IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 15, 2020, accepted August 2, 2020, date of publication August 19, 2020, date of current version September 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3017871

Blockchain Based Sensitive Data Management by Using Key Escrow Encryption System From the Perspective of Supply Chain

SUNGYONG CHA, SEUNGSOO BAEK, AND SEUNGJOO KIM, (Member, IEEE)
Center for Information Security and Technology (CIST), School of Cybersecurity, Korea University, Seoul 02841, South Korea

Corresponding author: Seungjoo Kim (skim71@korea.ac.kr)

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) funded by the Korea Government (MSIT) (Self-Learning Cyber Immune Technology Development) under Grant 2017-0-00184.

ABSTRACT The security of operation and maintenance phase in systems that share long-life cycles like weapon systems is of great importance. Even if the system passes the security evaluation at the development stage before release, it can be adversely affected by the penetration of counterfeit components (parts) during the operation and maintenance phase. Such security issues are concatenated with data related to supply chain, accordingly, system parts need to fulfill the traceability on a fundamental basis. In addition to traceability, supply chains should also meet the data security standards of availability, integrity and confidentiality in the long run. Also, even without trusted third party, these data should be available to users. In this paper, we, therefore, propose a framework that utilizes blockchain and key escrow encryption system in a bid to optimize the security of supply chains for long-lifecycle systems and provide better measures to improve services for global business survivability.

INDEX TERMS Blockchain, supply chain, key escrow system, key recovery, weapon system, SDLC.

I. INTRODUCTION

Information systems contriving confidential business information, arrays of national security information and intelligence, as well as the weapon systems in the military are becoming more advanced and intricate. As systems become more complex, supply chains of system become more complicated and more globalized. Therefore, to construct these systems in a more trustworthy context, unlike the usual security testing (e.g. penetration testing) measure that runs after system development, the adoption of Security by Design process becomes a recent trend, where security of the whole system life cycle is being undertaken from the system [1]. Yet, the application of Security by Design concept does not immune systems from other security threats. A representative potential threat is the cybersecurity threat to supply chains. In particular, these security threats to the supply chain have a more adverse impact on long-life systems like weapons systems as compared to those of shorter life-cycles. Recently, the U.S. government alleged that Huawei's hardware and software are full of deliberate security holes in order to enable Chinese government spying. So, Huawei is being tightened from exporting its semiconductors and products to the US and allies [2]. The reason for the U.S. to take such measure is because supply chain security is critically important, and it is very expensive to be fixed if problems arise [3]. Also, these measures would not be happened if U.S. companies kept their whole system manufacturing domestic or a trust relationship was established between the countries. However, we all know that is extremely hard.

As such, it is important to guarantee there is no backdoor in the parts one purchase, especially in cases where there is an absence of trust between the seller and buyer. However, threats related to supply chain do not occur at any point of time nor at backdoors only. One of the most dangerous factors in supply chain management is the problem of counterfeits, and it is important to be able to track transparently the related data throughout the life cycle [4], [5]. In other words, the seller must prove that the components (parts) provided are not forged, and that they are normally distributed in accordance with designated procedures, of which the buyer must ensure.

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

Other security threat related to supply chain is a persistent security update issue. While operating the system, it is necessary to continuously respond to technical threats such as zero-day attacks. This can usually be solved by the latest security patch released by the manufacturer. However, we cannot guarantee the supplier (manufacturer) will release up-to-date patches in a long run while the system is still in operation. In other words, buyers would still demand a sustainable security patch even though the supplier might be short of service provision for good. In this case, the only solution to this is the buyer creating one's own security patch. However, to create a patch, materials like source code, that is sensitive data that developer usually does not provide to buyer, are required. Also, if any problem (e.g. backdoor was founded in the system) occur in the supply chain, buyers should determine the cause to resolve the problem. During investigation of the problem, data will be collected as the evidence of which should be verified its originality to confirm its integrity. Regardless of the sensitiveness of these data, all must be disclosed to clarify the facts and be verified. However, some manufacturers (suppliers) may also delete data to cover up their faults, so safeguards must be taken to prevent such unauthorized deletion. In order to solve the above-mentioned problem, usually, supply chain-related data are submitted to a trusted third party (TTP), and in case problem occurs, it can be solved through the TTP. Nonetheless, there are few scenarios where the above approach is infeasible, for instance when all documents cannot be entrusted to a TTP or in national-wise transactions where a TTP does not exist. In sight of that, there is a necessity to sort out another way

To solve this situation, a blockchain that has properties of decentralization, transparency, immutability, and traceability can be a good candidate [6]. If records of supply chain, starting from system's phase of design, are stored in blockchain, users (both manufacturers and buyers) would be able to review transaction information and data deputed to TTP is inalterable at discretion. Added to the assurance of data integrity, data can be retrieved even if there is only one node present in blockchain. Yet, the storage size of data in blockchain is restrictive to the size of blocks, along with that, uploading sensitive data to blockchain will not be appropriate because blockchain is transparent to the public. For this reason, sensitive data is encrypted and stored in manufacturers' centralized database without using a blockchain. In this case, the problems raised above cannot be solved because the ownership of the data is entirely under their control. Also, it is vulnerable to hackers' APT attacks and data alteration.

In this paper, we propose a solution that utilizes a blockchain to provide the same role as TTP in a TTP-free environment with related data to solve the security problem in supply chain management in development and O&M phase, especially of long-lifecycle systems. In addition, this solution introduces a cryptographic escrow technique to have the same effect on large storage data.

This paper is structured as follows. Section II illustrates prerequisites of the solution's fundamental concept, and other related works. Then, the security requirements are derived from the scenario set in section III. In section IV, we propose the solution which can solve the stated problems and in section V & VI, there is an analysis of the security requirements. The overall conclusion will be covered in section VI.

II. PRELIMINARIES AND RELATED WORKS

In the following, we introduce the techniques to understand our proposed solution, including blockchain and cryptographic technique. Also, we review work related to blockchain-based supply-chain application.

A. BLOCKCHAIN TECHNOLOGY

A blockchain is a growing list of records, called blocks, which are linked using cryptographic primitives. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a Merkle tree) [7]. A blockchain is not only resistant to modification of the data but has other security properties. There is a lot of blockchain-based application in the real field using those properties, which are following

- **Decentralized:** Characteristics in which data is recorded and stored separately on each node without any dependence on the centralized node
- **Transparency:** Characteristics in which all the nodes constituting the blockchain can verify all data that are stored and updated by the blockchain.
- **Immutability:** Characteristics in which any data stored in blockchain is permanently preserved without any modification unless more than 51% of the nodes agree.
- **Anonymity:** Characteristics in which the node can transfer data to each other if they know only the receiver's address.
- **Traceability:** Characteristics in which a user can check the previous data (block) if a user knows the data at a certain point(block) because all blocks are connected by a hash chain.

Blockchain can be divided into public and private block chain (or consortium blockchain). A public block chain (e.g., Bitcoin, Ethereum) is a network where anyone can join the network; whereas a private block (e.g., Hyper ledger, R3) chain is a network that only authorized participants can take part in. Table 1 below shows the difference between two types of blockchain characteristics [7]. As shown in the Table, the private block chain is not completely decentralized because there is an administrator which governs the authority in the network,

B. DISTRIBUTED STORAGE SYSTEM

A blockchain is an immutable and append-only ledger that stores the network state. Distributed consensus between all the network nodes is required in order to extend the blockchain and store important network data among the network nodes. Therefore, for common data to store in blockchain could be prohibitively expensive. Hence, it would be more efficient to store other less important data in other means that share similar level of security level of the blockchain.

TABLE 1. Public vs. private blockchain.

	Public Blockchain	Private Blockchain
Participants	permissionless	only permissioned person
Verification	everyone	permissioned entity
Consensus algorithm	proof of work, proof of state, etc.	Practical Byzantine fault tolerance, Paxos
Authority	equal	not equal
Applications	Bitcoin, Ethereum	IBM Hyperledger, R3 Codra

Inter Planetary File System (IPFS) [8] is a distributed file system that has evolved from prior P2P systems such as DHT (distributed hash table), Bit Torrent, Git, etc. It was inspired by these technologies to provide an enhanced solution for hypermedia data sharing. It presents a new platform for users to write and deploy applications and to distribute and segregate large data. Since it is P2P, no nodes are privileged and, in this way, it can store data on a large number of computers.

IPFS is the most suitable storage medium for this category of data. IPFS allows for distributed storage of data that is immune to altering and forgery. Data stored on the IPFS network cannot be altered without changing the data identifier. In IPFS, the identifier is a cryptographic hash of the data. This means encrypted and big data can be stored to IPFS while storing this identifier to an underlying distributed ledger. This would result in less exhaustive operations over the distributed ledger. A user who has right hash value of the data is always able to get access to the data even they don't know where data is because this system is not address-based system but contents-based system.

C. CRYPTOGRAPHIC TECHNIQUES

One of the problems to be solved in our proposal is to provide an escrow function in an environment without a trusted third party. We intend to solve this by combining the secret sharing technique with the escrow encryption system.

1) THRESHOLD CRYPTOGRAPHY

Threshold cryptography is a secret sharing technique introduced by Shamir and Adi [9] at first. It is generally denoted as (t, n) where distributes a secret among n participants in such a way that any t or more of them can reconstruct a secret, but any $t-1$ or fewer members gain no information about the secret. However, this secret-sharing scheme is carried out by a trusted authority, called dealer. In other words, the only way to verify whether the correct secret shares were distributed is only possible when the secrets were restored. To deal with this, the validity of a share has to be proven before it is distributed. This first solution, called Verifiable Secret Sharing (VSS), was introduced by Chor *et. al* [10] but this scheme was based on interactive-VSS which is unpractical and in light of that, Feldman and Paul [11] suggested the first non-interactive VSS protocol.

However, these schemes must need the presence of a trusted dealer as a secret distributor. Consequently, Pederson and Torben Pryds [12] built the first Distributed Key Generation (DKG) protocol, which is an encryption process in which multiple parties contribute to the calculation of a shared public and private key set [12]. Unlike most public key encryption models, distributed key generation does not rely on Trusted Third Parties. Instead, the participation of a threshold of honest parties determines whether a key pair can be computed successfully.

2) KEY ESCROW ENCRYPTION SYSTEM

Key Escrow Encryption System is an encryption system with a backup decryption capability that allows authorized persons, such as user and government officials, to decrypt cipher text under certain prescribed conditions [13], [14]. This system can be referred to the terms as key recovery, exceptional access and data recovery. Key escrow systems typically have entities called ''escrow agents'' that can recover certain encrypted communication sessions or saved files. Such a system uses a session key encrypted with a key known to the escrow agent so that the escrow agent can decrypt the encrypted communication or files. However, these key escrow systems are configured on the assumption that the escrow agent is fully trusted, and if one escrow agent is not fully trusted, it is configured as multiple escrow agents. Therefore, a key escrow system consists of two group of users, entities holding a session key (or session key information), and entities restoring it as follow [13].

· **User Component:** The user component is software or hardware that provides data encryption and decryption functions. Commits the secret key to the key escrow component.

· **Key Escrow Component:** Key escrow agent manages the storage and distribution and use of data recovery keys.

· **Data Recovery Component:** The data recovery component consists of algorithms, protocols, and equipment needed to recover plaintext information from cipher text. Recovers cipher text using information from key escrow agent.

Such a system is established under the condition that a third party (Key escrow and data recovery components) is fully trusted, and cannot be used in an environment where trust relationship does not exist, such as a blockchain. Thus, in order to make use of these key encryption escrow concepts in such untrusted environment, we solve the problem by modifying this scheme and applying the concept of threshold cryptography.

D. BLOCKCHAIN APPLICATIONS

There are many different kinds of application fields based on the characteristics of the blockchain described in subsection A. The most typical applications are the fields of cryptocurrency such as Bitcoin [15] and Ethereum [16].

In addition to these financial sectors, blockchain has been applied in overseas payment transactions, smart contracts, proof of ownership, electronic voting systems, and real estate transactions. Supply Chain Management (SCM) is the most frequently used application among them in blockchain [17]. In SCM, the flow of materials and services required in manufacturing a given system are organized, which includes various intermediate storage and trade on a global scale within a given supply chain. Due to its complexity, associated costs of managing the inventory, processes and failure detection are particularly expensive. Several companies described in Table 2 (e.g. Everledger [18], Provenance [20], Walmart [22]) are the representative examples to provide blockchain based solutions to improve the efficiency of supply chain management solutions.

TABLE 2. Application of blockchain in supply chain.

Types	Applications	
Supply Chain Management	Everledger [18]	modum.IO AG [19]
	Provenance [20]	HR Hassan et al. [21]
	Walmart [22]	K toyoda. et al. [23]

Everledger [18] provides a service where characteristics and owner of the diamond is recorded on the blockchain to prevent fraud. They offer a permanent ledger for diamonds so that owners, insurance companies and law enforcers can easily check whether the fraud user claims ownership. Modum.IO AG [19] provide a system for monitoring the Cold Channing system by introducing IoT (Internet of Things) sensor devices when transporting medical products. Wal-Mart adopted IBM's Hyperledger Fabric to keep track of products' freshness and fake meats [22]. K toyoda et al also put blockchain technology in practice to verify products' authenticity in the course of second-hand trade [23]; and Hasan *et al.* [21] utilized the technology to confirm the product ownership and set up a deposit system to guarantee safe delivery service where deposit will be charged at a double of the product monetary value. There are many services and proposals that utilize these blockchain technologies, but systems are either those can trace of parts or track ownership of only single product not additive products. To our best knowledge, there was no case where a block chain technique was applied to a supply chain that supported a complex system combining many components(parts), such as a weapon system. However, a pilot program is underway to utilize blockchain to track important parts of the avionic system in the US Navy [24]. Added to that, when a blockchain is applied to such a supply chain it is mostly composed of a private blockchain due to the efficiency problem. However, the private chain is not suitable if there is no trusted third party because not all nodes have equivalent rights.

Key escrow encryption systems generally provide the requester with a key that can recover the cipher text if a dispute arises or certain conditions are met [13]. Key escrow encryption system was introduced first by the Clipper Chip called Escrow Encryption Standard EES) [25]. Reference [26] proposed a method by which a key trustee can decrypt an encrypted cipher at a certain time without revealing the private key of the key escrow system participant. However, this approach assumes that the key trustee(s) are trustworthy, and that no collusion can be made between trustees. In [13], they investigated and presented key escrow encryption systems that are proposed in practical or commercially available. However, these systems also assume that fully trusted third parties provide key escrow services. Although such escrow service is not subjected to secret keys, Goldfeder *et al.* [27] proposed escrow services for cryptocurrencies transactions using Bitcoin in untrusted environments. Reference [27] proposed a scheme of providing escrow services by introducing a multi-signature which is adopted (t, n) threshold cryptosystem when cryptocurrency transaction in an untrusted environment. In addition, Tian and Feng [28] proposed a traceability system for food supply chain based on blockchain, but he did not deal with increasing data availability like key escrow function. Therefore, we have to find a way to provide key escrow encryption services in an untrusted environment such as [29].

In this paper, we propose a solution that provides traceability of all hardware components(parts) and data needed for system manufacturing by adopting blockchain technology and ensuring data availability. In particular, the key escrow encryption system is applied so that someone who requested information disclosure can check the information under the contract even in an untrusted environment.

III. REQUIREMENTS AND ASSUMPTION

This section describes the security requirements that the supply chain must meet to address the current supply chain system problems mentioned in the previous section I and II. And then describe how and why these requirements were derived.

A. SECURITY REQUIREMENTS

Supply chains, which the key element is traceability are mostly based on trust and many emerging technologies have been applied in such systems. However, most systems to date have been centralized and opaque, which can cause trust issues such as fraud, corruption, tampering and false information [28], [30]. These issues may arise because ownership of all data belongs to a corresponding manufacturer(supplier). Also, these problems happen from the internal, or external enemies maliciously nor intentionally. In addition, centralized database management creates a single point of failure [28]. We are able to derive some security requirements if look at these problems in detail. That is, supplier (or buyer) may take following actions to evade liability.

1. They can conceal the truth by manipulating transaction data. And they may argue that the buyer's data is wrong.

2. They can deny the transaction with the buyer. Alternatively, they can leverage the previous data (which were actually approved and used) to evade liability.

3. They accept the fact of the transaction, but they can impersonate an accident to damage, destroy, or delete the data. Or, it could be said that the data was damaged by an external attack.

4. It can be said that the data cannot be disclosed because it is confidential business data. (Even if it is required to disclose as claimed by the terms and conditions in the contract)

Based on this, it is possible to derive security requirements that complement the current system.

1. All data should not be changed without approval, and it must be proved that the data at the time of creation and the data at the current time are consistent (Integrity). Data should also be disclosed among both sides (Transparency).

2. The creator or sender of the data should be identified and remain unchanged (Non-repudiation). In addition, the time is recorded when data is written or modified, and it must be recorded in chronological order and traceable (Traceability, time order).

3. Data should always be available (Availability) at any situation abide by terms and conditions of contact, and overcome the single point of failure, traditional database problems (i.e. vulnerability to APT).

4. Sensitive data must be encrypted and protected (Confidentiality), but decrypted and disclosed by agreement, or otherwise, escrow function will be required.

5. The buyer must be able to obtain the encrypted data and decryption key so that the encrypted confidential information can be decrypted even in the absence of TTP where terms and conditions for contract are met. Malicious users may trigger denial of service (DoS) attacks or collusion attacks with other users to hamper the transmission of the secret key. Hence, the availability of the secret key is very critical to prepare for the abovementioned situations.

Hereby in such cases where there are no trusted third parties, the following additional requirements may be needed to cope with this situation: it is generally the primary rule to follow a majority vote, but the most concern is collusion attack. therefore,

6. When making decisions by a majority, incidents like collusion attack cannot exist. It should also be secured against DoS attacks.

Let's see how the proposed system satisfies the security requirements drawn up and overcome malicious actions in the scenarios and assumptions.

B. NOTATION

The notation used in this paper is shown in Table 3.

TABLE 3. Notation.

Alice: Sender (Manufacturer)
Bob: Receiver (Manufacturer or System Integrator or Contractor)
Trent: A Key Escrow agent group
Charlie: Key Recovery Requester (Buyer)
t_n: agent(node) n for key escrow
P : participants (could be Alice, Bob, Charlie and Trent)
$E_y(m)$: Encrypt m with public key y, (e.g. ElGamal)
$^*E_y(m) = (c_1, c_2)$ but we express as C here for convenience
$Enc_x(M)$: Encrypt M with symmetric key x (e.g. AES-256)
$Dec_x(M)$: Decrypt M with symmetric key x
$Sign_x(a)$: Digital signing a with key x
PK_x : public key of x (e.g. ElGamal)
SK_x : private key of x (e.g. ElGamal)
x_i : i-th private key share of T
y_i : i-th public key share of T
p, q : large prime number s.t. $q \mid p-1$
G_q : subset of Z_p^*
g, h : generator of G_q s.t. $log_g h$
$h()$: cryptography hash (i.e. SHA-256)
$H()$: $h() \bmod p$
Q : qualified entity

IV. OVERVIEW OF PROPOSED SOLUTION

This section proposes a solution for addressing the requirements described in section III. The proposed blockchain solution can trace all the components(parts), including hardware, software and firmware which are needed for the system, and prove that the components are provided from the certified manufacturers. Also, in an unreliable environment, non-classified transaction data is transferred to receiver in plaintext and sensitive data is encrypted and communicated. This proposed solution provides an environment where the session key used for encryption is possible to be recovered and decrypts cipher text according to the contract. Figure 1 shows the proposed framework, which consists of key escrow network, supply chain network and distributed storage network. Explanation of the data flow in the framework is as follows. (The number in circle in Figure 1 corresponds to the steps described as follows.)

1. Setup: Alice, who wants to communicate with Bob, obtains a public key (PK_T) for key recovery in the key escrow network (①).

2. Transaction: Alice encrypts the plaintext (M) to be sent with an arbitrary session key (Ks), then she obtains a cipher text (c). Also, Alice encrypts the session key (Ks) with the public key of Bob (PK_B) and Trent (PK_T) and creates signature of M with Alice's private key. Alice concatenates all outputs from created the above and stores the outputs in their own database or uploads to the distributed storage

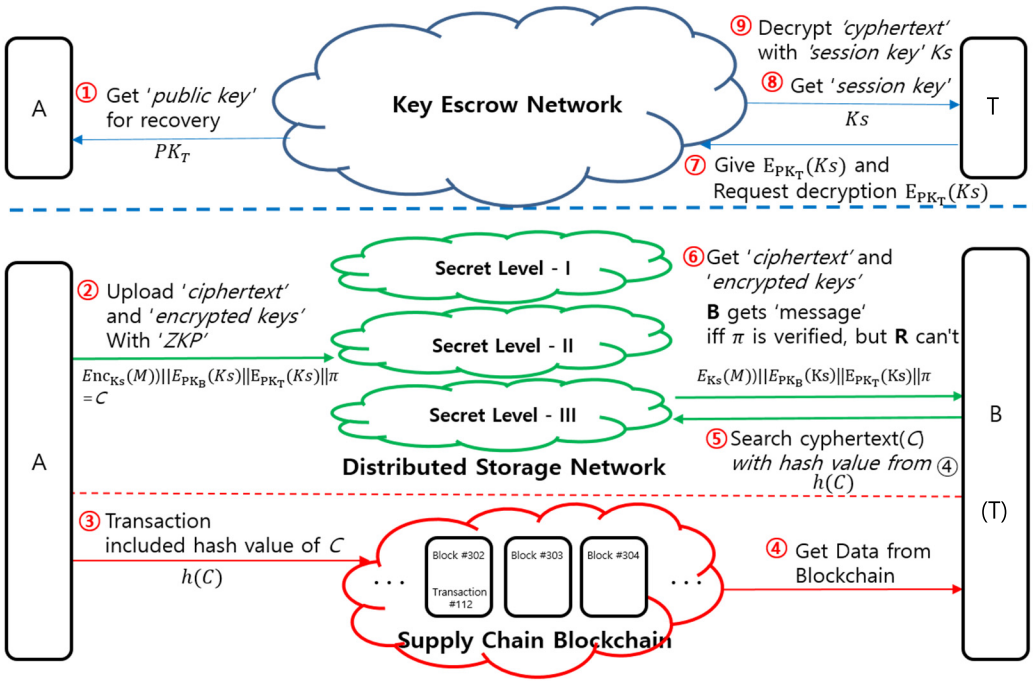

FIGURE 1. Our proposed solution.

network (②). Alice adds the hash value of data transmitted in ②, the hash value of the session key (*Ks*), and the hash value of the message into the transaction and records them in the block (③). Bob and Charlie obtain the data generated in ③ from the block in the supply chain network (④). Bob and Charlie find the desired set of cipher texts ② in the distributed storage network with the hash value obtained in ④. Then, Bob and Charlie obtain cipher text (*c*) and the encrypted key with the public key of the Bob and Charlie. Bob can acquire the session key(*Ks*) that could be decrypted with its own private key and decrypt the message, but Charlie cannot acquire the plaintext message because he does not have the private key(PK_T) of the Key Escrow Group for decryption(⑤⑥).

3. Key recovery: Charlie requests the private key of Key Escrow Agent Group (PK_T) in the key escrow network (⑦) when the contract term is satisfied. Then, he can decrypt ② from obtained the private key of Key Escrow Agent Group (⑧). Charlie gets the session key (*Ks*) with the private key (PK_T) of Key Escrow Agent Group(T) and decrypts the desired ciphertext (*c*) to obtain plaintext message(⑨).

A. KEY ESCROW NETWORK

It is a public network that anyone can participate if they meet certain criteria, and it is a completely separate network from the supply chain network and distributed storage network (when it is possible to select a public block chain having characteristics of decentralization, data anonymity, transparency, and integrity, it utilizes nodes itself of the public blockchain instead of the block). However, as in the previous assumption, there is a little change in network participants. It is a network for transmitting the session key to the requester, which encrypts important data such as the source code, design documents, etc., when contract conditions are met. The meaning of meeting the conditions depends on the contract, but when the seller or the manufacturer cannot provide the service properly because of bankruptcy, it is the case where the negation is confirmed and the investigation is necessary. Key escrow nodes (agents) are configured by applying *a (t, n)* threshold cryptography, rather than a single entity, to prevent denial of service attacks and collusion attacks It is possible to recover the key if t or more of all n nodes (agents) coincide). The participants and their roles of the key escrow network are shown in Table 4.

TABLE 4. Participants and their roles in key escrow network.

	Contract Setup		Contract Implementation	
	Generating private and public key shares of Key Escrow Group (T)	Generating private key of Key Escrow Group (T)	Requesting for decryption	Decryption
Sender(A)	O	-	-	-
Receiver(B)	O	-	-	-
Buyer(C)	-	-	O	-
Key Escrow Group(T)	-	O	-	O

B. DISTRUBUTED STORAGE NETWORK

Distributed storage networks must be accessible and available for all nodes. This accessibility can be achieved by using a distributed hash table in a blockchain network. This distributed storage network is similar to IPFS [8], BitTorrent [29]. The distributed storage networks can upload data that is required to be available in the supply chain or data that exceeds the maximum block size. It also distributes files corresponding to the hash value of the encrypted data such as source code. We can achieve the confidentiality of data by dividing and operating the network according to the classified security level of data.

C. SUPPLY CHAIN NETWROK

The supply chain network consists of private blockchain or consortium blockchain led by governments or companies that are responsible for the development of weapon systems. The supply chain network is also a network for keeping track of all components(parts) used in the development of weapon systems and ensuring the integrity of the related materials. Since this network is a private block chain, keys to be used in the network it must be authenticated by the administrator. Data on parts transactions are disclosed without encryption, sensitive data is encrypted and then stored in a distributed storage network, and its hash value is then recorded in a blockchain. Table 5 shows the participants and their roles in the supply chain network.

TABLE 5. Participants and their roles in supply chain network.

	Sign up	Permission (Key Issuance / Authentication)	Create a transaction	
			Session key generation, distribution, communication	Zero-knowledge proof
Sender(A)	O		O	O
Receiver(B)	△	△	O	O
Buyer(C)	O			
Admin		O		

The sign up and permission is the procedure for participating in the private blockchain, and the reason why the seller is marked as Δ is because the seller may become the Administrator too.

V. PROPOSED SOLUTION IN DETAILED

A. KEY ESCROW NETWORK

The Key recovery is required when: Alice and Bob share a session key with each other, and perform encrypted communication using the session key. Charlie, who does not have the session key, wants to decrypt the cipher text with which A and B have communicated if the contract conditions are met (i.e. When a manufacturer sends and receives cipher texts, and if security issues occur, the buyer would want to decrypt those cipher text where contract terms are met). The action in the key escrow network can be divided into two parts, as discussed previously.

1. Setup: The phase that establishes the contract terms and conditions, and obtains the public key of the key escrow group (T) in the key escrow network.

2. Key recovery: It is the phase where contract terms and conditions are fulfilled and the key escrow agent group (T) forwards their private key to the key recovery requester and the requester obtains the session key (Ks) and message with private key of key escrow agent group (T).

As the transaction phase is part of the supply chain network, it will be described in detail in the next sub-section.

1) SETUP

a: CONTRACT SETUP

This step is to set up the contract terms and conditions to specify when to recover the key. Since not all the contract terms can be presented quantitatively, we assume judgment from Key Escrow agents is needed for certain situations. In the event of dispute, the key escrow agents can execute the contract through investigation, and in this case, it is judged to be correct because it is agreed by threshold or more of agents.

b: KEY GENERATION OF KEY ESCROW AGENT GROUP

Once the key escrow agent group (T) are configured, T generates a private key (SK_T) and a public key (PK_T) to be used. The key is generated according to the distributed key generation [32], and a private key share (x_i) is first generated and a public key share (y_i) is calculated using the generated private key share (②). The key escrow agent group (T) then forwards the generated public key share (y_i) to the key recovery user (Alice) and the future key recovery requester (Charlie)(③). Alice and Charlie both who received the public key share, calculate the public key (PK_T) with the public key share(y_i)(④). Table 6 below shows how to create keys with the distribution key generation in [31].

2) KEY RECOVERY

When the contract terms are satisfied, each key escrow agent transmits a private key share (x_i) corresponding to the public key (PK_T) transmitted in step ① of Figure 1 to the key recovery requester (Charlie) at the request of the key recovery. Then, the key recovery requestor (Charlie) reconstructs the private key (SK_T) of the key escrow agent by combining each private key share using the Lagrangian interpolation. The key recovery requestor (Charlie) obtains the desired session key (Ks) and plaintext (M) using the reconstructed private key (SK_T).

However, this approach has the problem of not being able to reuse private key share. That is, the key recovery requestor (Charlie) having the private key (SK_T) of the key escrow agent group (T) can decrypt all cipher texts previously encrypted with the same public key (PK_T) of the key escrow

TABLE 6. Key generation of key escrow agents group.

Generation_shares_private_key_T (x_i)

1. Each party P_i perfoms Pedersen-VSS of secret z_i as a dealer
   (a) Choose random polynomials
   $f_i(z) = a_{i0} + a_{i1} z + \cdots + a z^t$
   $f_i'(z) = b_{i0} + b_{i1} z + \cdots + b_{i*t} z$
   over Z_q, $\leq t$, $z_i = a_{i0} = f_i(0)$,
   (b) Broadcast commitment $C_{ik} = g^{a_{ik}} h^{b_{ik}} \ mod \ p$
   for $k = 0, \cdots, t$ and
   send shares $s_{ij} = f_i(j) \bmod q$, and
   $s_{ij}' = f_i'(j) \bmod q$ to party P_j
   (c) Each party P_j verifies that $g^{s_{ij}} h^{s_{ij}'} = \Pi_{k=0}{}^{t} (C_{ik})^{j^k} \bmod p$
   (d) Resolution of received complaints from verification of the shares
2. Each party builds the set Q
3. Each P_i computes secret share: $x_i = \sum_{j \in Q} s_{ji} \bmod q$,

Extracting_shares_public_key_T (y_i)

Each party P_i exposes $y_i = g^{z_i} \bmod p$ via Feldman-VSS

1. Each party P_i broadcasts $A_{ik} = g^{a_{ik}} \bmod p$
   for $k = 0, \cdots, t$
2. Each party P_j verifies that $g^{s_{ij}} = \Pi_{k=0}{}^{t} (A_{ik})^{j^k} \bmod p$
3. Run reconstruction to compute z_l, $f_l(z)$, A_{lk} if P_l corrupted

Alice, Charlie:

Set $y_i = A_{i0} = g^{z_i} \bmod p$ and compute $y = \prod_{i \in Q} y_i \bmod p$

TABLE 7. Key generation of key escrow agents group.

Let assume message $m \in G_q$ is encrypted as $(g^k, y^k m)$,
where $y \in G_q$ is the corresponding public key and $k \in Z_q$

Charlie:

Give the cipher text(c) for recovery to T

Trent:

1. Each P_i broadcasts its decryption share $r_i = (g^k)^{x_i} \bmod p$ with NIZK that shows $log_g v_i = log_{(g^k)} r_i$,
   where $v_i = g^{x_i} \bmod p$ is a public verification key
   $$v_i = \prod_{j \in Q} \prod_{k=0}^{t} (A_{jk})^{i^k} \bmod p$$
2. Combine t+1 correct decryption shares by using Lagrange in terpolation in exponent
   $$m = \frac{y^k m}{\prod_{j \in \Lambda} r_j^{\lambda_{j,\Lambda}}} \bmod p$$

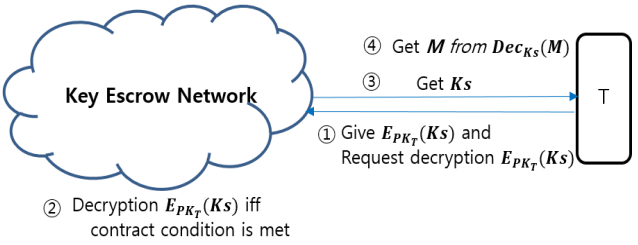

FIGURE 2. Key recovery phase.

agent group (T). To avoid this problem, the key escrow agent group (T) must use a different key for each key recovery user, which causes a huge overhead. In order to overcome this problem, Function Sharing concept [32] should be applied into key reconstruction and the key recovery step is applied as shown in Figure 2.

When the contract terms are satisfied, the key recovery requester (Charlie) transfers the cipher text ($E_{PK_T}(Ks)$) encrypted with the public key of the key escrow agent group (T) to the key escrow agent group (T) and requests decryption (①). The key escrow agent group (T) acquires the session key (Ks) using the distributed decryption method with shares (x_i) owned by each escrow agent (t_n) (②).

Then, the key escrow agent group (T) transfers the decrypted session key to the key recovery requester (Charlie) (③). Table 7 below shows the detailed procedure described above.

The key recovery requester (Charlie) obtains the private key (SK_T) in 3a, which can obtain the session key (Ks). Then, Charlie can decrypt the desired cipher text (c) which is acquired in the distributed storage network using the session key (Ks)(④).

$$M := Dec_{Ks}(Eec_{Ks}(M))$$

B. SUPPLY CHAIN NETWORK

The supply chain network is similar to Wal-Mart's supply chain based on Hyperledger fabric block chain provided by IBM. It can manage the parts of hardware and software that are being used in the system, where sources of parts can be traced. Figure 3 shows the concept of proposed supply chain.

1) ESTABLISHMENT

Since the network is a private block chain, participants who want to join the network are required to submit their information to Admin. The participants in this network are sellers, buyers, and all manufactures who produce hardware and software being used in the system. A user, who wants to join the network, creates the key pair (private key, public key) used in the supply chain network by itself and submits it to the administrator. The administrator shall issue the certificate that would be attached to the public key submitted by the user at the same time as the approval to join. Administrator can be a seller depending on the situation, and can also be the government of the nation selling the system.

2) BLOCKCHAIN TRANSACTION

A manufacturer (Alice) that transfers components(parts) create a transaction with a block structure as shown

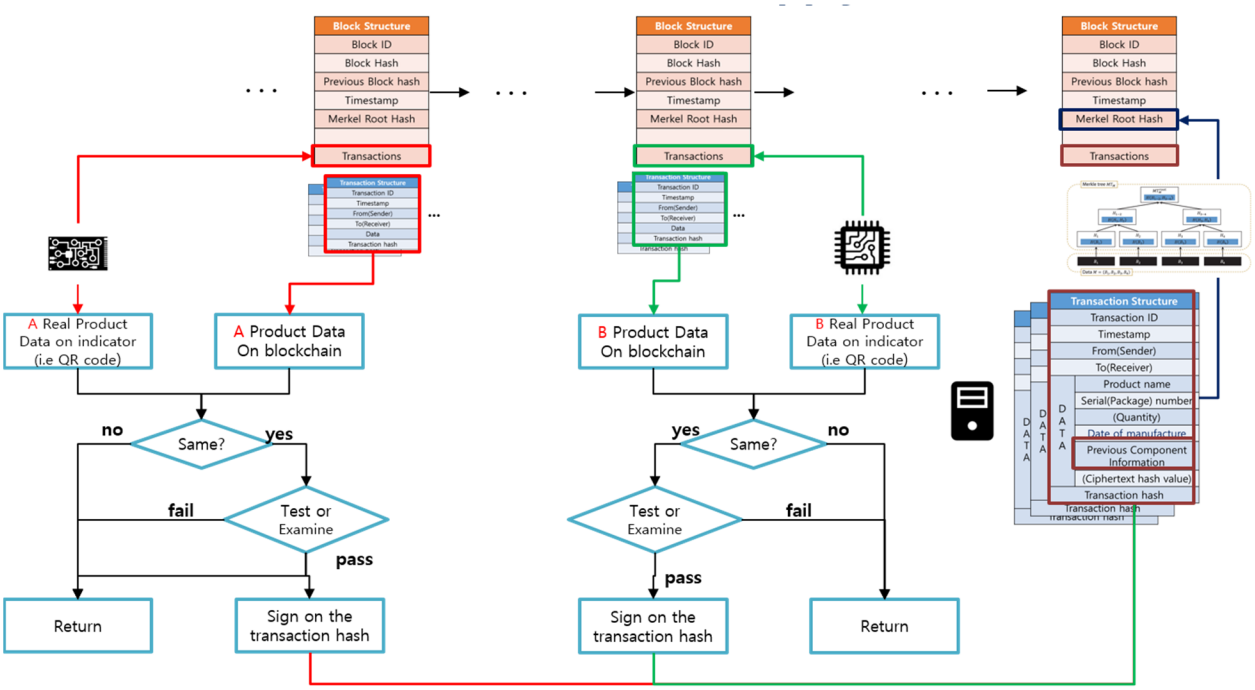

FIGURE 3. Proposed supply chain network.

in Figure 3 and writes the contents to the block. The block structure is similar to the bit coin block structure. Inside the block, the Merkle root hash value is the transactional hash value expressed in Merkle Tree format. The transaction sector basically includes the transaction ID and creation date and time of the transaction. In addition, the transaction records Sender, Receiver, Date of manufacture, and some unique information (i.e., system name, serial number, etc.) to identify the product. If the product you are sending is a very small parts (e.g., resistor, capacitor, etc.) that cannot generate a serial number, enter the serial number on package and the quantity contained in the package. Therefore, more parts than the quantity recorded in the corresponding transaction in the subsequent manufacturing process cannot be used. If you try to use more than the number of parts recorded in the transaction, the block verification will fail.

The 'Previous component information' field should contain information about what parts were used to deliver the product you want in the transaction. This is accomplished by inserting All previous transactions' hash values. This field also includes the hash value of software and firmware used in the currently manufactured products, if any.

When the transaction information is complete, the above information is then input and attached to distinguishable identifiers on the product (e.g., QR code, NFC, RFID, etc.) where it can be verified. At the time of confirmation, the receiver (Bob) compares the information on the blockchain with the information attached to the actual part to check whether they are the same. If the information is consistent, the receiver proceeds to the inspection and testing of the parts (products) that have been received. If there is no abnormality, the transaction hash value is digitally signed with receiver's key and could be added for the next product manufacturing. This information is served as a 'previous component information' for the next transaction.

3) TRANSACTION FOR KEY ESCROW

This section is related to the communication in the subsection IV-B key escrow network, and Figure 4 shows the flow of this communication step.

The sender (Alice) randomly chooses the session key (Ks) and generates a cipher text ($c = E_{Ks}(M)$) by encrypting (i.e., AES-256) the message (M) to be transmitted with the symmetric session key (Ks) (①). Then the sender (Alice) encrypts the session key (Ks) with the public key of Bob and Key escrow agent group ($E_{PK_B}(Ks)||E_{PK_T}(Ks)$) and uploads it to the network (②). This time, it should be uploaded to the network according to the classified level of the cipher text. Then, the sender (Alice) records the hash value of the data (C) transmitted in ② into the supply chain network (③). At this stage, an additional zero knowledge proof (ZKP,π) is attached. π is a zero-knowledge proof that both cipher texts ($E_{PK_T}(Ks)$, $E_{PK_B}(Ks)$) are encrypting the same message (Ks). If there is no ZKP value, sender (Alice) might encrypt Ks', not Ks, with key escrow agent group. Then, the key recovery requester will not know this unless he/she decrypts the communication every time. That is, if the ZKP (π) cannot pass by the verifiers, the communication will fail. The sender (Alice) must always prove to the recipient (Bob) and the Key Recovery Requester (Charlie) that they have encrypted

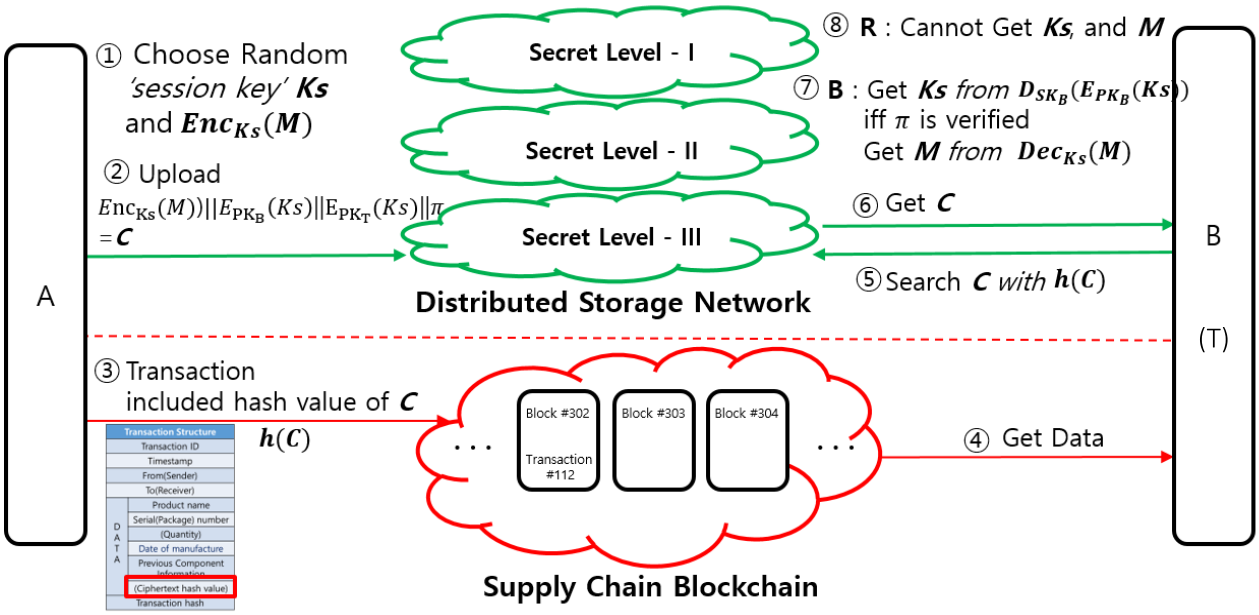

FIGURE 4. Transaction for Key Escrow.

the same session key. As mentioned earlier, if two ciphers are not encrypting the same session key, the key recovery network will be useless. Suppose the sender (Alice) sends the following value ($c_{2'} = E_{PK_B}(Ks')$, $c_{3'} = E_{PK_T}(Ks')$) to pass the verification, while this ZKP will pass the verification, the recipient (Bob) cannot decrypt the cipher text(c) because Bob has the wrong session key (Ks') instead of the correct session key (Ks). A proof method of verifying that c_2, c_3 are encrypting the same message which uses a zero knowledge proof of discrete logarithm and a zero-knowledge proof of equality of discrete logarithm [33], [34].

Afterwards, the recipient (Bob) and the key recovery requester (Charlie) obtain the hash value ($h(C)$) of the set of ciphertexts in the supply network (④) and the cipher text set (C) in the distributed storage network with hash value ⑤ and ⑥ respectively. The recipient (Bob) can acquire the session key (Ks) by decrypting the cipher text with his / her own private key (SK_B), and will be given the plaintext message (M) with this (⑦). However, the key recovery requester (Charlie) cannot acquire the plaintext message (M) because he does not have both keys, SK_T, SK_B (⑧).

VI. SECURITY ANALYSIS

This section evaluates how the requirements defined in section III are achieved based on set scenario. Then, we explain why the encryption scheme used in the proposed solution was selected.

A. CRYPTOGRAPHIC SCHEME

As described in section II, the contents in the first column of Table 8. were taken into account when selecting the cryptographic scheme. And the reason why Gennaro-DKG [31] was selected among them is that DKG protocol suggested by Pedersen and Torben Pryds [12] does not guarantee a uniformly random distribution of generated keys. In addition, Gennaro *et al.* states that the proposed scheme is suitable for key escrow service itself.

TABLE 8. Cryptography selection.

	Sharmir Secret Sharing [9]	Chor VSS [10]	Feldman VSS [11]	Pederson DKG [12]	Gennaro DKG [31]
Threshold cryptographic	O	O	O	O	O
Verifiable sharing	X	O	O	O	O
Non-interactive	-	X	O	O	O
Trusted third party	-	-	Required	Not Required	Not Required
Randomness	-	-	-	△	O

The symmetric key cryptography (e.g. AES) and one-way hash cryptography (e.g. SHA) used in the proposed solution are not specifically specified. The reason is that the method does not much affect the solution, and it is better way to use the appropriate method for the allocated resources. However, the public key encryption is recommended to use a discrete logarithm-based encryption technique. That is, the public key encryption technique, the DKG algorithm used in the Key Escrow Network, and the algorithm used in the

zero-knowledge proof are all linked to compute discrete-logarithm.

B. ARCHIVEMENT OF SECURITY ATRRIBUTES

In this subsection, we will examine how the proposed solution satisfies the security requirements derived from and prevents malicious behavior in the scenarios and assumptions established in section II.

1. **They can conceal the truth by manipulating transaction data. And they may argue that the buyer's data is wrong**: Data related to all supply chain transactions and key recovery are recorded on the supply chain blockchain. Therefore, according to the attributes of the blockchain, data cannot be modified without over-51-percent agreement in the network.

2. **They can deny the transaction with the buyer. Alternatively, they can leverage the previous data (which were actually approved and used) to evade liability**: All transactions data generated in the network left the record of the sender and the receiver on the blockchain, and the transaction hash value digitally signed. Thereby the non-repudiation property is accomplished. Also, the time stamp is applied to the block to be able to check the data generation time, and changing the order of blocks is also as difficult as modifying the contents of the block.

3. **They accept the fact of the transaction, but they can impersonate an accident to damage, destroy, or delete the data. Or, it could be said that the data was damaged by an external attack**: All data are stored on blocks, so those remain permanent. Although sensitive or large-size data is encrypted and stored in DSN, the hash value of the encrypted data is stored in the supply chain network. No data is lost even if only a few nodes are alive in the DSN. Therefore, the buyer (Charlie) can obtain the desired data with this hash value from the DSN at any time. Even if the blockchain system is composed of a private blockchain, it can be said to be more secured than a centralized database. In addition, data availability has been improved since all ciphers must utilize a key escrow network to decrypt by contract terms. (Even if a user lost the private key, the key could be recovered.)

4. **Sensitive data must be encrypted and protected (Confidentiality), but decrypted and disclosed by agreement, or otherwise, escrow function will be required**: In the proposed solution, all the general transaction data is publicly recorded in blocks. manufacturer/developer (Alice) encrypts sensitive data or large transaction data and store it in the DSN. Therefore, those data can be decrypted and read only by the user who owns the key. The decryption key is owned by the manufacturer / developer (Alice) and the key escrow agent (Trent). The key escrow agent (Trent) reconstructs the decryption key, when the manufacturer / developer (Alice) cannot or do not intentionally fulfill the contract, and delivers it to the buyer (Charlie). Therefore, the contract can be executed regardless of manufacturer's intend (Alice)

5. To prevent malicious actions of key escrow agents, key escrow agents cannot access the cipher text because they have no access to the distributed storage network, regardless that they know the hash value of cipher text. In the key escrow network, we applied *t of n* threshold cryptosystem to prevent DoS attacks or collusion attacks by agents (Trent). Therefore, if more than t nodes can operate, the system can function normally.

VII. CONCLUSION

In this paper, we propose a more secured solution for long-lifecycle systems (i.e. weapon systems) using blockchain. The proposed one can track the components(parts) needed for system manufacturing by introducing the blockchain, and we also proposed a way to acquire sensitive data (e.g. source code, design documents or large-size data) in an untrusted environment through key escrow function. Under this scenario, data's integrity, availability and traceability are enhanced and at the same time, single point failure can be resolved. Its design can also safeguard non-repudiation and Denial of Service attacks. However, since the efficiency of the system is not discussed, it is necessary to further study the efficiency of the system in the future.

REFERENCES

[1] M. Howard and S. Lipner, *The Security Development Lifecycle*, vol. 8. Redmond, WA, USA: Microsoft Press, 2006.

[2] BBC News, BBC. *US Targets Huawei With Tighter Chip Export Rules*. Accessed: May 15, 2020. [Online]. Available: https://www.bb.co.uk/news/business-52681414

[3] Defense Advanced Research Projects Agency. *Defense Advanced Research Projects Agency*. Accessed: May 2020. [Online]. Available: https://www.darpa.mil/news-events/2020-05-27

[4] M. Warren and W. Hutchinson, "Cyber attacks against supply chain management systems: A short note," *Int. J. Phys. Distrib. Logistics Manage.*, vol. 30, nos. 7–8, pp. 710–716, Sep. 2000.

[5] E. A. Fischer, "Cybersecurity issues and challenges: In brief," Congr. Res. Service, Washington, DC, USA, CRS Rep., Dec. 2014.

[6] M. Crosby, "Blockchain technology: Beyond bitcoin," *Appl. Innov.*, vol. 2, nos. 6–10, p. 71, 2016.

[7] M. Pilkington, "Blockchain technology: Principles and applications," in *Research Handbook on Digital Transformations*. Cheltenham, U.K.: Edward Elgar Publishing, 2016.

[8] J. Benet, "IPFS–content addressed, versioned, P2P file system," 2014, *arXiv:1407.3561*. [Online]. Available: http://arxiv.org/abs/1407.3561

[9] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[10] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," in *Proc. 26th Annu. Symp. Found. Comput. Sci. (SFCS)*, 1985, pp. 383–395.

[11] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proc. 28th Annu. Symp. Found. Comput. Sci. (SFCS)*, Oct. 1987, pp. 427–438.

[12] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 1991, pp. 129–140.

[13] D. E. Denning and D. K. Branstad, "A taxonomy for key escrow encryption systems," *Commun. ACM*, vol. 39, no. 3, pp. 34–40, Mar. 1996.

[14] J. Nechvatal, "A public-key-based key escrow system," *J. Syst. Softw.*, vol. 35, no. 1, pp. 73–83, Oct. 1996.

[15] S. Nakamoto and A. Bitcoin. (2008). *A Peer-to-Peer Electronic Cash System*. Bitcoin. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[16] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, pp. 1–32, Apr. 2014.

[17] R. Casado-Vara, J. Prieto, F. D. la Prieta, and J. M. Corchado, "How blockchain improves the supply chain: Case study alimentary supply chain," *Procedia Comput. Sci.*, vol. 134, pp. 393–398, Jan. 2018.

[18] *Everledger*. Accessed: Jun. 2019. [Online]. Available: https://diamonds.everledger.io/
[19] T. Bocek, B. B. Rodrigues, T. Strasser, and B. Stiller, "Blockchains everywhere—A use-case of blockchains in the pharma supply-chain," in *Proc. IFIP/IEEE Symp. Integr. Netw. Service Manage. (IM)*, May 2017, pp. 772–777.
[20] Provenance, Project Provenance. *Blockchain: The Solution for Transparency in Product Supply Chains*. [Online]. Available: https://www.provenance.org/whitepaper
[21] H. R. Hasan and K. Salah, "Blockchain-based solution for proof of delivery of physical assets," in *Proc. Int. Conf. Blockchain*. Cham, Switzerland: Springer, 2018, pp. 139–152.
[22] R. Kamath, "Food traceability on blockchain: Walmart's pork and mango pilots with IBM," *J. Brit. Blockchain Assoc.*, vol. 1, no. 1, p. 3712, 2018.
[23] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, "A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain," *IEEE Access*, vol. 5, pp. 17465–17477, 2017.
[24] J. McCarter, "DON innovator embraces a new disruptive technology: Blockchain," Dept. Navy, Naval Innov. Advisory Council, Washington, DC, USA, Tech. Rep., 2017. [Online]. Available: https://www.doncio.navy.mil/Chips/ArticleDetails.aspx?ID=9144
[25] M. Blaze, "Protocol failure in the escrowed encryption standard," in *Proc. 2nd ACM Conf. Comput. Commun. Secur. (CCS)*, 1994, pp. 59–67.
[26] A. K. Lenstra, P. Winkler, and Y. Yacobi, "A key escrow system with warrant bounds," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 1995, pp. 197–207.
[27] S. Goldfeder, J. Bonneau, R. Gennaro, and A. Narayanan, "Escrow protocols for cryptocurrencies: How to buy physical goods using bitcoin," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Cham, Switzerland: Springer, 2017, pp. 321–339.
[28] T. Feng, "A supply chain traceability system for food safety based on HACCP, blockchain & Internet of Things," in *Proc. Int. Conf. Service Syst. Service Manage.*, Jun. 2017, pp. 1–6.
[29] B. Cohen, "Incentives build robustness in BitTorrent," in *Proc. Workshop Econ. Peer-to-Peer Syst.*, vol. 6, 2003, pp. 68–72.
[30] K. Korpela, J. Hallikas, and T. Dahlberg, "Digital supply chain transformation toward blockchain integration," in *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, 2017, pp. 4182–4191.
[31] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin, "Secure distributed key generation for discrete-log based cryptosystems," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1999, pp. 295–310.
[32] A. De Santis, Y. Desmedt, Y. Frankel, and M. Yung, "How to share a function securely," in *Proc. 26th Annu. ACM Symp. Theory Comput. (STOC)*, 1994, pp. 522–533.
[33] D. Chaum and T. P. Pedersen, "Wallet databases with observers," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 1992, pp. 89–105.
[34] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, Jan. 1991.

SUNGYONG CHA received the B.S. degree in computer science from the Korea Military Academy, in 2004, the M.S. degree in electrical engineering with SUNY at Buffalo, USA, in 2008, and the Ph.D. degree in information security from Korea University, in 2019. He is currently working in cybersecurity with the Korea Ministry of National Defense. His research interests include information assurance, C4I, risk management, and SDLC.

SEUNGSOO BAEK received the B.S. degree in computer science from the Korea Military Academy, in 2002, the M.S. degree from the U.S. Naval Postgraduate School, in 2007, and the Ph.D. degree in information security from Korea University, in 2018. He was a Lecturer teaching on cyberwar and information security with the Korea Military Academy and Korea University. He is currently the Head Researcher of IT business development with Nanobrick that is specialized in security materials and services. His research interest includes industrial applications on information security.

SEUNGJOO KIM (Member, IEEE) was an Associate Professor with Sungkyunkwan University. He was a Team Leader with the Korea Internet & Security Agency (KISA). He is currently a Professor with the School of Cybersecurity, Korea University. He is also the Director of the Center for High-Assurance Operating Systems (CHAOS), the Head of Security Assessment aNd Engineering (SANE) Laboratory, an Adviser of the Undergraduate Hacking Club CyKor (DEFCON CTF 2015 & 2018 winner), School of Cybersecurity, Korea University, from 2011 to February 2020, and a Founder/Advisory Director of the International Security & Hacking Conference SECUINSIDE. His research interest includes trustworthy system development methodology (a.k.a. secure software development lifecycle). His numerous professional focus on a presidential committee member on the 4th Industrial Revolution and an Advisory Committee Member of several public and private organizations, such as National Intelligence Service (NIS), Ministry of National Defense, Ministry of Justice, Supreme Prosecutors' Office, Korea National Police Agency, Nuclear Safety and Security Commission, and so on. He also taught at the Korea Military Academy.