SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 4, (2026), pp. 317-341
Open Access. Online & Print

DOI: 10.5281/zenodo.1250030

SECURING BLOCKCHAIN-ENABLED SUPPLY CHAIN MANAGEMENT AGAINST HASH BASED ATTACKS USING EFFICIENT SCHEMES

[1]Ritika Shrimali and [*2]K. Kanagalakshmi

[1]Research Scholar, CMR University, Bangalore, Karnataka, India
*[*2]Professor, School of Science and Computer Studies, CMR University, Bangalore, Karnataka, India*
[*]kkanagalakshmi@gmail.com

Received: 01/12/2025
Accepted: 30/03/2026

Corresponding Author : K. Kanagalakshmi (kkanagalakshmi@gmail.com)

ABSTRACT

Supply chain is an interconnection of organizations, people and information for transforming raw materials into finished products. Modern supply chains have evolved into globally distributed ecosystems that demand high levels of data integrity and system transparency. Traditional Supply Chain Management (SCM) depends on centralized and opaque infrastructures, struggle to manage these realities effectively. To overcome this limitation, this research proposed a novel consensus-based blockchain mechanism and cryptographic encryption (ECBM-CE) framework to improve security, scalability and operational efficiency in the pharmaceutical supply chain. Existing blockchain-based supply chain solutions often experience high transaction latency and limited throughput. To overcome these challenges, the proposed approach introduces a forked blockchain architecture integrated with smart contracts and an upgraded delegated proof-of-stake (Up-DPoS) consensus mechanism, which improves transaction validation efficiency while maintaining decentralization and security. A key innovation of this work is the integration of SHA-256 hashing with an extended tiny encryption (ETE) scheme prior to data storage, providing an additional layer of confidentiality and resistance to data tampering beyond conventional blockchain implementations. The ECBM-CE framework jointly optimizes data security and network performance. The system enables transparent and traceable tracking of pharmaceutical products across the supply chain through automated smart contracts. The proposed framework is implemented in Python and evaluated using multiple performance metrics. Experimental results demonstrate that the ECBM-CE attains a higher throughput of 8.2165 transactions per second and a reduced network latency of 223.65 ms. These results confirm the effectiveness of the proposed innovations in advancing secure and efficient pharmaceutical supply chain management.

KEYWORDS: Supply chain management, Extended tiny encryption, SHA 256, Upgraded delegated proof of stake, Consensus algorithm, Smart contracts and Forked blockchain

1. INTRODUCTION

Supply chain management controls the entire manufacturing process, including the flow of supplies, information, and money. It monitors every stage till the completed product or item reaches its final location. Supply chain management reduces the extra cost of getting the items to the client [1]. A healthy, well-managed supply chain is essential for any industry's uninterrupted operation. Small supply chain disruptions can impact the market completely and result in significant money loss for the affected organizations [2]. A faultless supply chain is compulsory to identify the origins of fake products that have made their way to the consumer [3]. Several techniques have been developed to detect products in supply chain management systems, which use various tags like Radio Frequency Identification (RFID) tags, barcodes, as well as so on. These tags have limited features but are cost-effective for managing the supply chain. It is one of the most difficult responsibilities in the supply chain management system since it affects product privacy and transparency [4, 5]. A well-organized supply chain network allows the entire firm to ensure data track synchronization. It may also reduce the time and costs involved with the utilization of smart contracts. The application of blockchain technology in distribution networks not only increases efficacy as well as decreases costs but also strengthens ties among all chain participants [6, 7].

Blockchain technology has restored trust among the system's distributed components by introducing new currencies (such as Bitcoin), digital contracts that enforce themselves automatically (including smart contracts), and intelligent assets that are controlled and monitored via the Internet [8]. The majority of current blockchain research focuses on establishing affordable applications in a variety of sectors. Blockchain technology is currently utilized in supply chain management to store as well as process records and chain-transaction data, hence improving efficiency, trust, and transparency while lowering overall supply chain costs [9]. Academics and industry have used blockchain and its associated technologies to develop various supply chain systems because they ensure data integrity and protect it from manipulation assaults by chaining data in a secure hash method [10]. Transactions in several shards can be carried out concurrently by splitting the blockchain network into separate shards, increasing the blockchain's throughput [11]. But sharding also raises security concerns. The hash value of a transaction determines the output shard in a hash-based transaction sharding system. High levels of precision, trust, and transparency can be attained with blockchain technology, and it can also provide real-time tracking of goods, information, owners, as well as actions taken at every stage [12]. The important roles that trust, traceability, as well as transparency play in supply networks have been thoroughly studied in the scientific literature, which also demonstrates how these characteristics can significantly enhance supply chain performance [13]. Sustainability and operations management are among the supply chain management challenges that have been addressed. There is insufficient structure to produce optimal results in terms of efficiency, efficacy, and long-term sustainability [14]. Supply chain management systems oversee every stage of the process, from raw material production to consumer delivery of the finished product. To optimize transparency, all stakeholder transactions in this flow must be recorded in real-time and distributed to connected parties over a trusted and safe infrastructure [15]. Blockchain technology is becoming a popular solution for improving supply chain security as well as transparency. Because centralized management techniques lead to inefficiencies, a lack of transparency, as well as security concerns, traditional supply chains suffer [16]. Blockchain technology may assist in managing these interactions by improving asset use, stakeholder collaboration, and inventory management [17]. Hash functions provide a compact representation of data regardless of the input data size, which enables efficient storage and transmission. Hash functions are computationally efficient to compute; they are computationally infeasible to reverse [18].

Thereby, blockchain-based supply chain management is a revolutionary solution that resolves fundamental problems in traditional supply chains, comprising delays, product tampering, and security risks [19, 20]. The importance of this research is its capability to improve transparency, data integrity, and decentralization to guarantee that sensitive products are transferred securely from producers to end-users. Current solutions, like barcode-based tracking and RFID tags, are cost-effective for management but lack robust security features, which exposes them to tampering and fraud. Furthermore, classical supply chain systems do not have real-time synchronization, which tends to lead to inefficiencies and higher operational costs. As a result, the proposed solution improves the supply chain management system's security by combining a unique consensus-based blockchain technology with

an effective cryptographic algorithm. The proposed method combines smart contract-based forked blockchain technology with secure hash algorithm-256 (SHA-256) hashing and an improved lightweight encryption algorithm to counter hash-based attacks and enhance security. In addition, an enhanced consensus algorithm provides secure transaction verification, mitigating risks associated with unauthorized modifications. By resolving these limitations, this work provides an enhanced, more secure, efficient, and transparent supply chain management system that upsurges stakeholders' confidence and diminishes disruptions. **The key objectives are given below:**

- To propose an effective consensus-based blockchain mechanism and cryptographic encryption (ECBM-CE) method in pharmaceutical supply chain systems.
- To ensure data security, a SHA-256-based hashing and lightweight extended tiny encryption (ETE) algorithm will be used to generate a hash for logistic information and encrypt it to mitigate hash-based attacks in supply chain management.
- To incorporate a smart contract-based forked blockchain structure to enable secure transaction validation and tracking of goods within pharmaceutical supply chain systems.
- To offer improved blockchain verification through Upgraded Delegated Proof of Stake (Up-DPoS) to validate blockchain transactions more efficiently. This upgraded consensus process optimizes transaction processing speed, minimizes computational costs and improves security.
- To assess the performance of the ECBM-CE technique by computing the varied evaluation metrics and comparing the results with other existing algorithms.

The following describes the remaining paper structure: The relevant studies addressing the supply chain management problems are acknowledged in Section 2. In Section 3, the proposed model for securing blockchain-enabled supply chain management is discussed. The outcome and discussion are stated in Section 4. Section 5 brings an illustration of the conclusion as well as future directions.

2. RELATED WORKS

Piera Centrobelli et al. [21] created the integrated Triple Retry architecture for designing a circular blockchain network. It is developed for the supply chain, which comprises reverse logistics service providers, recycling centers, landfills, manufacturers, and a selection center. The blockchain technology used in this process was evaluated for the three key factors (trust, traceability, as well as transparency) that influence circular supply chain operations, which will be investigated: recycling, redistribution, and remanufacturing.

Nafisa Anjum and Pramit Dutta [22] developed a decentralized Blockchain-based application system (DApp) to identify counterfeit products in the supply chain system. This technique was developed due to the immutable and security of blockchain. Then, a Quick Response (QR) code produced by DApp was utilized to verify production distribution and the existence of every product linked to the blockchain.

Lufei Huang et al. [23] created a collective analytical hierarchy process (AHP) as well as a decision-making trial as well as evaluation laboratory (DEMATEL) technique to assess priorities as well as relationships of success variables for academic and professional experts. This technique is used to examine possible paths to identify crucial success elements and promote blockchain-enabled circular supply chain management.

Pedro Azevedo et al. [24] developed supply chain traceability through blockchain to assure the chain of custody as well as traceability. It enables the organization to show the product's origins, compliance, as well as integrity. This technique's genuine traceability connects supply chain actors (SCAs) as well as product identifications through the use of digital certificates. Certificates are imported, stored, and verified using an off-chain data storage system. A Public Key Infrastructure (PKI) is meant to validate certificates and establish a chain of trust. This model's final design consists of an Ethereum Smart Contract as well as a PKI-based certificate authentication system.

Pratyush Kumar Patro et al. [25] created a private Ethereum blockchain-based solution for managing fishing supply chain operations that is visible, traceable, decentralized, private, trustworthy, and safe. Five smart contracts were created to help automate the fishing supply chain operation. This method utilized ten algorithms to test and validate the performance in terms of different evaluation measures. Then, a security analysis was performed to ensure that this solution was both secure and trustworthy.

Visuvanathan et al. [26] suggested for internet of things (IoT) based on Intrusion Detection in

vehicular networks. To implement by blockchain enabled hierarchical factorized variation autoencoder for secure intrusion detection in vehicular network (BO- HAFEL) model, which uses a multi-level learning process for edge, fog, as well as cloud. In this intrusion detection model, use for a dataset in UNSW-NB15 as well as CIC-IDS-2017 datasets are used in a hierarchical training process is utilized. In this experimental achieve, an accuracy value by 96.83% and 97.36% are performed. Furthermore, this method creates an environment that faces complex challenges, includes secure communication and protecting users' privacy in dynamic as well as hostile environments.

A wireless sensor network for multi-factor authentication for hash-based attack detection is presented by Sholapurapu et al. [27]. Cuckoo Hash-Based Multi-Factor Authentication (CH-MFA) structure, which uses registration and authentication as part of a two-phases procedure for secure communication. This experiment demonstrates that hash-based assaults obtained an accuracy rating of 99%. However, the computational and energy expense of this hash authentication is increasing, which makes it less appropriate for wireless sensor nodes with limited resources.

Curado et al. [28] developed Zero-Knowledge Proofs (ZKPs), which make it possible to validate transactions and documents without disclosing private information. This suggested method is used for secure ZKP integration, document verification, fraud mitigation and regulatory compliance. This model assesses adoption hurdles, computational cost, and scalability and suggests an architecture backed by simulation-based validation utilizing Ethereum and ZoKrates to gauge performance and viability. This method faced some challenges, including scalability issues, high cost and interoperability issues. The contributions and limitations of existing methods are presented in Table 1.

Table 1: Contribution and limitations of existing methods

Author name and reference	Technique used	Performance analysis and features	Limitation
Piera Centrobelli et al. [21]	Integrated Triple Retry framework	The performance analysis on trust, traceability and transparency	Require an in-depth technique for a single circular supply chain
Nafisa Anjum and Pramit Dutta, [22]	DApp	Security, privacy, Decentralization, Transparency	This technique is more expensive
Lufei Huang et al. [23]	AHP and DEMATEL method	Data security, technological feasibility, and cost control	This technique has limitations regarding generalization
Pedro Azevedo et al. [24]	Ethereum Smart Contract, as well as a PKI-based certificate authentication system	Provenance, traceability and chain of custody	The supply chain lacks the functionality for every SCA to include product certificates while transforming products.
Pratyush Kumar Patro et al. [25]	a private Ethereum blockchain-based solution	Transparency, Decentralized, Traceability, Availability, Data integrity and so on	This technique is only used to trace the fish product in the fishery supply chain.
Visuvanathan et al. [26]	BO- HAFEL	-	The environment faces complex challenges, such as secure communication and protecting users' privacy.
Sholapurapu et al. [27]	CH-MFA	CH-MFA authentication capabilities through its superior	Hash authentication is increasing the computational and energy overhead, making it less suitable.

		scalability and reliability factors while upholding operational efficiency	
Curado et al. [28]	ZKPs	Improve security and privacy in the maritime supply chain, achieving transparency.	This method faced some challenges, including scalability issues, high cost and interoperability issues.

2.1 Problem statement

A fork in a blockchain-based supply chain management system can occur when members disagree on the legality of transactions or the blockchain's rules. A fork causes the blockchain to split into two or more competing versions, each with a distinct set of rules. Hash functions are critical to blockchain security because they secure the integrity of data within each block and provide a link across blocks via cryptographic hashing. However, manipulating hash functions to compromise the security of a forked blockchain in a supply chain management system would be highly challenging. Therefore, it is important to design a secured framework for reducing the security risks in the forked blockchain. In recent years, cryptographic methods and consensus algorithms have become more popular because of the benefits of affording higher security to the systems based on their significant working properties. Thus, it motivates us to use an effective cryptographic approach for encrypting the data and a consensus algorithm for validating the blockchain transaction in this study.

3. PROPOSED METHODOLOGY

This section discusses a novel consensus mechanism with cryptographic encryption (ECBM-CE) method for compromising the integrity of a forked blockchain in a supply chain management system by manipulating its hash functions. For supply chain management, logistic information is considered in this work, and the most crucial steps involved in the ECBM-CE method are data acquisition, hash value generation, encryption, blockchain storage and verification. To create an efficient supply chain, transaction data such as the supplier's location, the name of the raw material, and its quantity and quality are first gathered. Blockchain storage is essential to secure logistical information. The SHA 256 algorithm is utilized to produce hash values for storing information in blocks. For blockchain-based supply chain management, the presence of a fork in the blockchain raises the danger of assaults during hash function development. Thus, the generated hash values are encrypted by using a new extended tiny encryption (ETE) approach. Such encrypted data is then securely stored using smart contract-based blockchain technology. Finally, an Up-DPoS consensus mechanism is used to validate blockchain transactions. Thus, using the proposed framework, hash-based assaults on forked blockchains in supply chain management are reduced. The block diagram of the ECBM-CE technique is illustrated in Figure 1.

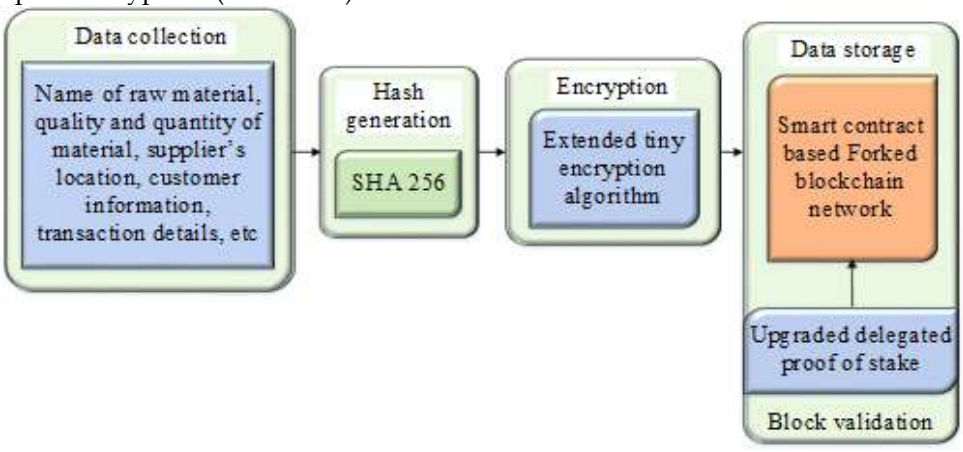

Figure 1: Block diagram of the proposed method

3.1 Hash value generation based on SHA 256

SHA-256 is a member of the secure hash algorithm family, established by the National Security Agency (NSA) as well as released by the National Institute of Standards and Technology (NIST). It is commonly used for digital signatures and data integrity verification. Deriving the actual data from the hash value is nearly impossible because each single input generates a unique hash. In the proposed method, SHA-256 is utilized to generate hash values for storing information in blockchain-enabled supply chain management because of a secure cryptographic foundation that guarantees data authenticity, integrity, and security. It integrates a 256-bit hash function with collision resistance. In supply chains, SHA-256 is a core component in transactional and record security. Its deterministic quality assures that the same input will always yield the same output, which is essential for supply chain data verification and tracing over decentralized networks. Besides, the SHA-256 is computationally efficient and broadly supported across blockchain platforms, making it easier to assimilate into supply chain applications. Although SHA-256 is superior in security, its computational demands can be high, with the necessity for substantial processing power in large-scale operations. Additionally, it is advantageous to remain vulnerable to future threats from quantum computing. In spite of these factors, SHA-256 continues to be an essential option for securing supply chains based on blockchain technology owing to its tamper-resistance, proven reliability, and support for immutable and transparent record-keeping.

SHA-256 accepts input and generates a fixed-size 32-byte (256-bit) hash value, which is often revealed as a 64-character hexadecimal number [29]. The following ways offer how the SHA-256 algorithm [29] determines a digest of any message length. The input message, which has an arbitrary length of M bits, is transformed to binary form and padded with leading $'0's$ and $'1's$ until its length equals 448 modulo 512. Then, add the message length M's 64-bit binary representation to make a multiple of 512 bits. This lengthy padded message is split into 512-bit blocks, $E^1, E^2, ..., E^P$, after padding. Then, the main function processes every 512-bit block of data consecutively during 64 rounds. The next 512-bit data block is provided if the data block E^j has been fully processed and a partial 256-bit hash value H^P is calculated. After the processing of the final data block E^P, the final hash value H^P is attained. The SHA-256 makes use of 64 constants, each with a size of 32 bits and 8 initial hash values $H_0^0, H_1^0, ,, ..., H_7^0$. The fundamental structure of the SHA-256 procedure is as follows:

- Require: E^j, a 512-bit data block as well as initial hash values.
- Verify that E^j from H^{j-1} corresponds to hash value H^j.
- for each $j = 1$ to P (the number of 512-bit data blocks) do
- for $u = 0$ to 63 do
- Preparing the message schedule X_u
- Determine $\sum_0 a$ and $\sum_1 e$.
- Compute Tmp_1, Tmp_2, $CH(e,f,g)$ and $MAJ(a,b,c)$
- $h \leftarrow g$

$g \leftarrow f$

$f \leftarrow e$

$e \leftarrow d + Tmp_1$

$d \leftarrow c$

$c \leftarrow b$

$b \leftarrow a$

$a \leftarrow Tmp_1 + Tmp_2$

End for

- Determine $H_{temp} = a \mid b \mid c \mid d \mid e \mid f \mid g \mid h$
- Calculate $H^j = H^{j-1} + H_{temp}$
- Return H^j

End for

Every function comprised in the SHA-256 procedure can be determined using the equations below:

$$Maj(a,b,c) = (a \wedge b) \oplus (\bar{a} \wedge c) \quad (1)$$

$$Ch(e,f,g) = (e \wedge f) \oplus (e \wedge g) \oplus (f \wedge g) \quad (2)$$

$$\sum_0 a = ROTr(a,2) \oplus ROTr(a,13) \oplus ROTr(a,22) \quad (3)$$

$$\sum_1 e = ROTr(e,6) \oplus ROTr(e,11) \oplus ROTr(e,25) \quad (4)$$

$$Tmp_1 = h + \sum_1 e + Ch(e,f,g) + K_u + X_u \quad (5)$$

$$Tmp_2 = h + \sum_0 a + Maj(a,b,c) \quad (6)$$

where, $ROTr$ indicates rotation right through specified bits, $\oplus$ and $\wedge$ indicates logical XOR as well as AND operation, whereas $+$ resembles modulo 2^{32} addition.

The following procedure can be utilized to calculate the message schedule X_u that is employed in each transformation:

$$X_u = \begin{cases} E_u^j & 0 \le u \le 15 \\ \sigma_1(X_{u-2}) + X_{u-7} + \sigma_0(X_{u-15}) + X_{u-16} & 16 \le u \le 63 \end{cases} \quad (7)$$

As illustrated in the above equation, X_u is comparable to a 512-bit message for the first 16 rounds. Besides, it is computed utilizing functions σ_0 and σ_1 for the next 48 rounds.

$$\sigma_0(X) = ROTr(X,7) \oplus ROTr(X,18) \oplus SHr(X,3) \quad (8)$$

$$\sigma_1(X) = ROTr(X,17) \oplus ROTr(X,19) \oplus SHr(X,10) \quad (9)$$

where, SHr resembles the shift hand right by specified bits.

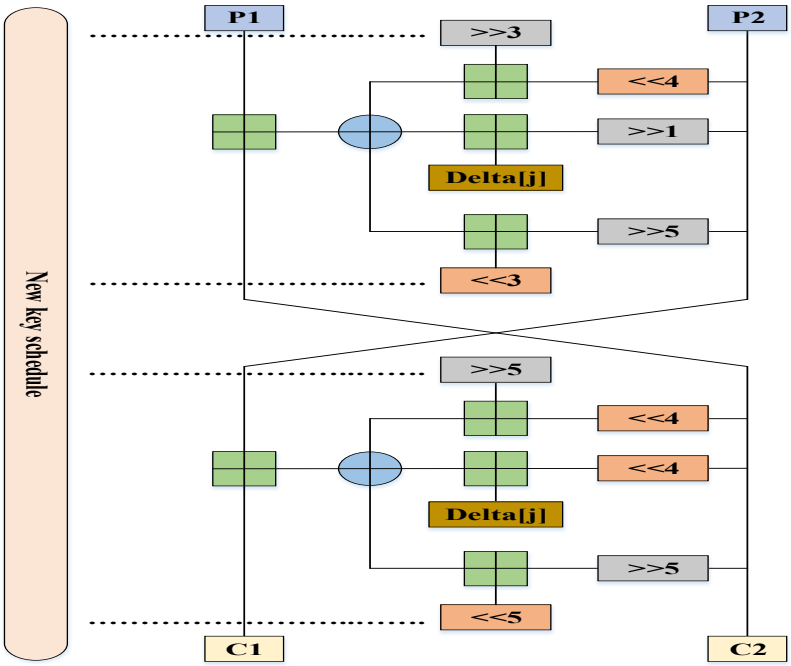

Figure 2: Block diagram of ETE algorithm

3.2 Extended Tiny Encryption

The tiny encryption algorithm (TEA), developed by Roger Needham as well as David Wheeler of Cambridge Computer Laboratory, is an effective as well as lightweight symmetric key block cipher. TEA is increasingly popular because of its small code size and ease of use. It uses a 128-bit key and works with 64-bit plaintext blocks [30]. Furthermore, this encryption method is easy and quick to implement in software and hardware since it employs 64 rounds of Feistel network operations, which include bitwise shifts, XORs, and basic arithmetic operations.

TEA provides adequate security despite its simplicity. Furthermore, TEA is a popular choice for lightweight encryption operations due to its efficacy and ease of implementation. TEA has been proven to be appropriate because of its modest size; however, it is vulnerable to comparable key and associated key attacks. As a result, an extended tiny encryption (ETE) method is described for encrypting the resulting hash using a novel key scheduling strategy. ETE is an appropriate algorithm to use in blockchain-based supply chain management due to its improvements to security, lightweight properties, as well as efficient performance in constrained environments. As discussed earlier, the original TEA algorithm contained possible vulnerabilities because of related key and equivalent key attacks, while ETE utilizes key rotation and modified round functions to improve security. The improvements to ETE allow for better diffusion and security against cryptanalysis, proving it to be a better choice for use in supply chain compatibility due to the need for secure as well as tamper-proof data sharing. Furthermore, the lighter memory footprint and lower computational overhead of ETE make it more desirable for lower-powered systems such as IoT devices or devices with lower capabilities found in supply chain applications. Overall, ETE offers a better level of security, but it still suffers from some limitations, like having a very low-key expansion compared to other encryption algorithms and potential performance limitations within highly complex data environments. In blockchain data applications that support immutability and decentralization, ETE needs to be built carefully to guarantee compatibility with cryptographic hashing processes. Security stipulations must also be executed to minimize risks within blocks in the supply chain transaction data process. The block diagram of the ETE algorithm is illustrated in Figure 2. In order to maximize security, the proposed key scheduling strategy can rotate the TEA subkeys to every round function. Furthermore, the round function of TEA is enhanced by shifting keys prior to function integration.

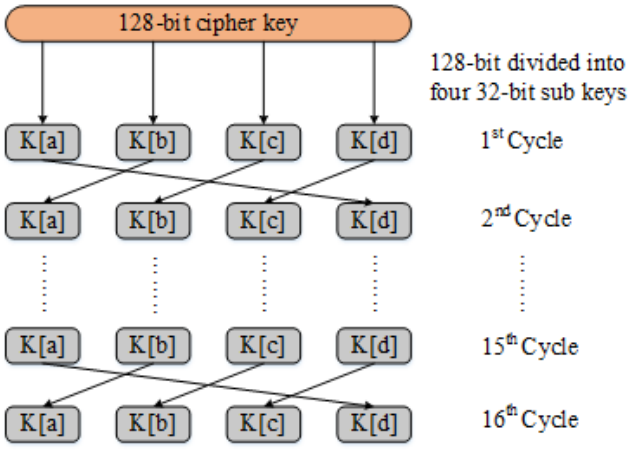

Figure 3: Structure of the new key scheduling

As mentioned earlier, the ETE applies the alteration to key handling and function rounds of TEA instead of utilizing additional methods for managing key scheduling to strengthen the security of TEA. The operations addition (ADD), exclusive OR (XOR), and shift operation (SHIFT) are examples of round functions. Here, the 128-bit encryption key can be divided into four subkeys, each of which rotates once. The key is created on each cycle, as opposed to previous TEA improvements, which saved the 32-64-bit key in an array that required additional memory. Every round function can use the value of every subkey in each cycle to generate a new key. The fundamental goal of ETE is to address the primary flaw of the original TEA, in which subkeys are assigned to a single round for the entire cycle, by spreading the subkeys across multiple rounds in each cycle. Figure 3 shows the structure of the new key

scheduling. This new key scheduling method has been integrated into TEA's enhanced function rounds. Each function round shifts the subkeys to raise the level of confusion and diffusion. Before adding to the delta, each half of the plaintext is shifted. The ETE was utilized to complete 16 cycles, as opposed to 32 cycles for the conventional TEA. Furthermore, TEA is chosen due to its lightweight design, low computational complexity and 128-bit key size, which make it suitable for high-throughput blockchain settings, compared to AES and Blowfish. TEA needs fewer resource and unlike DES, it is not limited by a short key length. An extended TEA combined with SHA-256 is used to enhance security and overcome known weaknesses.

3.3 Smart contract based blockchain technology

The blockchain is defined as a collection of append-only blocks, each containing numerous pieces of data that are stored by a peer-to-peer network that follows an inter-node communication protocol [31]. The consensus mechanism, or protocol, that validates new blocks contributes to the enchantment of blockchain. Through a consensus process, the majority of nodes will concur on the presence of each block's data once it has been verified. As the majority of the nodes on the blockchain will not admit it, altering the data on the blockchain is very difficult. Certain application scenarios are addressed by distinct kinds of blockchain systems, like permissionless as well as permissioned blockchains. The nature of blockchain technology allows for system decentralization and data immutability. These features suggest strategies to meet supply chain data management requirements. Since the data flow is managed and stored on the blockchain, it is originally immutable and can be trusted as a reliable proof of existence. Next, rather than being kept independently in different systems, the data from different supply chain participants can be connected to the blockchain system. This offers data exchange while saving time and cost on data retrieval.

3.3.1 Forks in Blockchain

In general, the blockchain follows certain rules, just like any other program. These rules are stated as "Protocols." The software must be updated on a regular basis for a variety of reasons, including greater throughput, performance enhancements, bug corrections, and more. A forked blockchain is a divergence of the blockchain ledger, forking the chain into two separate paths due to a change in the established protocol updates, security improvements, or governance. Forks can be hard forks, which permanently split the blockchain and require all nodes to upgrade, or soft forks, which follow the implications of backwards-compatible changes. Forked blockchains are typically perceived as a better alternative to conventional blockchain structures since they permit flexibility, scalability and enhancements in security. In other words, a forked blockchain allows developers to fix vulnerabilities in the protocol, improve consensus mechanisms, or add new functionality without disturbing the existing network. Furthermore, it can address the risks associated with an outdated protocol, guaranteeing that transactions remain efficient and secure.

Because of the flexibility of a forked blockchain, it can be employed in supply chain management, financial systems, and decentralized applications, where performance and security are crucial.

In terms of proposed blockchain-enabled supply chain management, a forked blockchain is a split in the blockchain ledger that creates two independent chains due to security upgrades, protocol updates, or governance decisions. A soft fork allows nodes with a backwards-compatible upgrade to validate the new transaction. In contrast, a hard fork creates a permanent split and requires all nodes to upgrade to the new protocol to maintain complete and valid copies of the blockchain. In another way, it is determined that the hard fork occurs when the updated new protocols are incompatible with the old protocols. Failure of the consensus mechanism can also lead to hard forks [32]. Nodes following the old protocol will reject the node if the new protocol submits a new block. Nodes following the old protocol may come to accept the new protocol when the quantity of blocks pushed by nodes following the new protocol rises. But if the nodes using the previous protocol disagree with the new one, the blockchain will be irreparably split into two chains of blocks, each with its own protocol. A soft fork happens when a new protocol is compatible with an existing protocol. As the number of blocks using the new protocol increases, nodes using the old protocol in soft forks gradually move to the new protocol.

A drawback of hard forks is that miners no longer receive fees for adding transactions to blocks. This is useful for a client who wants the transaction to go through. A hard fork is the only way to increase the size of transactions on a blockchain [33]. In addition, forking allows for novel measures of security where hash vulnerabilities related to hash-based attacks, such as collision attacks or preimage attacks, can be reduced with the implementation of forking. In hash-based attacks, the adversary attempts to manipulate

hash values to change transaction records. Due to the use of a forked blockchain, the proposed supply chain system offers better cryptographic hashing algorithms, consensus and improves validation protocols so that the threats and shortcomings from unstable vulnerabilities can be eliminated. Also, the forked blockchains provide mechanisms of adaptive security so that organizations can quickly respond to new threats by updating the blockchain's structure without compromising the integrity of data. This supports immutability, transparency, as well as traceability and makes supply chain management more resilient against cyber threats while upholding trust among stakeholders.

3.3.2 Smart contract

Smart contracts frequently refer to programs (or scripts) that are translated into high-level computer languages [34]. These scripts are kept on every node in the blockchain-based network. In particular, smart contracts indicate whether or not a transaction between nodes will follow predetermined rules (scripts). When a transaction does not adhere to the rules (scripts), the system will produce an error message stating that "transactions cannot be ended". Otherwise, the node can carry out transactions. Smart contracts are commonly influenced by state-machine techniques, which use a predictable method. A directed graph is used to characterize the deterministic state machine policy, with each vertex (or node) representing a machine state and each edge revealing how the machine transitions from one state to another.

The proposed work's state machine-based smart contract has six states: State 0, State I, State II, State III, State IV, and State V, as seen in Figure 4. These dissimilar states represent different entities. State 0 indicates the manufacturer's entity, state I depicts the warehouse, state II the wholesaler, state III the retailer, state IV the end user, and state V the dead state. Furthermore, the suggested method expresses numerous measures. The actions realize the transition from one state to another. Actions contain demand, supply, purchase, delivery, no action and violation. For instance, if the action is a violation and the machine is in state III, the machine changes from state III to state V and shows an error message. When the action is a delivery, and the machine is in state I, it inevitably shifts from state I to state II. When the action is entreated, and the machine is in state IV, the system transitions from state IV to state III, as well as so on. Furthermore, ownership of the asset can be altered based on the machine's condition. The pseudocode for the smart contract mechanism is illustrated in Table 2.

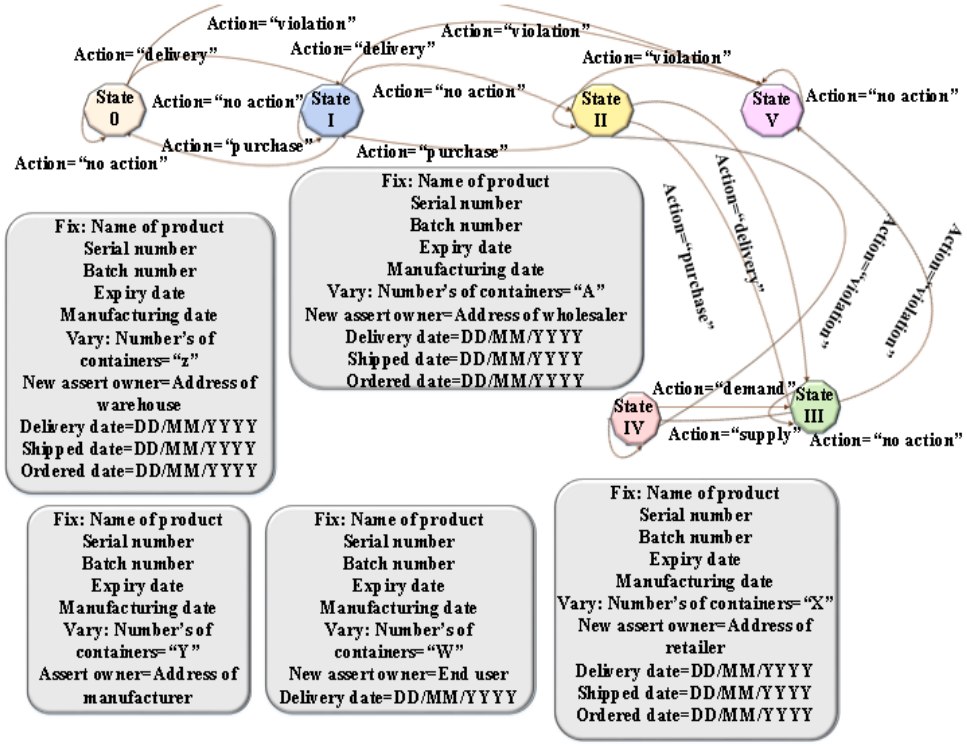

Figure 4: State machine model smart enhance

Table 2: Pseudocode of smart contract

Algorithm 1: Pseudocode of smart contract
Input: Actions like delivery, violation, demand, no action, supply, and purchase
Output: Messages like no action, error and abort, and ownership of product

```
If (g(ACtion) == DElivery) then
  If (delivery _ date < expiry _ date) then
    If (predefined {delivery _ date < time _ delivery) then
       Distribute the product to other entity
       Alter the product ownership
    Else
       Report a message "Product not supplied"
    End if
  Else
       Send an error message "Product expired"
  End if
End if
If (g(ACtion) == SUpply) then
      If (PRoduct = AVailable) then
            Product can be supplied to end user
            Alter the product ownership
         Else
            Report a message "No product availability"
      End if
End if
If (g(ACtion) == PUrchase) then
  If (PRoduct = Available) then
    Function (ACtion) = DElivery ;
   Else
      Report a message "No product availability"
   End if
End if
If (g(ACtion) == DEmand) then
   If (PRoduct = AVailable) then
     If (expiry _ date > delivery _ date) then
        Function (action) = sup ply ;
      Else
       Report a message "Product expired"
     End if
    Else
       Report a message "No product availability"
    End if
End if
If (g(ACtion) == No _ ACtion) then
     Remain the entity in the same state
End if
If (g(ACtion) == VIolation) then
    Report error message "Executed wrong operation"
End if
```

3.4 Verification based on the consensus algorithm

In the proposed method, an Up-DPoS based consensus procedure is utilized to validate blockchain transactions. Delegated proof of stake (DPoS) is one of the consensus techniques in which the participants choose a small number of delegates

to approve transactions as well as secure the network. These delegates are responsible for block creation and validation. Stakeholder voting is used to replace dishonest or ineffective delegates following regular performance evaluations. A DPoS voting architecture was proposed in [35] to drastically reduce processing time and computation power consumption for producing new blocks. In the DPoS, validators are chosen using a variety of procedures or reputation scores. To reach consensus, DPoS employing blockchains relies on reputation-based voting procedures. The size of a user's assets determines their voting power.

Users with larger holdings have more say over who is preferred by the nodes. These chosen nodes are referred to as delegates. The classic DPoS approach allows each node in the blockchain network to vote based on stakes and then choose its own approved node. Comparable systems are less democratic than DPoS since they use a decentralized voting procedure. DPoS has a system in place to ensure that people are trusted to sign blocks on behalf of the network and do so impartially as well as correctly rather than ignoring the requirement for trust. Additionally, every signed block must prove that it was generated from a trusted node. With DPoS, a transaction is no longer required to be confirmed until a predetermined number of untrusted nodes have been confirmed. Besides, it permits more transactions to be encompassed in a block in contrast to proof of stake (PoS) or proof of work (PoW). Even if DPoS still faces security issues and less decentralization than PoW [36]. However, it commonly supports offering faster processing-based transactions, with a turnaround time of about 3 s.

Users vote in DPoS to select a set of delegates who will generate the blocks. Users select the cluster of delegates using a variety of approaches and reputation scores. Delegates act independently when proposing new blocks. In each round, a leader is chosen from among the delegates capable of generating the block in that network. The matching technique is used to determine who will be the leader or forerunner. In the event that the forerunners misbehave, they are removed from the validator cluster; otherwise, they are rewarded for creating a new block. Each representative competes with the others for admission in the validation cluster. In this instance, voters who plan to vote for each validator may get a variety of incentives. For instance, if a representative is elected for a block proposal, they can share a portion of the reward value among the selected users. It is generally assumed that if the number of validators is small, the consensus will occur quickly.

Up-DPoS is an advanced version of DPoS meant to improve transaction efficiency and scalability while ensuring robust security, which is mainly significant for blockchain-based supply chain management. The classical DPoS uses elected delegates to confirm transactions and ensure consensus; however, Up-DPoS employs a dual-layer architecture that streamlines transaction processing. Layer 1 (L1) is accountable for final settlement, and Layer 2 (L2) processes high-frequency transactions at lower fees prior to aggregating and settling them in L1. This segregation delivers a balance between efficiency and security, decreasing network congestion and gas costs. Up-DPoS also improves DPoS by integrating Super Block Producers (Super BPs), a group of elected validators who sign off on transactions with better credibility, providing strong security.

In contrast to typical DPoS, Up-DPoS incorporates Practical Byzantine Fault Tolerance (pBFT) to overcome fraudulent transactions and double-spending, a critical factor for supply chain management where the integrity of transactions is essential. Furthermore, Master-Slave Architecture delivers redundancy, storing transaction copies to prevent potential failures. When it comes to performance, Up-DPoS highly accelerates Transactions Per Second (TPS) for fast exchanges of data across supply chains. This improvement benefits supply chain operations directly by reducing delays, lowering operational expenses, and permitting real-time monitoring of goods and shipments. In contrast to conventional DPoS, where transaction finality relies on delegate consensus alone, Up-DPoS necessitates validation from a minimum of three out of four Super BPs before a transaction is confirmed, improving security. Accordingly, Up-DPoS-driven supply chains with blockchain technology can undergo greater efficiency, transparency, and reliability, building a scalable solution for industries that demand cost-efficient and secure transaction management.

3.4.1 Up-DPoS for consensus

This section discusses a modified DPoS schema known as Up-DPoS, which aims to reduce transaction costs further while increasing transaction efficiency per second for DPoS. By modifying the present DPoS consensus, Up-DPoS allows the blockchain to build a Layer 2 network on top. Additionally, the concept of Super BP is introduced. For processing, the user can send it to the mainnet or L2. The cost of an L2 transaction may be lower than a

mainnet transaction. Up-DPoS is considered the name of a transaction that is sent directly to the mainnet. The consumer will pay more because it achieves finality faster than L2. On the mainnet, DPoS might be employed, but block producers would need to stake additional tokens in order to validate a transaction. Furthermore, the incentives for voting and validating ought to be higher than those on the L2 network. The user would have to pay more to send the transaction directly to L1, and even though the L2 network would still be in charge of DPoS consensus, the transaction could not be confirmed. All L2 transactions have the potential to join the mainnet in the end. This would occur every time, or the mainnet throughput would drastically decrease after 24 hours (limits to be established). The illustration of the Up-DPoS transaction is provided in Figure 5.

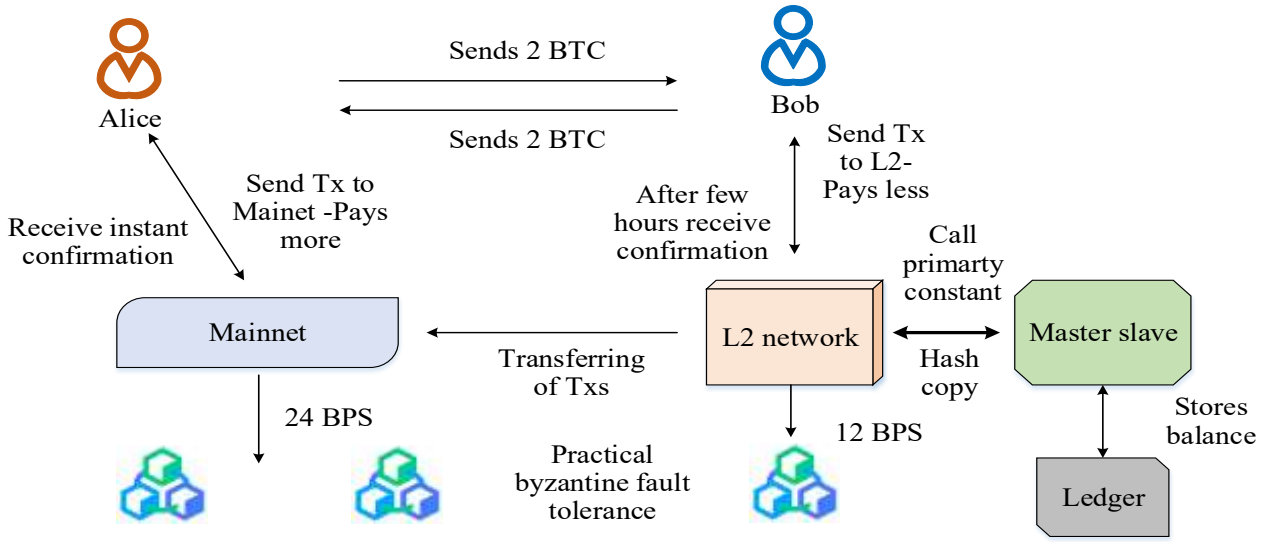

Figure 5: Illustration of Up-DPoS transaction

Meanwhile, L2 block producers are unable to partake in mainnet activities, and vice versa. For validation, the reward will be less because L2 would have cheaper gas prices. L2 TPS on the mainnet is expected to reach around 60,000, with a transaction rate of nearly 20,000 [37]. The number of block creators may vary among mainnet and L2 transactions, with 24 on the mainnet and 12 on the L2. This consensus mechanism can improve network transaction performance by reducing congestion on the mainnet. Additionally, it allows users to profit from blockchain technology at a low cost. Customers can pay according to the urgency and use case under this structure. When they proceed from L2 to mainnet, customers can track their transactions and receive updates. When the user requires the transaction to be validated quickly, an additional cost will be charged; otherwise, the transaction will move immediately to the network as well as accept a timely confirmation. If a transaction from L2 to mainnet fails or is not completed, the platform will store the hash in an external database.

To be a BP on the mainnet, one must have a stake for a set period of time and possess a specific level of processing power. The BPs would be placed in the election pool and subject to hourly voting by community members once the allowed period had passed. A total of 24 block producers (BPs) would exist, with the top four designated as Super BPs. Furthermore, to accomplish finality, a minimum of 20 BPs must validate a transaction or the transaction must be signed and validated by at least 3 of 4 super BPs. This maximizes the network's security and efficiency. Elections are held after each hour to determine the next BP for the following one-hour period. The identical system has been applied to L2, with the distinction that there can be 10 BPs rather than 5, and the concept of Super BP has been discarded. The election is held every hour, and the number of tokens that BP must stake decreases with time. Furthermore, the voting and mainnet vote rewards are similar.

In Up-DPoS, each user on the network can get a reward. The compensation of both BPs and Super BPs will depend on the number of blocks they validate in an hour. The voters will also take the voting prizes. Both deferred payment transactions and high-frequency instantaneous transactions can be utilized over this network. The platform incorporates Master-Slave architecture and is equipped with practical Byzantine fault tolerance to protect the network against double-spending and

malicious transactions. When a transaction is refused from the L2 to the mainnet, a copy of the transaction is stored. The sequential procedures of Up-DPoS are provided below as follows:

➢ ***Consensus node selection***

- There are two layers in this specific blockchain, namely, Layer 1 and Layer 2.
- Layer 1 is used for the final settlement (the transaction will reach finality).
- Layers 1 and 2 rely on delegated proof of mechanism.
- There are 24 block producers in Layer 1, with each chosen by the community as well as from the community.
- There are 12 block producers in Layer 2, with everyone contributing to the community.
- In order to avert double-spending and malicious transactions, each layer is secure.

➢ ***Nomination procedure of block producer***

- The members of the community are used to nominate the block producers for Layer1 by-elections.
- For Layer 1, the block producers are eligible to be nominated as block producers after being a block producer on the layer for two weeks, as well as contributing to the chain for at least two weeks.
- The producers should also stake a specific amount of coins to participate in the block producer nomination process for layers.
- The community's participation defines the nomination of block producers for Layer 2.
- The compensation to block producers of Layer 1 depends on the number of transactions they authorize.

➢ ***Transaction journey***

- Users can choose between Layer 1 and Layer 2 based on where they need to send the transaction.
- Users who desire to send directly to Layer 1 can attain transaction finality within a few seconds; however, they could have to pay a higher charge.
- Users who desire to send through Layer 2 have less fess; however, their transaction finality will become in ($T+1$) time.
- Every 12 hours, the Layer 2 transactions are bundled and forwarded to Layer 1.
- This bundled transaction is designed to require fewer fees in order to accomplish finality.

3.5 Security Analysis

This section examines common security threats in peer-to-peer blockchain networks and their influence on supply chain operations. Such attacks aim to interrupt node-to-node communication, compromise data confidentiality, integrity and availability. It is essential to consider these threats and design robust security mechanisms to reduce their impact.

3.5.1 Sybil Attacks

A Sybil attack is a critical threat in blockchain-based peer-to-peer networks, where an Attacker generates and controls numerous fake identities to influence the network. By using these fraudulent nodes, the attacker can spread incorrect information and manipulate network behavior, thereby compromising data confidentiality and integrity. The effectiveness of this attack depends on weaknesses such as low identity creation costs, insufficient trust mechanisms between nodes and insecure private key management. Although completely preventing Sybil attacks is difficult, their impact can be reduced through robust authentication and security mechanisms. The proposed work mitigates Sybil attacks by employing the Up-DPoS consensus mechanism, which restricts block validation to verify and reputation-based nodes, making the creation of fake identities ineffective.

3.5.2 Eclipse Attack

An eclipse attack occurs when an attacker uses multiple coordinated identities to isolate legitimate nodes by taking control of their network connections, typically targeting the victim's public IP address. Due to the decentralized nature of peer-to-peer networks, such attacks are more complex and harder to detect than those in traditional client-server systems. By dominating a node's neighboring connections, attackers can manipulate transactions, restrict communications with the rest of the network as well as distort the victim's view of the blockchain. This isolation can also lead to Denial of Service (DoS) attacks. While complete prevention is challenging, limiting node access to trusted and authorized participants can significantly reduce the attack's impact. The proposed framework reduces the risk of Eclipse attacks by limiting peer connections through access-controlled and authorized nodes within the forked blockchain architecture.

4. RESULTS AND DISCUSSION

This section deliberates on the implementation results of the suggested method for logistic supply

chain management. In the supply chain network, the proposed approach has been employed by several organizations to complete express delivery business, analyze algorithm performance, and decentralize the security of logistical transactions in the system using Up-DPoS. The proposed solution for successful supply chain management is simulated in a standalone computer using the Python tool. It is configured with an Intel(R) Core(TM) i5-9500 CPU @ 3.00 GHz processor, 15.00 GB (15.8 GB usable) of main memory, as well as a 64-bit Windows 10 operating system. Table 3 illustrates the parameter details.

Table 3: Parameters

Category	Parameters	Values
Blockchain Platform	Framework	Hyperledger Fabric
Network Topology	Number of nodes	10 distributed nodes
	Peer nodes	5 peers executing smart contracts and maintaining the ledger
	Ordering Nodes	3 nodes managing transaction ordering and block generation
	Client Nodes	2 nodes initiating transaction ordering and block generation
Channel setup	Channel configuration	Single channel for shared ledger access
Consensus Mechanism	Consensus Model	Upgraded Delegated Proof of Stake (Up-DPoS)
	Block Validation	Delegate nodes validate and confirm blocks.
Security Configuration	Hash Algorithm	SHA-256 for data integrity
	Hash length	256 bits
	Encryption Method	Extended Tiny Encryption (ETE)
	Key Size (ETE)	128 bits
Transaction settings	Transaction Block Size	1 MB
	Transaction Rate	50 transactions per second (tps)
Network Constraints	Latency Tolerance	300 ms delay

The blockchain network and cryptographic setup utilized in the ECBM-CE framework are presented in depth in Capability. These guidelines guarantee safe, impenetrable, and repeatable transaction processing in the pharmaceutical supply chain.

The proposed framework does not rely externally on sourced public datasets; instead, it captures transactional and operational data produced during supply chain activities executed through smart contracts on blockchain platforms such as Hyperledger Fabric and Ethereum. Each participating entity in the supply chain, including suppliers, manufacturers, distributors, retailers, as well as customers, interacts with the system by initiating transactions. These transactions generate structured textual data comprising product identifiers, batch numbers, quantities, timestamps, sender and receiver details, shipment status, and other logistics-related information. All collected data are automatically recorded as blockchain transactions and stored in a distributed and immutable ledger. Data collection occurs in real time through application programming interfaces (APIs) implemented using Node.js, where user actions such as data entry, updates, and retrieval trigger transaction creation. Before storage, each transaction undergoes pre-processing that includes hashing and encryption to ensure integrity, confidentiality and authenticity. Hash validation mechanisms are applied to verify both incoming and outgoing data blocks, thereby preventing unauthorized or spam data from being recorded on the blockchain. This internally generated dataset enables controlled evaluation of system performance metrics includes execution time, throughput, latency as well as security effectiveness under varying transaction volumes.

4.1 Performance metrics

In this section, various performance indicators utilized to evaluate the efficacy of the suggested supply chain management system are discussed. The proposed method utilized a variety of performance metrics, including throughput, network latency, encryption time, decryption time, block generation time, as well as execution time. Throughput is defined as the total number of logistic service transactions completed by many consensus nodes per unit of time. It is mathematically stated as follows.

$$Q = \frac{\sum_{j=1}^{q} Tra \times q}{ET} \tag{10}$$

where, Q characterizes the size of throughput, Tra designates the transactions, q suggests the number of transactions, and ET states the execution time.

The network latency is mostly determined by the time that the logistics blockchain is out of blocks in a similar environment, i.e., the time that multiple logistical services have been completed. It is statistically specified as follows.

$$\sum_{j=1}^{q} V = \sum_{j=1} (v_0 + v_1 + v_2) \times q \tag{11}$$

where, V designates the total network latency, v_0 represents the broadcast time of logistics blockchain, v_1 implies the execution time of consensus algorithm, and v_2 resembles the conformation time of logistic blockchain.

The total time required to accomplish a given number of transactions is revealed as the execution time. The following expression is utilized to compute the execution time:

$$Time\ of\ execution = TT_{\sin gle} \times number\ of\ transactions \tag{12}$$

where, $TT_{\sin gle}$ designates the time needed to process a single transaction, covering hashing, encryption, decryption, blockchain validation and storage operations.

The time taken by the encryption algorithm to encrypt hashed logistic information is designated as encryption time. In other terms, the variance among the start time as well as the end time of encryption is considered as the encryption time. The time taken by the encryption algorithm to decrypt the hashed logistic information is deliberated as the decryption time.

4.2 Performance evaluation

The proposed method's simulation results are assessed in this subsection. The outcomes of the suggested approach are compared to those of the most popular prevailing consensus mechanisms, inludes PoS, PoW and DPoS [38], as well as the cryptographic algorithms like TEA, blowfish, data encryption standard (DES) and advanced encryption standard (AES). The consensus mechanisms are assessed from the perspective of blockchain-based supply chain management to determine their resistance to hash-based attacks in the form of collision and preimage that could threaten data authenticity and traceability. PoS is known for comparison, as it enhances energy effectiveness and relies less on mining hardware, which makes it appropriate for sustainable blockchain solutions. PoW has been selected because of its well-established security and decentralized technique, where computational challenges provide resistance to adversarial attacks. Meanwhile, DPoS has been chosen for its better transaction speed and scalability, permitting faster validation and network efficiency by delegating transactions confirmed to elected representatives. These mechanisms are tested for their effect on scalability, network security, and resistance against hash-based attacks, which are indispensable for guaranteeing data integrity and traceability throughout complex supply chain networks. Moreover, cryptographic algorithms like TEA, Blowfish, DES, and AES are chosen for comparative evaluation because they are commonly employed to secure data transmission. These algorithms differ in computational overhead, encryption strength, and resistance to cryptanalytic attacks. Comparing the schemes provides an overall

evaluation of their aptness in providing protection against hash-based attacks like replay, preimage, and collision attacks. The study compares approaches to identify compromises in security, efficiency, and scalability. It provides a baseline for designing an efficient approach to protect supply chain data from hash-based threats.

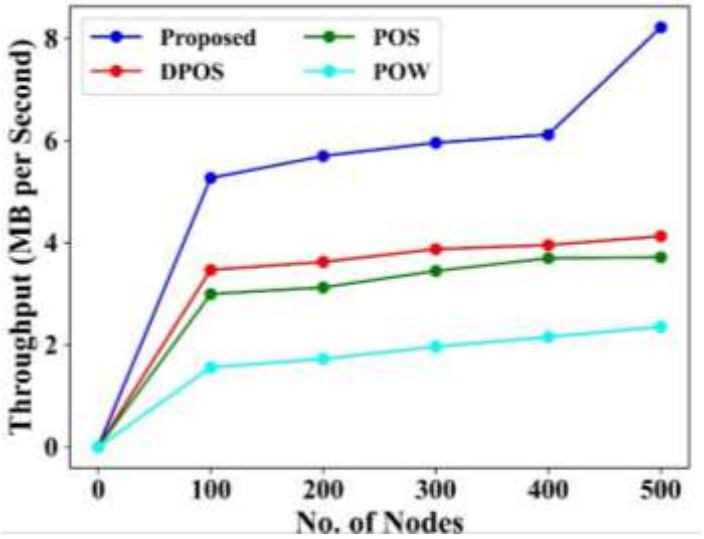

Figure 6: Comparison of throughput with proposed and existing methods

The throughput performance for a varying number of nodes from 100 to 500 is displayed in Figure 6. From the observed data, it is clear that the throughput is greatly impacted by the proposed method. The transactions per second (tps) measure is used to examine throughput performance. By altering the number of nodes, the suggested approach is compared with the current algorithms, comprising PoS, PoW and DPoS algorithms [38]. The throughput of the proposed method is 8.2165 tps when the number of nodes is set to 500. A system needs to have a higher throughput to be efficient.

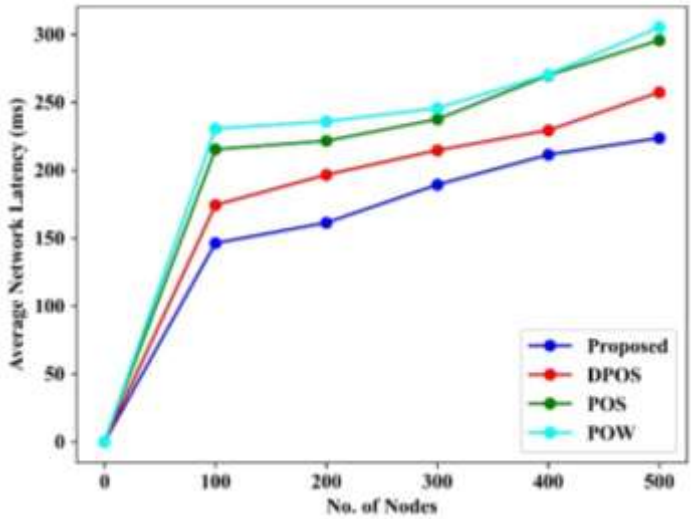

Figure 7: Comparison of network latency with proposed and existing methods

The comparison of average network latency with proposed and modern techniques is exhibited in Figure 7. The graphical representation demonstrates that the proposed strategy significantly reduces latency. Latency measures are quantified in milliseconds (ms). By varying the number of nodes to 100, 200, 300, 400, and 500, the proposed method is compared to current approaches. The graph shows that when there are 500 nodes, the POW takes 305.24 ms, the PoS takes 295.64 ms, and the DPoS takes 257.35 ms. However, when compared to the existing approaches, the proposed method achieved a minimum latency of 223.65 ms for 500 nodes. For the system to be effective, there must be little latency.

The current methods take longer to execute and handle data transfers incorrectly, resulting in increased delay. As a result, it is demonstrated that the proposed strategy minimizes latency. A comparison of throughput and network latency using proposed and current approaches is offered in Table 4.

Table 4: Analysis of throughput and network latency with existing methods

Metrics	Methods	Number of nodes				
		100	**200**	**300**	**400**	**500**
Throughput (tps)	PoW	1.5563	1.7245	1.9635	2.1535	2.3524
	PoS	2.9935	3.1245	3.4456	3.6966	3.7135
	DPoS	3.4666	3.6236	3.8736	3.9536	4.1255
	Proposed	**5.2665**	**5.6965**	**5.9563**	**6.1165**	**8.2165**
Network latency (ms)	PoW	230.54	235.85	245.65	270.36	305.24
	PoS	215.35	221.54	237.48	269.78	295.64
	DPoS	174.36	196.63	214.65	229.36	257.35
	Proposed	**146.29**	**161.32**	**189.29**	**211.36**	**223.65**

Encryption and decryption times become indispensable performance metrics for every security solution. Based on the time performance, the encryption time is assessed and offered. Encryption time typically measures how long it takes to encrypt data or change plain text into ciphertext. Figure 8 displays the encryption time performance for the proposed method using prevailing algorithms of TEA, blowfish, DES as well as AES. The minimum encryption time taken by the proposed method is 11.6332 ms, whereas the use of prevailing AES, ordinary TEA, DES, and blowfish upsurges the overhead of processing and communication. The higher encryption time taken by prevailing AES, ordinary TEA, DES, and blowfish is 22.5634 ms, 12.3341 ms, 18.6542 ms and 13.5689 ms. As a result, it is revealed that the encryption time of the suggested approach is much shorter than that of the prevailing techniques.

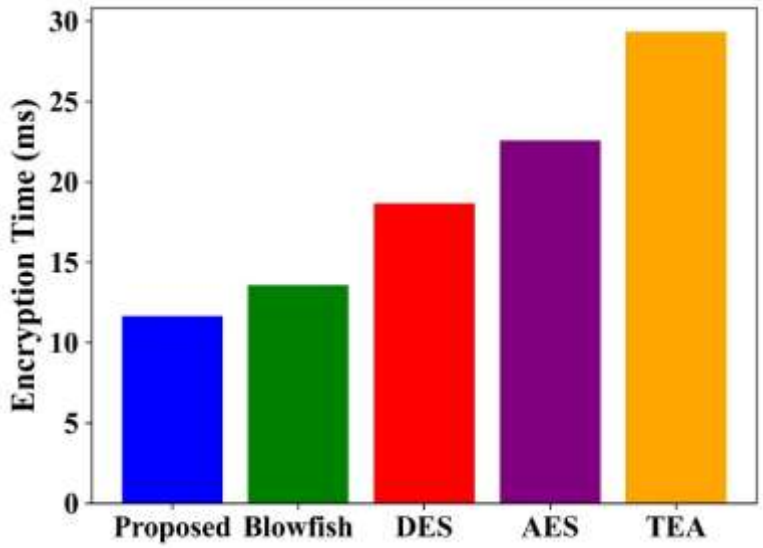

Figure 8: Comparison of encryption time with proposed and existing methods

On the other hand, the decryption time determines the amount of time required to finish data decryption or the reverse of encryption. The comparison of the decryption times proposed with the current AES, ordinary TEA, DES, and Blowfish algorithms is shown in Figure 9. The graphical demonstration clearly showed that the suggested method takes the least amount of time to decrypt the data when compared to other algorithms. The minimum decryption time taken by the suggested method is 16.3544 ms, whereas the prevailing AES, ordinary TEA, DES, and Blowfish have obtained higher decryption times of 25.6235 ms, 17.3654 ms, 21.6344 ms, and 18.6364 ms.

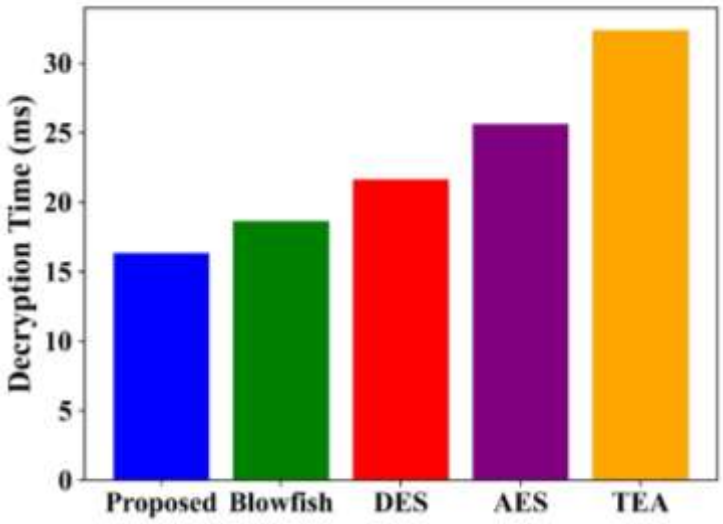

Figure 9: Comparison of decryption time with proposed and existing methods

The performance comparison of overall execution time for various numbers of transaction records is demonstrated in Figure 10. The execution time is maximized in tandem with the number of transactions. On the other hand, the graphical illustration shows a significant reduction in overall execution time for 100, 200, 300, 400, and 500 transactions. The proposed technique executes in 18.8067 seconds for 100 transactions and 86.2456 seconds for 500 transactions. Overall, the smart contract-based blockchain performs better due to its lower execution time. A comparison of execution time using the proposed and current approaches is presented in Table 5.

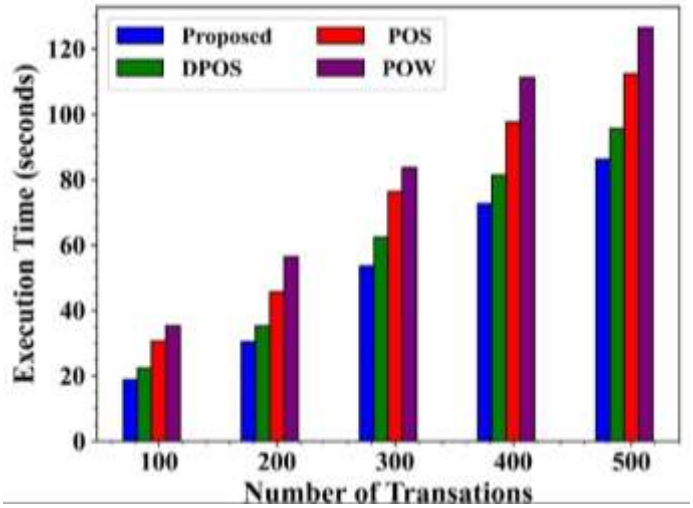

Figure 10: Comparison of execution time with proposed and existing methods

Table 5: Analysis of execution time with existing methods

Methods	Number of transactions (seconds)				
	100	**200**	**300**	**400**	**500**
PoW	35.3655	56.3654	83.6354	111.36	126.544
PoS	30.6954	45.6354	76.3654	97.6354	112.365
DPoS	22.3654	35.2448	62.3655	81.3658	95.5544
Proposed	**18.8067**	**30.3664**	**53.6544**	**72.6645**	**86.2456**

In the context of supply chain management, evaluating block creation time while adjusting the number of blocks is critical for improving system security, transparency, and efficiency. Figure 11 compares block creation times for 200, 400, 600, 800, and 1000 blocks. Blocks are generated to store

transaction-related data. The suggested method allows the smart contract-based blockchain to record and process transactions more quickly by speeding up block generation and reducing block generation time. The proposed method has achieved minimum block generation times of 6.4432 s, 7.1325 s, 8.2563 s, 9.4498 s, and 10.2365 s for 200, 400, 600, 800, and 1000 blocks, respectively. This is especially useful when tracking high-value assets or perishable items, where timely data is required. Because block formation takes such little time, it also helps to speed up execution, which significantly improves system efficiency.

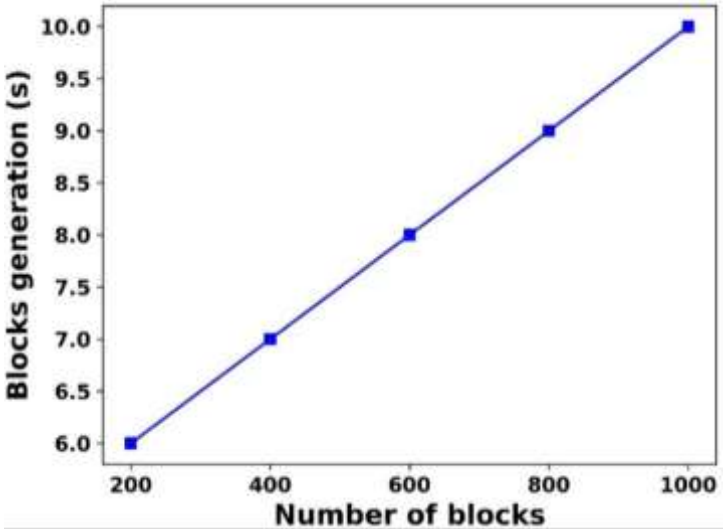

Figure 11: Comparison of execution time with proposed and existing methods

Transaction time is considered the overall amount of time needed to finish a given number of transactions. In the proposed work, the execution time for processing 100 transactions is emphasized. Table 4 compares the performance of the proposed and existing algorithms in terms of overall execution time for different numbers of transaction records. As expected, execution time upsurges with the number of transactions. However, as the number of transactions rises, the graphical representation shows that the rate of increase in execution time decreases. In particular, when processing 20, 40, 69, 80, and 100 transactions, the proposed method takes 2 2.32 s, 4.12 s, 5.03 s, 6.32 s and 9.76 s to execute. In contrast, the existing methods have a longer execution time. The decrease in the marginal increase in execution time for the proposed method indicates better scalability, even though the overall execution time stays higher for increasing transaction counts. Comparative analysis in terms of execution time is given in Table 6.

Table 6: Comparative analysis in terms of execution time

Number of transactions	PoS	PoW	DPoS	Proposed
20	3.86 s	3.53 s	2.98 s	2.32 s
40	4.45 s	4.98 s	4.57 s	4.12 s
60	6.73 s	6.35 s	5.78 s	5.03 s
80	8.26 s	7.42 s	6.83 s	6.32 s
100	10.96 s	10.75 s	10.03 s	9.76 s

Additionally, the performance comparison is made in the proposed model for the security metric, attack resistance, in order to assess the capability to resist various attacks. Table 5 presents the comparative analysis of attack resistance for the proposed and existing methods against different cyber threats. This assessment is significant in determining how robust each method is in mitigating attacks like collision, preimage, denial-of-service (DoS), distributed DoS (DDoS), and information leakage. In the proposed work, the blockchain-based smart contract model combines an advanced encryption method and an improved consensus approach to address hash-based vulnerabilities, guaranteeing data integrity and secure transactions across the supply chain. Due to the computational complexity of SHA-256, the

proposed method provides powerful resistance against both preimage and collision attacks. The usage of ETE allows for secure and lightweight communication. This significantly decreases the potential for information leakage from input data. The forked blockchain design, paired with smart contract-based validation, aids in the prevention of DoS/DDoS attacks by distributing trust and successfully isolating suspicious nodes. In comparison to traditional PoS, PoW and DPoS, which can hardly handle selective attacks, the proposed method guarantees robustness against all major attacks, thus improving the integrity, confidentiality and availability of supply chain data in blockchain ecosystems. Comparative analysis of attack resistance against different cyber threats is given in Table 7.

Table 7: Comparative analysis of attack resistance against different cyber threads

Methods	**Attacks**				
	Collision	**DoS**	**Preimage**	**DDoS**	**Information leakage**
PoS	No	Yes	No	Yes	No
PoW	Yes	No	No	No	No
DPoS	Yes	No	No	No	Yes
Proposed	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

4.3 Rationale and Implementation of the Work

The Tiny encryption algorithm is selected in this work due to its lightweight structure, low computational complexity and minimal memoryneeds, which are important for blockchain-enabled pharmaceutical supply chain systems where fast transaction processing and resource efficiency are critical. Compared with conventional encryption schemes such as AES and RSE, TEA imposes lower computational overhead, importantly, making it more appropriate for decentralized blockchain settings. Furthermore, traditional TEA is susceptible to related-key and equivalent-key attacks. To mitigate these challenges, an Extended Tiny Encryption method is developed by improving the key-mixing process and incorporating SHA-256-generated hash values prior to encryption, therefore enhancing resistance to hash-based and cryptographic attacks. The integration of ETE within the proposed EBCM-CE approach, along with smart contract-based sorted blockchain structure and the Upgraded Delgated Proof of Stack (Up-DPoS) consensus mechanism, assists secure data storage, efficient transaction validation and reliable tracking of pharmaceutical goods. The implementation of this approach includes improved security, higher transaction throughput, reduced network latency and a scalable solution appropriate for real-world pharmaceutical supply chain management systems.

4.4 Discussion

As mentioned in the earlier section, a supply chain is considered the collection of all the activities and interrelated information flows engaged in transferring products or services from a seller to a consumer.

The management of activities and information concerning sourcing, conversion, procurement, and all logistics is a constituent of supply chain management. Supply chain management offers numerous environmental, financial, and social benefits, such as a shorter order-to-delivery cycle, better resource usage, and early problem detection. Conceptual research was conducted using a few existing blockchain technology solutions to improve supply chain management. Blockchain use is frequently related to supply chain data management. The blockchain system's input is represented in supply chain management information. The initial step in data management, especially in the supply chain, is data collection. During the methodical examination of blockchain's applicability in supply chain management, multiple conceptual and deployed systems were created for various supply chain contexts. Because the food supply chain can resolve issues about food safety and origins, it is widely explored. Several ongoing efforts, such as Provenance and FarmShare, are focused on food traceability and the supply chain. Blockchain technology is exceptionally attractive to the drug and pharmaceutical industries because healthcare is also a substantial social issue.

Nevertheless, the majority of existing research is carried out from the perspectives of economics, logistics, and management, specifically if it comes to studies relating to many disciplines. If the blockchain is used for supply chain management, the benefits of introducing the blockchain are admitted. However, the complications still endure in the practice of existing blockchain's immaturity and deployment costs are argued. In the interim, very few research has

addressed the challenges that arise from the technological architecture of the blockchain system, particularly the privacy and security problems. Additionally, the hash-based threats pose serious risks to blockchain-enabled supply chains by targeting the integrity as well as security of the information being stored. These threats take advantage of weaknesses that are created by hash functions, such as collision vulnerabilities. With hash collisions, the same hash output can result from two different inputs, allowing malicious actors to use the hash to circumvent previously stored supply chain records. Attacks can also involve preimage attacks where the attacker defeats the confidentiality and reverses the hash back to the original data. For example, in pharmaceutical supply chains, an attacker could change the shipment records of drugs for a shipment of counterfeit drugs using hash collisions, which could then continue through the system. To address these issues, a proposed solution exists where an ETE is used with a SHA-256 hash, thus allowing for more security. SHA-256 allows for unique hash values, given the current capabilities of technology, making collision attacks relatively difficult to execute. In case the hash is broken, the ETE ensures that the hashed data cannot be retrieved without authorized access. The Up-DPoS consensus algorithm authenticates the transaction on the chain, guaranteeing that only legitimate data is recorded on the blockchain. It achieves an acceptable level of trust that a transaction represented by a smart contract can be executed to ensure supply chain security. The proposed method upholds traceability and transparency and minimizes the chance of fraudulent activities by using smart-contract-based forked blockchain technology. Overall, these elements of a unique SHA-256 hash algorithm, ETE, and transaction validity using Up-DPoS consensus in a smart contract-based forked blockchain have greatly improved a blockchain-enabled supply chain's resistance to hash-based attacks, as well as provided security and data integrity. The experimental results demonstrate that ETE reduces encryption overhead, leading to higher throughput and lower latency compared to AES, Blowfich and DES. This confirms that TAE provides a better security performance trade-off for real-time pharmaceutical supply chain applications. Comparative analysis of the proposed model with the state-of-the-art methods is given in Table 8.

Table 8: Comparative analysis of the proposed model with the state-of-the-art methods

Author name & References	Methods	Performance
Piera Centrobelli et al. [21]	Integrated Triple Retry framework	The performance analysis on trust, traceability and transparency
Nafisa Anjum and Pramit Dutta, [22]	DApp	Security, privacy, Decentralization, and transparency for
Lufei Huang et al. [23]	AHP and DEMATEL method	Data security, technological feasibility, and cost control
Pedro Azevedo et al. [24]	Ethereum Smart Contract, as well as a PKI-based certificate authentication system	Provenance, traceability and chain of custody
Pratyush Kumar Patro et al. [25]	A private Ethereum blockchain-based solution	Transparency, Decentralized, Traceability, Availability, Data integrity and so on
Sholapurapu et al. [27]	CH-MFA	CH-MFA authentication capabilities through its superior scalability and reliability factors while upholding operational efficiency
Curado et al. [28]	ZKPs	Improve security and privacy in the maritime supply chain, achieving transparency.
Proposed method	**ECBM-CE**	ECBM-CE achieves a higher throughput of 8.2165 transactions

		per second and a reduced network latency of 223.65 ms.

5. CONCLUSION

This paper contributes a novel ECBM-CE method to provide a robust solution for improving the security of forked blockchain in supply chain management. This method uses logistic information to guarantee the integrity as well as privacy of supply chain transactions by implementing key procedures such as data collection, SHA-256 hash value generation, encryption using the new ETE, and safe blockchain storage.

The supply chain is protected from potential security threats by utilizing an Up-DPoS consensus algorithm and a blockchain based on smart contracts, which improve the validation process and reduce hash-based assaults. To assess the efficiency of the ECBM-CE approach, various evaluation criteria are used, including throughput, network latency, encryption time, decryption time, block creation time, as well as execution time. As a result, the performance of the ECBM-CE method is compared to other recently developed algorithms by altering the number of nodes and transactions in the network. Overall, the ECBM-CE technique has demonstrated significant performance and boosted security in supply chain management systems by integrating cryptographic encryption as well as an improved consensus mechanism. Nevertheless, certain limitations need to be addressed. Even if the SHA-256 hashing algorithm provides high data integrity, its computational complexity can result in longer processing time, which may influence system scalability. Besides, the efficacy of the ETE strategy in resisting sophisticated cryptographic attacks necessitates further validation. The use of the Up-DPoS consensus mechanism introduces potential vulnerabilities related to delegate choice and centralization threats. Future research can focus on optimizing encryption efficiency, investigating alternative consensus mechanisms for further decentralizing validation processes, and evaluating the method's strength against emerging cyber threats. Furthermore, expanding the use of the ECBM-CE model to other areas beyond the pharmaceutical industry may yield useful information regarding its flexibility and performance in various supply chain environments.

REFERENCES

[1] A. Rizwan, D. A. Karras, J. Kumar, M. Sánchez-Chero, M. M. Mogollón Taboada, & G. C. Altamirano, An internet of things (IoT) based block chain technology to enhance the quality of supply chain management (SCM), Mathematical Problems in Engineering. 2022(1) (2022) 9679050.

[2] T.Dursun, F. Birinci, B. Alptekin, I. Sertkaya, O. Hasekioglu, B. Tunaboylu, & S. Zaim, Blockchain technology for supply chain management. In Industrial Engineering in the Internet-of-Things World: Selected Papers from the Virtual Global Joint Conference on Industrial Engineering and Its Application Areas, GJCIE 2020, August 14–15, 2020 (2022) (pp. 203-217). Springer International Publishing.

[3] M. S. Al-Rakhami, & M.Al-Mashari, A blockchain-based trust model for the internet of things supply chain management, Sensors. 21(5) (2021) 1759.

[4] A. Park, & H. Li, The effect of blockchain technology on supply chain sustainability performances, Sustainability. 13(4) (2021) 1726.

[5] M. Alazab, S. Alhyari, A. Awajan, & A. B. Abdallah, Blockchain technology in supply chain management: an empirical study of the factors affecting user adoption/acceptance, Cluster Computing. 24(1) (2021) 83-101.

[6] J. Lohmer, E. Ribeiro da Silva, & R. Lasch, Blockchain technology in operations & supply chain management: a content analysis, Sustainability. 14(10) (2022) 6192.

[7] A. Chang, N. El-Rayes, & J. Shi, Blockchain technology for supply chain management: A comprehensive review, FinTech. 1(2) (2022) 191-205.

[8] S. Al-Farsi, M. M. Rathore, & S. Bakiras, Security of blockchain-based supply chain management systems: challenges and opportunities. Applied Sciences. 11(12) (2021) 5585.

[9] S. Al-Amin, S. R. Sharkar, M. S. Kaiser, & M. Biswas, Towards a blockchain-based supply chain management for e-agro business system, In Proceedings of International Conference on Trends in Computational and Cognitive Engineering: Proceedings of TCCE 2020 (2021) (pp. 329-339). Springer Singapore.

[10] G. Elkady, & R. Samrat, An analysis of Blockchain in Supply Chain Management: System Perspective in Current and Future Research, International Business Logistics. 1(2) (2021).

[11] E. Fathalla, & M. Azab, Beyond classical cryptography: A systematic review of post-quantum hash-based signature schemes, security, and optimizations. IEEE access, 12 (2024) 175969-175987.

[12] A. Farouk, B. K. Behera, & E. A. Ahmed, Design and Implement a Quantum Blockchain Framework to Secure 6G Communication for Consumer Applications. IEEE Transactions on Consumer Electronics (2025).

[13] K. Zheng, L. J. Zheng, J. Gauthier, L. Zhou, Y. Xu, A. Behl, & J. Z. Zhang, Blockchain technology for enterprise credit information sharing in supply chain finance, Journal of Innovation & Knowledge. 7(4) (2022) 100256.

[14] R. M. Difrancesco, P. Meena, & G. Kumar, How blockchain technology improves sustainable supply chain processes: a practical guide, Operations Management Research. 16(2) (2023) 620-641.

[15] E. Gökalp, M. O. Gökalp, & S. Çoban, Blockchain-based supply chain management: understanding the determinants of adoption in the context of organizations, Information systems management. 39(2) (2022) 100-121.

[16] E. R. Rizvi, & S.Khurram, BLOCK CHAIN DRIVEN SUPPLY CHAIN SECURITY: INTEGRATING POST QUANTUM CRYPTOGRAPHY WITH AES. Spectrum of Engineering Sciences, 1-14 (2025).

[17] T. Zhang, F. Rahman, M. Tehranipoor, & F. Farahmandi, FPGA-chain: Enabling holistic protection of FPGA supply chain with blockchain technology, IEEE Design & Test. 40(2) (2022) 127-136.

[18] S. K. Sinha, & D. Mukhopadhyay, Time efficient hash key generation for blockchain enabled framework. IEEE Access, 12 (2024) 155867-155884.

[19] M. Xu, S. Ma, & G. Wang, Differential game model of information sharing among supply chain finance based on blockchain technology, Sustainability. 14(12) (2022) 7139.

[20] R. R. Suman, B. Mondal, & T. Mandal, A secure encryption scheme using a Composite Logistic Sine Map (CLSM) and SHA-256, Multimedia Tools and Applications. 81(19) (2022) 27089-27110.

[21] P. Centobelli, R. Cerchione, P. Del Vecchio, E. Oropallo, & G. Secundo, Blockchain technology for bridging trust, traceability and transparency in circular supply chain, Information & Management. 59(7) (2022) 103508.

[22] N. Anjum, & P. Dutta, Identifying counterfeit products using blockchain technology in supply chain system, In 2022 16th international conference on ubiquitous information management and communication (IMCOM) (2022) (pp. 1-5). IEEE.

[23] L. Huang, L. Zhen, J. Wang, & X. Zhang, Blockchain implementation for circular supply chain management: Evaluating critical success factors, Industrial Marketing Management. 102 (2022) 451-464.

[24] P. Azevedo, J. Gomes, & M. Romão, Supply chain traceability using blockchain, Operations Management Research. 16(3) (2023) 1359-1381.

[25] P. K. Patro, R. Jayaraman, K. Salah, & I. Yaqoob, Blockchain-based traceability for the fishery supply chain, Ieee Access. 10 (2022) 81134-81154.

[26] G. E. Visuvanathan, M. S. Sayeed, & S. Yogarayan, BHFVAL: block chain-enabled hierarchical federated variational auto encoder framework for secure intrusion detection in vehicular networks. Scientific Reports (2025).

[27] P. K. Sholapurapu, J. Omkar, S. Bansal, T. Gandhi, P. Tanna, & G. Kalpana, Secure Communication in Wireless Sensor Networks Using Cuckoo Hash-Based Multi-Factor Authentication. In 2025 World Skills Conference on Universal Data Analytics and Sciences (WorldSUAS) (pp. 1-6). IEEE (2025).

[28] J. Curado, M. Bhandari, J. C. Ferreira, & A. L. Martins, Blockchain for Maritime Supply Chain: Efficiency and Security Advancements. IEEE Access, 13 (2025) 201527-201544.

[29] R. García, I. Algredo-Badillo, M. Morales-Sandoval, C.Feregrino-Uribe, & R. Cumplido, A compact FPGA-based processor for the Secure Hash Algorithm SHA-256, Computers & Electrical Engineering. 40(1) (2014) 194-202.

[30] M. R. Adiwiganda, E. Ariyanto, R. Yasirandi, N. A. Suwastika, & Y. A. Setyoko, Adopting tiny encryption algorithm for patient healthcare record on smart card, In 2019 International Conference of Computer Science and Information Technology (ICoSNIKOM) (2019) (pp. 1-5). IEEE.

[31] Z. Ullah, B. Raza, H. Shah, S. Khan, & A. Waheed, Towards blockchain-based secure storage and trusted data sharing scheme for IoT environment, IEEE access. 10, 36978-36994 (2022).

[32] M. Schwarz, Crypto Transaction Speeds 2018-All the Major Cryptocurrencies, aBitGreedy.[Online]. Available: https://www. abitgreedy. com/transaction-speed/.[Accessed: 01-Jun-2019] (2018).

[33] A. Baliga, Understanding blockchain consensus models. Persistent, 4(1) (2017) 14.

[34] S. Datta, & S. Namasudra, Blockchain-based smart contract model for securing healthcare transactions by using consumer electronics and mobile-edge computing, IEEE Transactions on Consumer Electronics. 70(1) (2024) 4026-4036.

[35] Bitshares. Delegated Proof of Stake (DPOS). Available online: https://how.bitshares.works/en/master/technology/dpos, html (accessed on 30 October 2022) (2016).

[36] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, & E. Dutkiewicz, Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities, IEEE access. 7 (2019) 85727-85745.

[37] V. Bachani, Y. Wan, & A.Bhattacharjya, Preferential DPoS: A Scalable Blockchain Schema for High-Frequency Transaction (2022).

[38] M. Shaikh, U. K. Wiil, A. Ebrahimi, & Y.Memon, An Overview and Comparison of Blockchain Consensus Mechanisms. International Journal of Networked and Distributed Computing (2025).