M.; Giaffreda, R. Blockchain-based traceability in agri-food supply chain management: A practical implementation. In Proceedings of the 2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany), Tuscany, Italy, 8–9 May 2018; pp. 1–4.
22. Abdel-Basset, M.; Manogaran, G.; Mohamed, M. Internet of things (iot) and its impact on supply chain: A framework for building smart, secure and efficient systems. *Future Gener. Comput. Syst.* **2018**, *86*, 614–628. [CrossRef]
23. Cui, P.; Dixon, J.; Guin, U.; Dimase, D. A blockchain-based framework for supply chain provenance. *IEEE Access* **2019**, *7*, 157113–157125. [CrossRef]
24. Cocco, L.; Mannaro, K.; Tonelli, R.; Mariani, L.; Lodi, M.B.; Melis, A.; Fanti, A. A Blockchain-Based Traceability System in Agri-Food SME: Case Study of a Traditional Bakery. *IEEE Access* **2021**, *9*, 62899–62915. [CrossRef]
25. Benčić, F.M.; Skočir, P.; Žarko, I.P. DL-Tags: DLT and smart tags for decentralized, privacy-preserving, and verifiable supply chain management. *IEEE Access* **2019**, *7*, 46198–46209. [CrossRef]
26. Bhutta, M.N.M.; Ahmad, M. Secure Identification, Traceability and Real-Time Tracking of Agricultural Food Supply During Transportation Using Internet of Things. *IEEE Access* **2021**, *9*, 65660–65675. [CrossRef]

27. Grida, M.; Mohamed, R.; Zaid, A.H. A novel plithogenic MCDM framework for evaluating the performance of IoT based supply chain. *Neutrosophic Sets Syst.* **2020**, *33*, 323–341.
28. Yadav, S.; Garg, D.; Luthra, S. Development of IoT based data-driven agriculture supply chain performance measurement framework. *J. Enterp. Inf. Manag.* **2020**, *34*, 292–327. [CrossRef]
29. Yadav, S.; Luthra, S.; Garg, D. Internet of things (IoT) based coordination system in Agri-food supply chain: development of an efficient framework using DEMATEL-ISM. *Oper. Manag. Res.* **2020**. [CrossRef]
30. Zhang, H.; Sakurai, K. Blockchain for iot-based digital supply chain: A survey. In Proceedings of the International Conference on Emerging Internetworking, Data & Web Technologies, Kitakyushu, Japan, 24–26 February 2020; Springer: Cham, Switzerland, 2020; pp. 564–573.
31. Simmons, G.J. Symmetric and asymmetric encryption. *ACM Comput. Surv. CSUR* **1979**, *11*, 305–330. [CrossRef]
32. Islam, T.; Youki, R.A.; Chowdhury, B.R.; Hasan, A.S.M. An ECC Based Secure Communication Protocol for Resource Constraints IoT Devices in Smart Home. In Proceedings of the International Conference on Big Data, IoT, and Machine Learning, Cox's Bazar, Bangladesh, 23–25 September 2021; Springer: Singapore, 2022; pp. 431–444.
33. Hao, X.; Yu, L.; Zhiqiang, L.; Zhen, L.; Dawu, G. Dynamic practical byzantine fault tolerance. In Proceedings of the 2018 IEEE Conference on Communications and Network Security (CNS), Beijing, China, 30 May–1 June 2018; pp. 1–8.
34. Wu, F.; Xu, L.; Li, X.; Kumari, S.; Karuppiah, M.; Obaidat, M.S. A lightweight and provably secure key agreement system for a smart grid with elliptic curve cryptography. *IEEE Syst. J.* **2018**, *13*, 2830–2838. [CrossRef]
35. Haque, R.U.; Hasan, A.S.M.; Jiang, Q.; Qu, Q. Privacy-preserving K-nearest neighbors training over blockchain-based encrypted health data. *Electronics* **2020**, *9*, 2096. [CrossRef]
36. Sidorov, M.; Ong, M.T.; Sridharan, R.V.; Nakamura, J.; Ohmura, R.; Khor, J.H. Ultralightweight Mutual Authentication RFID Protocol for Blockchain Enabled Supply Chains. *IEEE Access* **2019**, *7*, 7273–7285. [CrossRef]
37. Mujahid, U.; Islam, M.N.; Sarwar, S. A new ultralightweight RFID authentication protocol for passive low cost tags: KMAP. *Wirel. Pers. Commun.* **2017**, *94*, 725–744. [CrossRef]
38. Fan, K.; Gong, Y.; Liang, C.; Li, H.; Yang, Y. Lightweight and ultralightweight RFID mutual authentication protocol with cache in the reader for IoT in 5G. *Secur. Commun. Netw.* **2016**, *9*, 3095–3104. [CrossRef]
39. Jangirala, S.; Das, A.K.; Vasilakos, A.V. Designing secure lightweight blockchain-enabled RFID-based authentication protocol for supply chains in 5G mobile edge computing environment. *IEEE Trans. Ind. Inform.* **2019**, *16*, 7081–7093. [CrossRef]