A BLOCKCHAIN-BASED PRIVACY-PRESERVING SOLUTION FOR AGRICULTURAL PRODUCT TRACEABILITY BASED ON GROUP SIGNATURES
/
基于群签名的区块链农产品溯源隐私保护方案

Jiarui ZHANG, Jianyu CHEN, Xuanyu CAO, Shuo LIU, Hao JI, Yongshuai YANG, Lijun CHENG*)
College of Software, Shanxi Agricultural University, Taigu, Shanxi / China;
Corresponding authors: Lijun CHENG; Tel: +86-13835441585; E-mail: cljzyb@sxau.edu.cn
DOI: https://doi.org/10.35633/inmateh-76-90

Keywords: *Blockchain; Group signatures; Agricultural-product traceability; Privacy preservation*

ABSTRACT

Traceability is a crucial component in ensuring the secure sharing of information throughout the entire supply chain of agricultural products. However, existing blockchain-based traceability solutions lack identity anonymity, face challenges in regulatory traceability, and lack dynamic member management capabilities. This paper proposes a blockchain privacy protection scheme integrating group signatures, pseudonymity mechanisms, and revocable accumulators to address these issues. The core design of this scheme encompasses three key aspects: first, this study introduces a dynamic group signature algorithm to balance the anonymity of data signatures with the traceability of responsibility, ensuring both privacy protection and accountability for data sources. Second, this study designs a pseudonym-based identity-hiding and authentication-obfuscation mechanism to enhance privacy protection further, improve resistance to on-chain analysis, and prevent the leakage of user identities. Finally, an efficient dynamic member management protocol is constructed to support rapid node joining and flexible revocation, thereby addressing frequent member changes in agricultural supply chains. Security analysis indicates that the scheme meets stringent security requirements regarding anonymity, non-repudiation, and traceability. Experimental results show that the proposed scheme outperforms existing solutions regarding signature verification overhead, communication costs, and operational efficiency, demonstrating good practicality and scalability, and providing practical support for privacy protection and digital regulation in agricultural supply chains.

摘要

溯源是保障农产品全流程信息安全共享的重要部分。然而，现有基于区块链的溯源方案不具备身份匿名性、监管可追溯性困难以及缺少动态成员管理方面。为了解决这些问题，本文提出了一种融合群签名、伪名机制与可撤销累加器的区块链隐私保护方案。该方案的核心设计包括以下三个方面：首先，引入了动态群签名算法，以实现数据签名的匿名性与责任可追溯性的平衡，确保数据源的隐私保护与责任追溯；其次，设计了基于伪名的身份隐藏与认证混淆机制，进一步加强了隐私保护，提升了抗链上分析的能力，防止用户身份泄露；最后，构建了高效的动态成员管理协议，支持节点的快速加入与灵活撤销，从而应对农业供应链中频繁的成员变动。安全性分析表明，该方案在匿名性、不可伪造性和可追溯性等方面满足严格的安全要求。实验结果显示，本文方案在签名验证开销、通信成本以及运行效率上优于现有方案，具备较好的实用性与可扩展性，为农业供应链中的隐私保护与数字监管提供了有效的支持。

INTRODUCTION

With the ongoing digitalisation of agriculture and the deepening of supply-chain management, enabling trustworthy, end-to-end data sharing and traceability for agricultural products has become a prominent research focus in food-safety governance (*Demestichas et al., 2020; Bermeo-Almeida et al., 2018*). Owing to its decentralisation, immutability and verifiable auditability, blockchain technology provides a technical foundation for multi-stakeholder collaboration “from farm to table.” It has already shown promise in organic certification, geographical-indication protection and cold-chain logistics supervision (*Stranieri et al., 2021; Sharma et al., 2023; Zhang et al., 2020*). Nevertheless, when agricultural participants change frequently, blockchain-based traceability schemes reveal several limitations, including static identity management, weak compatibility between tamper resistance and privacy, and insufficient data trustworthiness (*Xu et al., 2022*). Consequently, devising a privacy-preserving solution that simultaneously offers member anonymity, regulatory traceability, and dynamic identity management has become a pressing challenge.

Current privacy-enhanced blockchain traceability schemes concentrate on three core capabilities—identity authentication, identity anonymity and member management—but still display conspicuous shortcomings in agricultural scenarios. First, group authentication often lacks efficiency, hindering the rapid onboarding of large numbers of nodes. Second, sensitive information may leak during authentication, resulting in inadequate privacy protection. Third, efficient dynamic member management is absent, so permission updates lag in the face of frequent node churn. Therefore, a comprehensive solution that integrates group authentication, advanced privacy preservation and dynamic member management is needed to improve both the applicability and reliability of blockchain traceability in agricultural supply chains. To clarify the practical roots of this demand, representative recent studies are reviewed and assessed below.

In identity authentication, Li et al combined RFID with smart contracts to map physical-tag IDs to on-chain accounts, enabling rapid, correct confirmation across production, storage and sales stages (*Li et al., 2024*). Soy et al merged decentralised identifiers (DIDs) with zero-knowledge proofs to create a one-registration, multi-verification cycle (*Soy et al., 2025*). Gong et al designed a certificate-less lightweight protocol that bridges terminals, gateways and a Hyperledger Fabric blockchain via ABAC and a three-way handshake (*Gong et al., 2024*). Although these schemes shorten initial access latency and increase throughput, they generally rely on static certificates or hardware identifiers. When nodes change frequently, keys must still be reissued or rotated; systems usually broadcast revocation and update operations on-chain or store them in auxiliary tables, compromising real-time performance and cost efficiency. Accordingly, authentication bottlenecks remain unresolved under highly concurrent, group-oriented conditions.

For identity-anonymity protection, Zhang et al proposed a fine-grained and flexible terminal-data access-control scheme based on ciphertext-policy attribute-based encryption (CP-ABE) and a hybrid data-encryption method that supports multi-dimensional sharing by “crop type–batch–role” (*Zhang et al., 2022*). Zhang et al built a multi-chain blockchain architecture supported by zero-knowledge proofs for inter-chain data collaboration and privacy preservation in food supply chains (*Zhang et al., 2023*). Wang et al adopted ring signatures to conceal signer identity and origin, strengthening the anonymity of on-chain information release. While these approaches reinforce the principle of minimal disclosure, they exhibit common weaknesses (*Wang et al., 2024*). As access policies, participants, or the number of chains grow, encryption and proof costs escalate sharply, impeding deployment on resource-constrained agricultural terminals; moreover, strong anonymity often sacrifices accountability, because few schemes offer an optional identity-decryption window for compliance audits, making it challenging to balance privacy with regulatory needs.

Regarding member management, Wang et al presented a blockchain-based proxy re-encryption access-control method (BBPR-AC) that protects agro-biological risk information by defining attributes for each supply-chain stage, establishing policies and executing them via smart contracts to achieve decentralised, automated authorization (*Wang et al., 2024*). Peng et al developed a dynamic supervision model driven by smart contracts to revoke real-time members, enhancing transparency and quality assurance in the rice supply chain (*Peng et al., 2022*). Sun et al designed a provenance-aware dynamic access-control scheme with a fast lookup table for member control (*Sun et al., 2023*). Although these studies improve permission-configuration flexibility, they still face common challenges: on-chain revocation operations can cause congestion and high gas costs; centralised permission logic may become a performance bottleneck; and “expired identities” must be synchronised off-chain. Low-cost yet real-time dynamic-management capability remains insufficient for agricultural scenarios with high node-turnover rates. In summary, existing research has not yet achieved coordinated optimisation across the triple dimensions of large-scale group authentication, strong privacy protection and efficient dynamic management, leaving room for a comprehensive framework that reconciles anonymity, accountability and scalability.

To address the deficiencies in identity anonymity, traceability and member revocation, this paper makes the following contributions from the perspectives of group signatures, pseudonym protection and dynamic identity management:

1. Group signature for agricultural traceability. A group-signature scheme tailored to agricultural product traceability that combines blockchain verifiability with signature anonymity is proposed. While protecting the identities of ordinary users, designated authorities can decrypt identity information and revoke signatures, thereby achieving an effective balance between privacy preservation and accountability. The scheme enhances the traceability of responsibilities under strict privacy requirements.

2. Pseudonym-based identity-mapping and dynamic-obfuscation mechanism. A pseudonym-identity mapping mechanism that dynamically obfuscates authentication information during data interactions is introduced. This mechanism prevents on-chain behavioural pattern tracking, boosts resistance to analytical

attacks and enforces business-data separation, guaranteeing anonymity in traceability information while deterring the abuse of malicious behavioural tracking.

3. Lightweight member-management protocol. Given the frequent node changes in agricultural supply chains, a lightweight member-management protocol that supports rapid node admission, flexible revocation and synchronised key-state updates is designed. Through group key-agreement and authentication-reconstruction algorithms, the protocol ensures security and consistency while providing scalability and robustness. It effectively meets the demands of dynamic network environments, improving the adaptability and flexibility of blockchain traceability systems in agriculture.

The notation used throughout this paper is defined in Table 1.

Table 1

Symbol Definition

Symbol	Meaning
G_1, G_2, G_T	Pairing source and target groups ($e: G_1 \times G_2 \rightarrow G_T$)
q	G_1, G_2, G_T of order of a large integer prime
P_1 , P_2	Generators of groups G_1, G_2 ($\varphi(P_2) = P_1$)
P_{pub}	System master public key ($P_{pub} = s \cdot P_1$)
$H(\cdot)$	Collision-resistant hash function
L	Table of real identities
P	Pseudonymised table of identities
id_i	User's real identities
PID	Pseudonymised identities ($PID = H(id_i) \cdot P_1$)
ACC_0	Initial accumulator ($ACC_0 = r \cdot P_2$)

MATERIALS AND METHODS

System model

Figure 1 illustrates the structure of each functional role in this scheme and its specific responsibilities within the agricultural product traceability system.

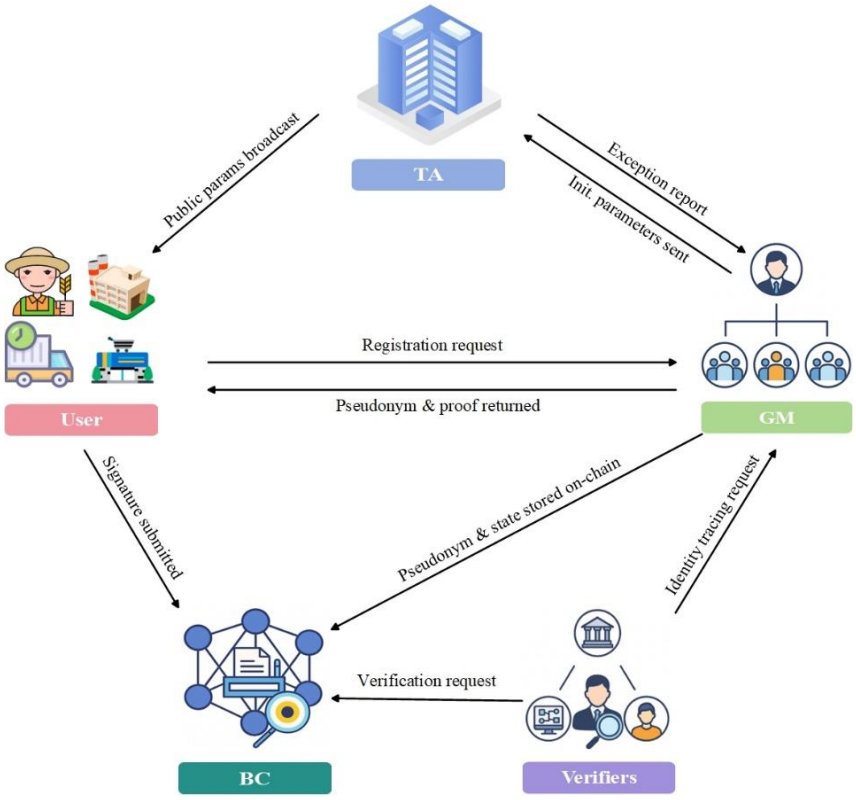


Fig. 1 - System Model

The diagram defines five primary roles:

Trusted Authority (TA): Responsible for generating and publishing cryptographic parameters during the initialisation phase of the scheme, coordinating user identity registration processes, distributing authentication

credentials, and managing dynamic updates to group keys when network members change. The TA is assumed to be a fully trusted entity operating offline, participating in system operations only when necessary.

Group Manager (GM): Responsible for user joining and revocation, signature validity verification, and identity de-anonymisation when necessary. Typically authorised and generated by the TA, the GM assumes daily identity management and audit responsibilities during regular system operation.

Users (U): Entities or organisations involved in various stages of the supply chain, including agricultural product cultivation, processing, transportation, and sales. Users obtain anonymous authentication credentials during registration and use them for traceability data signing operations. The scheme supports dynamic user joining and secure exit mechanisms to ensure long-term scalability.

Verifiers (V): Verification entities, including regulatory authorities, traceability platforms, and end consumers. Their primary tasks are to verify the validity and integrity of on-chain data signatures and determine the validity of the signer's identity. Among these, regulatory authorities serve as privileged verifiers, able to decrypt member identities and trace abnormal behaviour once they meet the required conditions.

Blockchain (BC): As a distributed storage infrastructure, it is responsible for recording and storing signature information and hash summaries of traceability data. The blockchain network only stores cryptographic proofs required for verification and does not contain plaintext identity information, effectively combining data auditability with identity anonymity.

Scheme Overview

Combining the requirements for secure sharing and privacy protection of agricultural product data in traceability scenarios, the traceability authentication scheme based on group signatures proposed in this paper consists of the following six probabilistic polynomial-time algorithms:

$Setup(1^{\lambda}) \rightarrow (params, msk)$: System initialisation algorithm. Under security parameter λ, it generates bilinear group parameters, hash functions, and accumulator initial values, outputs system public parameters params and master key msk, and publishes $params$ to the blockchain.

$Join(U_i, T_1, \sigma_{id_i}) \rightarrow (PID_i, W_i, ACC_j)$: User U_i sends a registration request. After the group administrator verifies the identity signature σ_{id_i} it assigns a pseudonym PID_i and membership proof W_i to the user and updates the accumulator state ACC_j.

$Sign(sk_i, M) \rightarrow \sigma$: The user U_i uses their private key sk_i to generate an anonymous group signature $= \{M, T_1, T_2, T_3, A, c, S_1, S_2\}$ for message M, ensuring the integrity and anonymity of the traceable data.

$Verify(\sigma, params, \mathcal{P}) \rightarrow \{0,1\}$: The verifier checks the integrity and validity of the signature structure based on the public parameters and pseudonym list P to confirm whether the signature comes from a valid, non-revoked member.

$Open(\sigma, s) \rightarrow id_i$: The group administrator uses the private parameter s from the master key to recover the corresponding pseudonym from the signature σ and searches the identity list to restore the real identity id_i, thereby achieving traceability of behaviour.

$Revoke(id_k, L) \rightarrow (ACC, W_i)$: When member id_k is revoked, the system calculates the update factor based on their identity information, updates the accumulator state ACC and the proof values W_i of the remaining members, ensuring the security of subsequent signatures.

Scheme Construction

(1) System Initialisation

During the initial deployment phase of the system, the TA is responsible for generating global public parameters and the system's initial state. This corresponds to the standard setup algorithm $Setup(1^{\lambda}) \rightarrow (params, msk)$. To reduce trust dependencies, this scheme centralises the initialisation responsibilities in the TA, which performs a one-time execution. After publishing the parameters, the TA transitions to an offline state, and subsequent operations are handled by the GM. The initialisation steps are as follows:

First, the TA sets a finite field $\mathbb{F}_q$. It constructs groups G_1 and G_2 and defines a bilinear mapping $e: G_1 \times G_2 \rightarrow G_T$. It randomly selects P_1 and P_2 such that there exists a homomorphic mapping φ that maps group G_2 to group G_1, satisfying $\varphi(P_2) = P_1$. Next, the TA randomly selects two non-zero integers $s, r \in \mathbb{Z}_q^*$, where s does the GM hold the system master private key. Based on this, $P_{pub} = s \cdot P_1$ is calculated, and the accumulator state is initialised as $ACC_0 = r \cdot P_2$, providing verifiable support for subsequent user registration and revocation operations. To enhance identity privacy protection, a collision-resistant hash function $H(\cdot)$ is introduced for identity pseudonymisation processing.

The system maintains two types of identity mapping tables to efficiently manage user identity states:

① Real Identity list L: The record format is a quintuple $< id_i, PID, join/delete, W_i, ACC_j >$, where id_i is the user's real identity identifier, and the pseudonym PID is calculated using the formula $PID = H(id_i) \cdot P_1$ to protect the user's real identity information;

② Pseudonym identity list P: The record format is $< PID, join/delete, W_i, ACC_j >$, used to publish the user's registration and revocation status publicly. This list is stored on-chain via blockchain, providing external verifiers with an auditable, tamper-proof mechanism for querying revocation status.

TA ultimately discloses the following parameter set to the system:

$$params = \{q, G_1, G_2, G_T, e, P_1, P_2, P_{pub}, ACC_0, H(\,)\}$$

(2) User Registration Phase

The user registration phase corresponds to the Join algorithm $Join(U_i, T_1, \sigma_{id_i}) \rightarrow (PID_i, W_i, ACC_j)$. Users first submit a registration request during the user registration phase by generating a private value and an authentication signature. The GM verifies the user's identity and generates a pseudonymous identity identifier based on the accumulator mechanism. Next, the GM writes the relevant information into the real Identity list L and synchronises it with the blockchain to ensure the system's transparency and auditability. The user ultimately receives the private key triplet via a secure channel, which is used for subsequent agricultural product traceability data signing and privacy protection. As shown in Figure 2, the detailed steps of the user registration phase are presented.

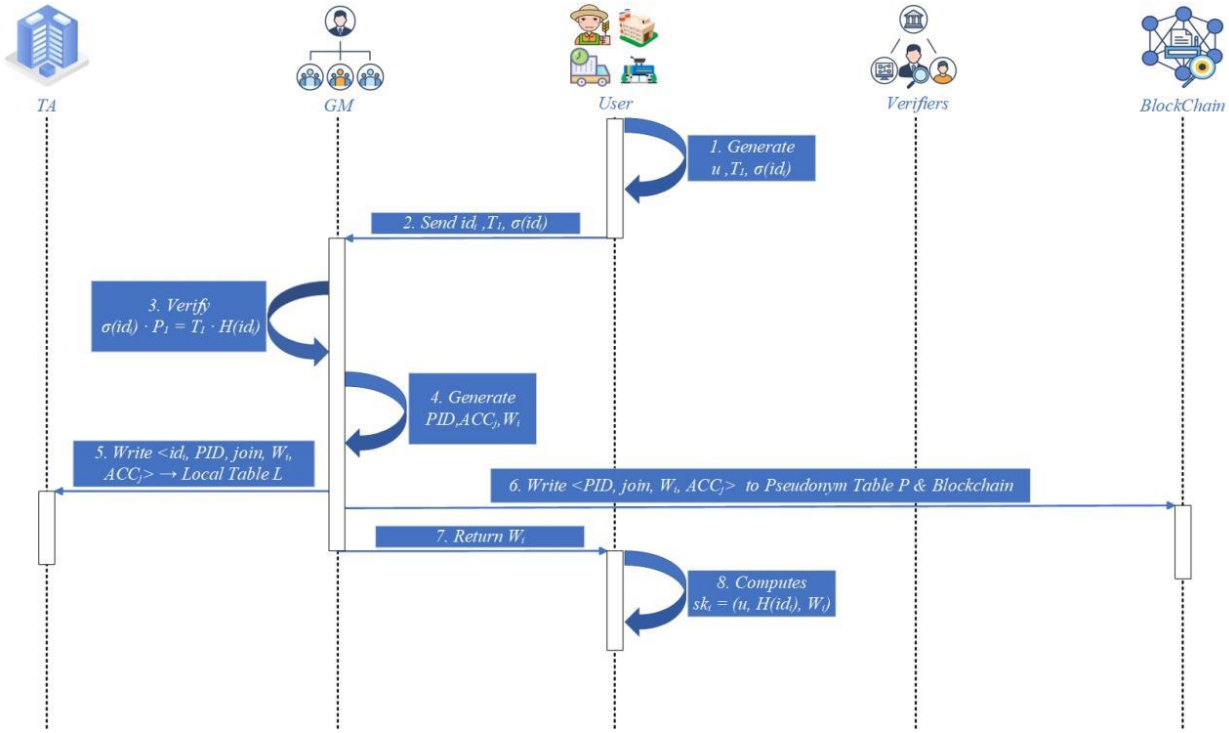


Fig. 2 - User Registration Phase

If U_i wishes to become a chain-based authentication entity and obtain legitimate signing permissions, it must submit a registration request to GM. The specific registration phase is as follows:

Step 1: U_i first randomly selects a private value $u \in \mathbb{Z}_q^*$, calculates the authentication parameter $T_1 = u \cdot P_1$, and generates an authentication signature $\sigma_{id_i} = u \cdot H(id_i)$ based on its identity identifier.

Step 2: Subsequently, U_i submits the authentication triplet $\{\sigma_{id_i}, id_i, T_1\}$ to GM, which verifies its validity through $\sigma_{id_i} \cdot P_1 = T_1 \cdot H(id_i)$. If the verification is successful, it indicates that U_i is valid in the local regulatory whitelist and proceeds to the next stage.

Step 3: GM generates a pseudonymous identity identifier $PID = H(id_i) \cdot P_1$ for $U_i$based on this, and updates the system state according to the accumulator mechanism to generate a new accumulator value $ACC_j = (H(id_i) + s) \cdot ACC_{j-1}$. At the same time, $W_i = ACC_{j-1}$ is recorded as the membership proof value for U_i. Subsequent signature verification relies on this value to ensure that its identity has not been revoked.

Step 4: The GM writes the five-tuple $< id_i, PID, join, W_i, ACC_j >$ into the local real identity list L, while recording the pseudonymous identity information $< PID, join/delete, W_i, ACC_j >$ in the pseudonym identity list P and synchronously writing it to the blockchain. This enables the registration status of participants in the traceability system to be queried in real time by upstream and downstream enterprises, consumers, and regulatory authorities, thereby enhancing the system's transparency and auditability.

Step 5: The GM returns W_i to U_i via a secure channel. U_i combines its private key parameters to locally generate the final private key triplet $sk_i = (u, H(id_i), W_i)$. This key will be used in subsequent data signing to achieve responsibility signing and privacy protection for agricultural product traceability data.

(3) Signature Generation Phase

The signature generation phase corresponds to the Sign algorithm $Sign(sk_i, M) \rightarrow \sigma$. After completing registration and obtaining legal member status, U_i can perform group signature operations on any message M to achieve identity authentication and accountability tracing under anonymity. As shown in Figure 3, the specific steps for generating group signatures and the information exchange process between participating parties are illustrated.

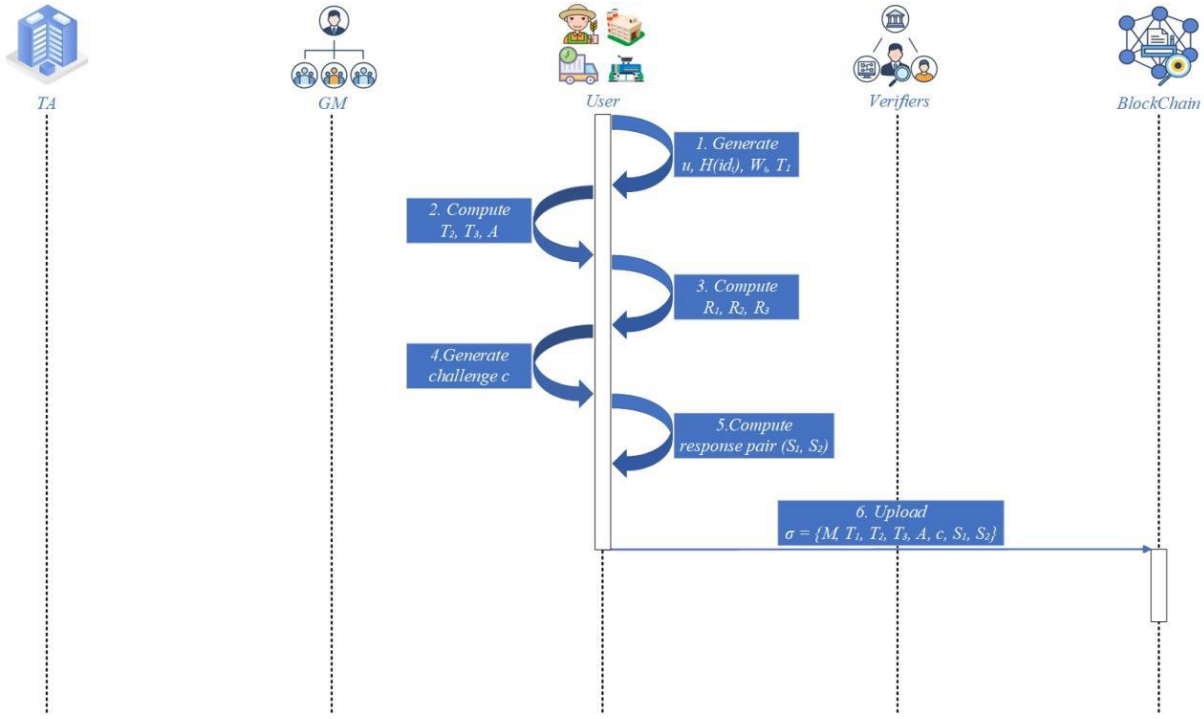


Fig. 3 - Group Signature Generation Phase

The group signature generation phase proceeds as follows:

Step 1: User U_i first uses the membership proof value W_i generated in the registration stage, the identity hash value $H(id_i)$, and the private parameter $u \in \mathbb{Z}_q^*$ to construct the following signature auxiliary quantities. Where $T_1 = u \cdot P_1$ is an intermediate variable generated and retained by U_i during the registration phase, followed by the calculation of $T_2 = H(id_i) \cdot P_1 + u \cdot P_{pub}$, $T_3 = u \cdot P_{pub}, A = u \cdot W_i$.

Step 2: Randomly select v from group $\mathbb{Z}_q^*$ to calculate the intermediate values $R_1 = v \cdot P_1$, $R_2 = e(P_{pub}, A) \cdot e(T_1, W_i)^{H(id_i)}$, $R_3 = v \cdot P_1 + v \cdot P_{pub}$.

Step 3: Based on the above intermediate values, U_i uses the hash function $H()$ to calculate the challenge value $c = H(M, T_1, T_2, T_3, A, R_1, R_2, R_3)$. Subsequently, U_i calculates the signature response pair $S_1 = v + c \cdot u$, $S_2 = v + c \cdot H(id_i)$.

Step 4: User U_i constructs the final group signature result as $\sigma = \{M, T_1, T_2, T_3, A, c, S_1, S_2\}$and uploads it to the blockchain. Since the signature does not contain the user's real identity information, it ensures that the data records of the participants in the supply chain are traceable but not disclosed.

At the same time, the on-chain records also serve as irrefutable evidence for subsequent audits, meeting the comprehensive requirements of the agricultural product traceability system for data security, anonymity, and verifiability. The verifier V can complete the verification process by simply accessing the blockchain, without needing to contact the signers or GM, thereby significantly enhancing the system's scalability.

(4) Signature Verification Phase

The signature verification phase corresponds to the Verify algorithm $Verify(M, \sigma, params) \rightarrow \{0,1\}$.The verifier V only needs to access the blockchain to complete the verification operation, without contacting the signer or GM, thereby enhancing the system's scalability. As shown in Figure 4, the specific steps of the signature verification process and the information exchange process between the parties involved are illustrated. Through this sequence diagram, readers can clearly understand the execution order of each step and the detailed process of signature verification.

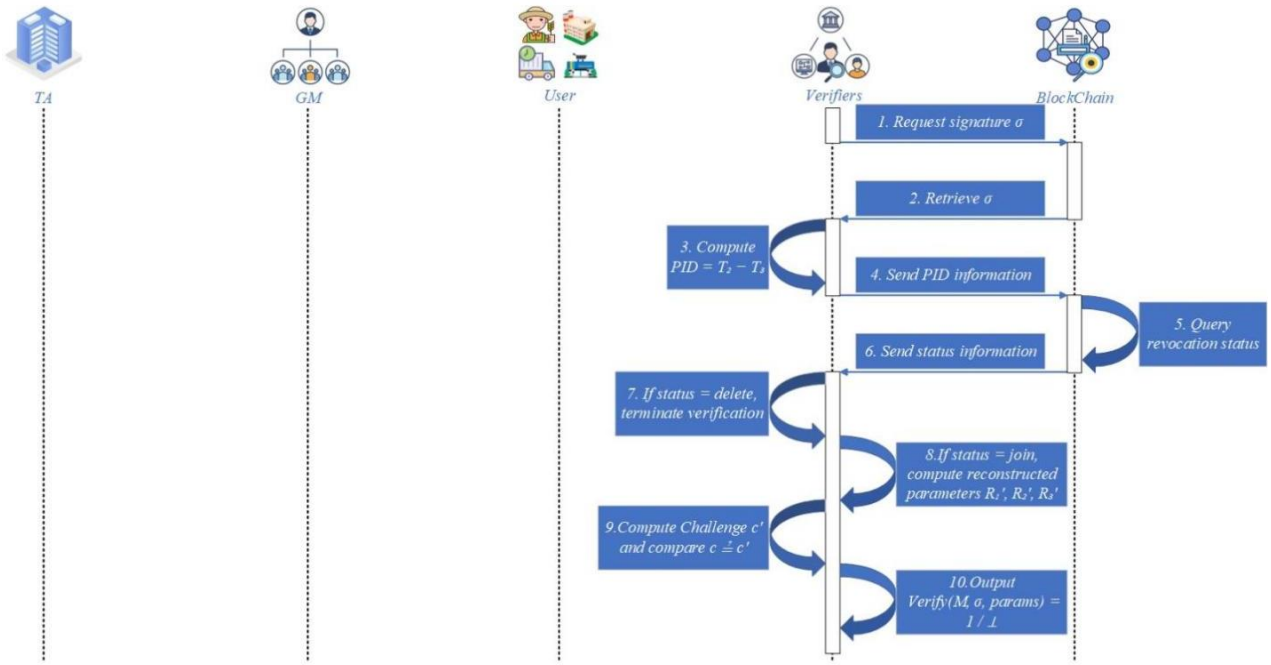


Fig. 4 - Signature Verification Phase

When the verifier obtains the signature $\sigma = \{\mathrm{M}, \mathrm{T}_1, \mathrm{T}_2, \mathrm{T}_3, \mathrm{A}, \mathrm{c}, \mathrm{S}_1, \mathrm{S}_2\}$ from the blockchain, the following steps must be performed in sequence to confirm its validity and the validity of the member's identity:

Step 1: The verifier V performs a pseudonym extraction and revocation status query. Based on the intermediate variables in the signature, the verifier calculates the user's pseudonym $PID = T_2 - T_3$. Then, the verifier searches the pseudonym identity list P maintained by the system for a matching record in the format $< \mathrm{PID}, \mathrm{join/delete}, \mathrm{W_i}, \mathrm{ACC_j} >$. If the current record status is deleted, it indicates that the member has been revoked, and the verification process terminates; otherwise, the following step is executed.

Step 2: Verifier V reconstructs the intermediate variables that should be generated during the signing process based on the public parameters and the signature response value, and calculates $R_1' = S_1 \cdot P_1 - c \cdot T_1$, $R_2' = e(T_1, ACC_j)$, $R_3' = S_2 \cdot P_1 + S_1 \cdot P_{pub} - c \cdot T_2$. Where ACC_j is the current accumulator state obtained from the pseudonymous identity list.

Step 3: Perform a hash consistency check. Verifier V inputs the message and intermediate parameters into the hash function to calculate the challenge value $c' = H(M, T_1, T_2, T_3, A, R_1', R_2', R_3')$. Next, compare the challenge value c' with the original challenge value c carried in the signature. If $c = c'$, it indicates that the signature was indeed generated by a legitimate member in accordance with the protocol specifications, the content has not been tampered with, and the associated pseudonym has not been revoked; otherwise, it is considered an invalid signature.

Step 4: If the consistency verification is valid, i.e., $c = c'$ the verification passes, and the output result is $Verify(M, \sigma, params) = 1$, indicating that the signature is valid and effective. If it is not valid, i.e., $c \neq c'$, the verification fails, and the output result is $Verify(M, \sigma, params) = 0$, indicating that the signature is invalid or the signer's identity has been revoked.

(5) Signature De-anonymisation Phase

The Signature de-anonymization phase corresponds to the Open algorithm $Open(\sigma, s) \rightarrow id_i$. Figure 5 illustrates the specific steps of the de-anonymisation process and the information exchange flow between parties.

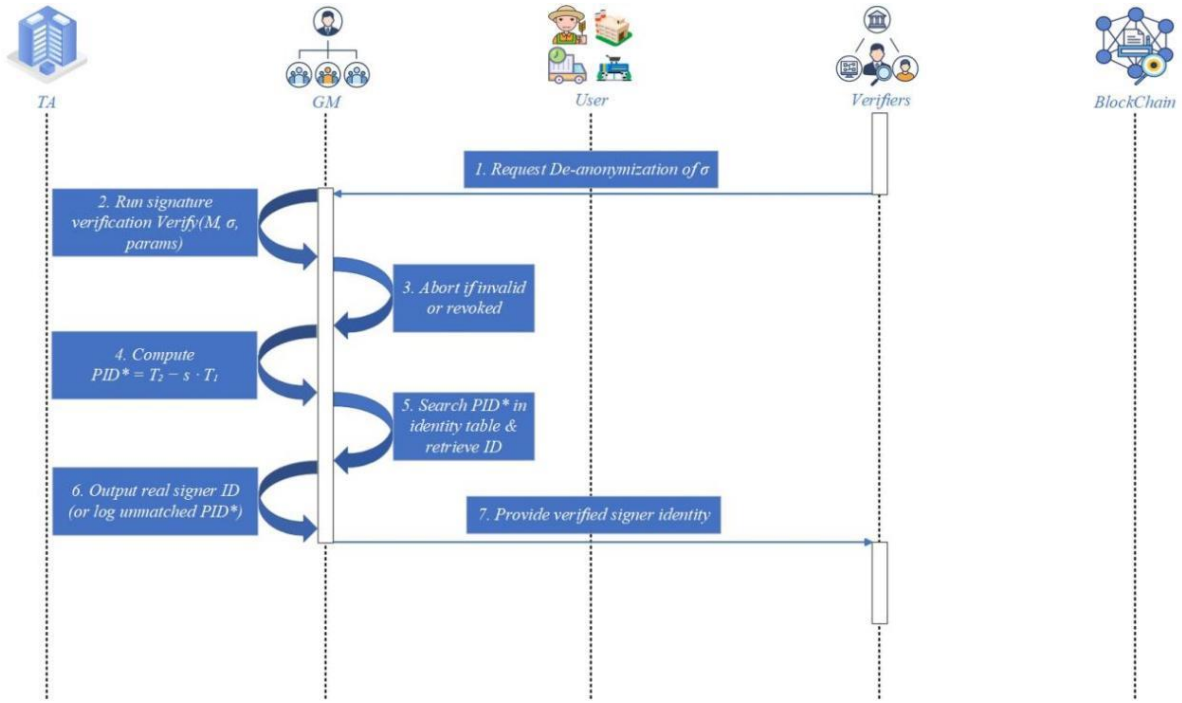


Fig. 5 - De-anonymization Operation Process

When the regulatory authority needs to perform de-anonymization on a chain signature $\sigma = \{M, T_1, T_2, T_3, A, c, S_1, S_2\}$ to trace the identity of the actual signer, the following steps must be executed:

Step 1: Before performing de-anonymization, the GM first calls the verification algorithm $Verify(M, \sigma, params)$ to validate the signature's validity. If the result is $\perp$, indicating the signature is invalid or the member has been revoked, the de-anonymization process is terminated, and the signer's identity is not disclosed; otherwise, the process continues to the next step.

Step 2: After verifying the signature's validity, the GM calculates the signer's pseudonym identifier based on the system master private key s generated during system initialisation, combined with the intermediate variables T_1 and T_2 contained in the signature $PID_* = T_2 - s \cdot T_1 = H(id_*) \cdot P_1$. Here, PID_* is the pseudonym bound to the signer's identity, used to locate their identity record in the member list.

Step 3: The GM then traverses the real identity list L maintained by the system to find the record item $< id_*, PID_*, join, W_i, ACC_k >$ corresponding to PID_*. If a match is found, the real identity i id_* of the signer can be restored, completing the traceable authentication of the signature behavior, thereby enabling the regulatory authority to implement accountability mechanisms for the behaviour of group members. If no match is found, it may indicate that the signature did not originate from a current valid member, posing risks such as signature forgery, member revocation, or protocol inconsistency. The system refuses to recognize the signature in such cases and logs the relevant data for subsequent audit and traceability analysis.

(6) Membership Revocation Phase

The membership revocation phase corresponds to the Revoke algorithm $Revoke(id_k, L) \rightarrow (ACC, W_i)$.To ensure the security and verifiability of the group signature system under dynamic changes in member identities, when the regulatory authority identifies violations, identity anomalies, or termination of cooperation involving a supply chain participant (such as a farmer, transporter, or seller) and decides to revoke their signature permissions, the system must execute the following steps to dynamically update the status, thereby ensuring the reliability and consistency of identity management across all nodes in the agricultural product traceability chain. As shown in Figure 6, the specific steps of the cancellation operation and the information exchange process among all parties are elaborated in detail.

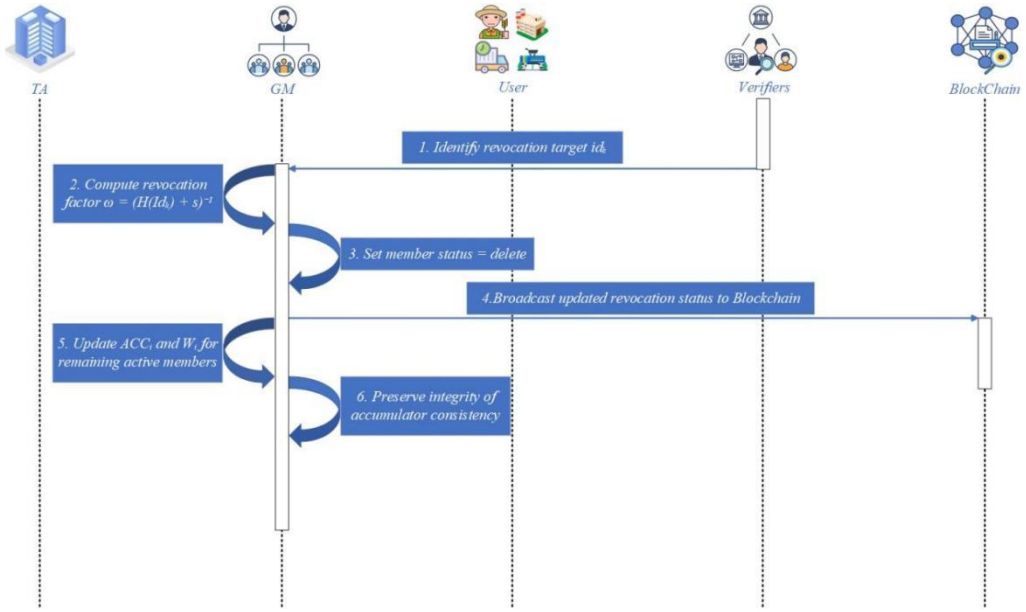


Fig. 6 - Revocation Operation Process

For the member entity l_k to be revoked, it should be a valid member in the current identity table L.

Step 1: The GM calculates the update factor $\omega = (H(id_k) + s)^{-1}$ required for revocation based on the identity identifier id_k of the member to be revoked and the system master private key s. This factor is used to adjust the current accumulator state and completely remove the member's valid contribution from the system state.

Step 2: The GM sets the status of member l_k to "$delete$". The updated pseudonym status quadruple $< PID_k, delete, W_i, ACC_j >$ is written to the pseudonym identity list P and synchronously broadcast to the blockchain network. This process ensures that all validators (such as government regulatory nodes and downstream enterprises) can obtain real-time updates on member status changes, preventing revoked members from continuing to perform signing operations, thereby enhancing the system's regulatory capabilities and transparency.

Step 3: To avoid the continued impact of revoked members on the system's signature logic, the GM must recalculate the accumulator state and proof value $ACC_i = ACC_i \cdot \omega, W_i = W_i \cdot \omega$ for all subsequent members in the member list with index $i \in [k+1, n]$ and current status as "$join$". This process ensures the mathematical consistency of the revocation operation, ensuring that the accumulator states and proof values of all valid members are synchronized with the current system state, thereby preventing revoked entities from bypassing the verification mechanism through historical signatures or leaked information.

RESULTS

Security Analysis

(1) Correctness Proof

The following equation can be used to derive the correctness of this group signature scheme:

$$\begin{aligned} e(P_{pub}, A) \cdot e(T_1, A)^{H(id_i)} &= e(s \cdot P_1, u \cdot W_i) \cdot e(u \cdot P_1, W_i)^{H(id_i)} \\ &= e(T_1, W_i)^s \cdot e(T_1, W_i)^{H(id_i)} \\ &= e(T_1, W_i)^{s+H(id_i)} \\ &= e(T_1, ACC_j) \end{aligned} \quad (1)$$

The above derivation shows that if the signature is generated by a legitimate member based on the protocol, the submitted signature data satisfies the final pairing equation, which completes the validation of the signature's legitimacy.

(2) Anonymity

Theorem 1: Assuming the hash function $H(\,)$ is modeled as a random oracle and the underlying bilinear map satisfies the Decisional Bilinear Diffie–Hellman (DBDH) assumption (*Boneh et al., 2001*), the proposed group-signature scheme achieves the anonymity security goal.

Proof of Theorem 1: Assuming a polynomial-time adversary $\mathcal{A}$, capable of attacking anonymity with a non-negligible probabilistic advantage, an algorithm $\mathcal{B}$ is constructed that solves the DBDH problem with $\mathcal{A}$ as a subroutine, thus inducing a contradiction.

To portray the ability of the adversary, the following challenge game $\mathcal{G}_{anon}^{\mathcal{A}}$ is defined:

① System initialisation: the adversary $\mathcal{A}$ obtains the system parameter params and can optionally initiate queries such as registration and signature.

② Challenge phase: the adversary $\mathcal{A}$ specifies two legitimate members U_0, U_1 and a message M^*, and the system randomly selects $b \in \{0,1\}$, and the signature U_b is generated σ^*.

③ Guessing phase: the adversary $\mathcal{A}$ outputs a guess b', and the challenge succeeds if $b' = b$.

The advantage of adversary $\mathcal{A}$ is defined as:

$$\mathrm{Adv}_{\Pi}^{anon}(\mathcal{A}) = \left| Pr[\, b' = b] - \frac{1}{2} \right| \tag{2}$$

If for any PPT (Probabilistic Polynomial Time) adversary $\mathcal{A}$, there is $\mathrm{Adv}_{\Pi}^{anon}(\mathcal{A}) \leq \mu(\lambda)$, where $\mu(\lambda)$ is a negligible function, then the scheme satisfies anonymity.

Based on the above game, construct Algorithm B to perform the following simulation:

① Input instances: Algorithm B receives DBDH instances (g, g^a, g^b, g^c, Z), and the goal is to determine whether $Z = e(g,g)^{abc}$ or $Z \in R \in G_T$.

② Parameter simulation: Algorithm B constructs the group signature system with the public parameters $P_1 = g, P_2 = g^a, P_{pub} = g^b$, and the initial accumulator is $ACC_0 = g^{ar}$, where $r \in \mathbb{Z}_q^*$ is randomly generated. Subsequently, the system parameter $params$ is returned to the adversary $\mathcal{A}$.

③ Query simulation: Algorithm $\mathcal{B}$ simulates the registration and signature operations, and models the hash function $H(\cdot)$ as a random predicator, which returns consistent random values for each hash query and maintains query consistency.

④ Challenge phase: adversary $\mathcal{A}$ submits (U_0, U_1, M^*). Algorithm $\mathcal{B}$ randomly selects $b \in \{0,1\}$, constructs a signature using member U_b, and embeds the DBDH instance parameters. Let $T_3 = g^c = u \cdot P_{pub}$, $T_2 = PID_b + T_3$, $A = u \cdot W_b$, where $r \in \mathbb{Z}_q^*$ is randomly generated and the pairing term $R_2 = Z = e(g,g)^{abc}$, and finally construct the signature $\sigma^* = \{M^*, T_1, T_2, T_3, A, c, S_1, S_2\}$ and return it to the adversary $\mathcal{A}$.

⑤ Judgment of the output: If adversary $\mathcal{A}$outputs guess b', algorithm B decides accordingly: if $b' = b$ then output 1, considering $Z = e(g,g)^{abc}$. If $b' \neq b$, then output 0, considering $Z \in R \in G_T$ as a random element.

⑥ Successfulness analysis: if $Z = e(g,g)^{abc}$, the signature structure is legitimate and $\mathcal{A}$ cannot distinguish the membership; if $Z \in R \in G_T$, the pairing term R_2 is a random value, the signature does not satisfy the verification condition, and the adversary cannot obtain valid information. Therefore, if $\mathcal{A}$ has a non-negligible advantage ϵ to distinguish b, then $\mathcal{B}$ can distinguish DBDH instances from random values with the same advantage, which yields:

$$\mathrm{Adv}_{DBDH}^{\mathcal{B}} = |Pr[\mathcal{B}(Z = e(g,g)^{abc}) = 1] - Pr[\mathcal{B}(Z \in R \in G_T) = 1]| = \epsilon \tag{3}$$

This contradicts the DBDH assumption, so the original assumption is not valid. In summary, this group signature scheme satisfies the anonymity security requirement under the defined anonymity challenge model. The proof is complete.

(3) Unforgeability

Theorem 2: If the Discrete Logarithm Problem (DLP) is challenging in a cyclic group G_1 and the adversary has no access to the private keys of the legitimate members, then the signature scheme of this group satisfies existential unforgeability under the Choice Message Attack (EUF-CMA) model (*Bellare et al., 2003*).

Proof of Theorem 2: Assuming that a polynomial-time adversary $\mathcal{A}$ that can successfully forge an unqueried legitimate signature with non-negligible probability, an algorithm $\mathcal{B}$ is constructed that solves the DLP problem with $\mathcal{A}$ as a subroutine, thus inducing a contradiction.

To portray the ability of the adversary, the following challenge game $\mathcal{G}_{\mathrm{ufg}}^{\mathcal{A}}$ is defined:

① System initialisation: the system generates the parameter $params$ and registers multiple members. The adversary $\mathcal{A}$ obtains all public parameters and pseudonym information, but cannot obtain the private key of a challenge member.

② Query phase: adversary $\mathcal{A}$ can initiate registration, signature and hash function $H(\cdot)$ queries.

③ Output phase: Adversary $\mathcal{A}$ outputs a forged signature $\sigma^* = \{M^*, T_1^*, T_2^*, T_3^*, A^*, c^*, S_1^*, S_2^*\}$, where M^* has not been queried and the signature corresponds to a challenge member but is not generated by it. The forgery success probability of adversary $\mathcal{A}$ is:

$$\mathrm{Adv}_{\Pi}^{\mathrm{ufg}}(\mathcal{A}) = \Pr[\mathcal{A}\mathrm{wins}\mathcal{G}_{\mathrm{ufg}}^{\mathcal{A}}] \quad (4)$$

If for any PPT adversary $\mathcal{A}$, there is $\mathrm{Adv}_{\Pi}^{\mathrm{ufg}}(\mathcal{A}) \leq \mu(\lambda)$, where $\mu(\lambda)$ is a negligible function, then this scheme satisfies EUF-CMA unforgeability.

Based on the above game, construct Algorithm $\mathcal{B}$ to perform the following simulation:

① Input instance: algorithm $\mathcal{B}$ receives a DLP instance $(P_1, P_{pub} = s \cdot P_1)$, and the goal is to compute $s \in \mathbb{Z}_q^*$.

② Parameter simulation: Algorithm $\mathcal{B}$ designates P_1 as a generator in the DLP instance, randomly chooses $r \in \mathbb{Z}_q^*$, and sets the initial accumulator to be $ACC_0 = r \cdot P_2$. The rest of the parameters, such as P_2, G_2, G_T, e, are randomly set by Algorithm $\mathcal{B}$. The hash function $H(\,)$ is modeled as a random oracle, with all queries recorded to ensure consistency. Eventually, Algorithm B returns the system parameter $params$ to the adversary $\mathcal{A}$.

③ Simulated querying: Algorithm $\mathcal{B}$ completely simulates all the registration and signature requests of $\mathcal{A}$; autonomously assigns the pseudo-name PID_i and computes the accumulator state; constructs a legitimate signature using the simulated membership key $(u_i, H(id_i), W_i)$, and specifically marks a challenge member U^*, for which it does not return the private key and does not accept signature request.

④ Forged signature and private key extraction: when the adversary $\mathcal{A}$ outputs a legitimate signature σ^* for the message M^*, if it satisfies the verification conditions and is attributed to the challenge member U^*, then from the signature parameter relationship, it can be seen that $T_1^* = u^* \cdot P_1, T_3^* = u^* \cdot P_{pub} = u^* \cdot s \cdot P_1$. Since T_1^* and T_3^* are in the same subgroup of the generator P_1, there exists a unique scalar s satisfying $T_3^* = s \cdot T_1^*$. Therefore, $\mathcal{B}$ can solve the DLP by the following formula:

$$T_3^* = s \cdot T_1^* \Rightarrow s = \frac{T_3^*}{T_1^*} \quad (5)$$

⑤ Successfulness analysis: if the probability of success of the forgery of the adversary $\mathcal{A}$ is ϵ, the algorithm $\mathcal{B}$ solves the DLP with the same advantage, which is in contradiction with the DLP assumption.

Therefore, assuming the DLP assumption is valid and the adversary cannot access the private keys of legitimate members, this group signature scheme satisfies existential unforgeability under the EUF-CMA model. The proof is complete.

(4) Traceability

Given any group signature $\sigma = \{M, T_1, T_2, T_3, A, c, S_1, S_2\}$, and only the GM owns the master private key s in the system, the pseudonym identity of the signer can be recovered by calculating the $PID_* = T_2 - s \cdot T_1$, and combining with the local identity mapping table to uniquely locate the real identity id_i, to achieve the effective traceability of the signer. Each valid signature corresponds to a unique identity identifier to ensure the accuracy of the traceability result. In the process of signature generation, the random factor u introduced each time makes that even if the same member generates signatures for many times, the parameters of its signature components T_1, T_2, etc. remain random in distribution, ensuring the uniqueness and unpredictability of each signature. Only if the signature is verified and confirmed to be valid, the GM performs the de-anonymisation operation, thus restoring the real identity and completing the traceability. This mechanism ensures that all signatures can be effectively traced back to the corresponding authentic signer in the produce traceability system.

(5) Forward-backward unlinkability

Suppose member U_i is revoked and the attacker holds the updated parameter ACC_j'. Since ACC_j' is inconsistent with $\mathrm{ACC_j}$ used for historical signatures and PID_i is no longer publicly available, the verifier is unable to obtain W_i, and thus is unable to associate the signature with U_i, thus verifying forward privacy.

Assuming that the new member U_k tries to identify the signer of the old signature σ_i, since the new member's key does not contain $H(id_i)$ or W_i and u in $T_3 = u \cdot P_{pub}$ was generated by the old member, the new member is unable to construct signatures that can validate this signature structure, and thus cannot distinguish historical identities.

Thus, forward privacy arises from the binding of the historical accumulator snapshot $\mathrm{ACC_j}$ in the signature structure, while backward privacy relies on the unidirectional evolution of the accumulator with the

irreversibility of the hash function. As long as the hash function is collision-resistant and the accumulator is irreversible, the present swarm signature system satisfies forward and backward privacy.

Performance Analysis

(1) Experimental Environment

This experiment was conducted on a high-performance platform with a hardware configuration of an Intel Core™ i7-9750H (2.60 GHz) central processor, an NVIDIA GeForce GTX 1650 graphics processor, and 16 GB of operating memory. The software environment comprised the Windows 11 operating system, IntelliJ IDEA 2023.2.1 development tool, and Java programming language. In addition, the JPBC library (version 2.0.0) was used in the experiments and the cryptographic operations were performed based on type A pairing profiles (a.properties) with a 512-bit base field. The key generation process used the SHA-256 hash algorithm, and the encryption algorithm used the AES encryption standard. All experiments were conducted under uniform hardware and software conditions to ensure the reproducibility of the experimental results.

The content of Table 2 demonstrates the execution time statistics of the cryptographic operations used in this experiment:

Table 2

Cryptographic Operational Definitions and Average Runtime

Symbol Definition	Type of operation	Execution time (average)
T_{bp}	Bilinear pairing operation execution time	22ms
T_{Ex}	Power operation execution time	16ms
T_{Mu}	Multiplication operation execution time	0.06ms
T_h	Hash operation execution time	0.056ms

(2) Functionality Comparison

There exist multiple blockchain-based privacy-preserving schemes for agricultural-product traceability; representative examples include a zero-knowledge-proof–based model, ProChain (*Li et al., 2024*), a proxy re-encryption (PRE) approach (*Wang et al., 2024*), and an attribute-based encryption (ABE) scheme (*Yang et al., 2024*). In addition, Cai et al and Zeng et al propose efficient group-signature–based authentication frameworks in non-agricultural settings (*Cai et al., 2023; Zeng et al., 2024*). Although developed for different domains, their cryptographic designs are largely transferable and thus informative for our setting.

Table 3

Functionality comparison table

Programme	Privacy-preserving mechanisms	Signature anonymity	Signature traceability Member	Revocation mechanism
Yang et al.,2024	ABE	×	×	×
Wang et al., 2024	PRE	×	√	×
Li et al., 2024	Zero-knowledge proofs	√	√	×
Cai et al., 2023	Group signatures	√	√	×
Zeng et al., 2024	Group signatures	√	√	√
Ours	Group signatures + pseudonyms + revocable accumulators	√	√	√

Table 3 contrasts these studies along key functional dimensions—underlying mechanism, support for signature anonymity and traceability, and the presence of a membership-revocation mechanism. Guided by the characteristics of agricultural supply chains, a scheme is designed that integrates group signatures, pseudonyms, and revocable accumulator, which is better suited to the privacy and identity-management requirements of agricultural-product traceability.

(3) Computational Overhead

To systematically assess the computational efficiency of our scheme, benchmarks are conducted against two representative group-signature–based authentication schemes—*Cai et al. (2023)* and *Zeng et al. (2024)*. Table 4 compares the algorithmic complexity of different schemes in the Signature Generation, Signature Verification, and Membership Revocation phases.

Table 4

Comparison of algorithmic complexity of different schemes

Programme	Signature	Verification
Cai et al., 2023	$11T_{Mu} + T_{bp} + T_{Ex} + 2T_h$	$2T_{Mu} + 4T_{bp} + T_{Ex} + 2T_h$
Zeng et al., 2024	$12T_{Mu} + 2T_{bp} + 2T_{Ex} + T_h$	$9T_{Mu} + 3T_{bp} + 3T_{Ex} + T_h$
Ours	$10T_{Mu} + 2T_{bp} + T_{Ex} + T_h$	$5T_{Mu} + T_{bp} + T_h$

As shown in Table 4, our scheme has lower computational complexity in the Signature Generation phase than *Cai et al. (2023)* and is slightly higher than *Zeng et al. (2024)*. In the Signature Verification phase, our scheme enjoys a clear advantage: it requires only one bilinear pairing and one hash evaluation, thereby reducing the computational load and improving response time. Both our scheme and *Zeng et al. (2024)* support dynamic membership revocation with comparable revocation complexity, whereas *Cai et al. (2023)* does not provide a revocation mechanism, which limits its practicality and flexibility in dynamic settings. Fig. 7 compares the measured average time overheads for Signature Generation, Signature Verification, and Membership Revocation.

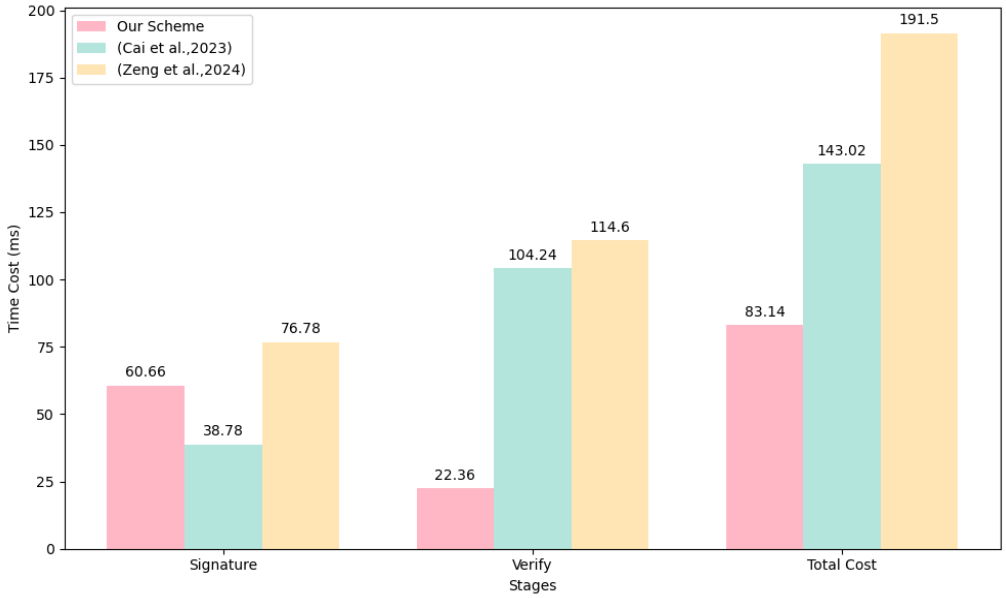


Fig. 7 - Comparison of average time overhead of different group signature schemes in signature and verification phases

As shown in Fig. 7, our scheme achieves an average Signature Generation time of 60.66 ms, lower than Zeng et al. (2024) (76.78 ms) and higher than *Cai et al. (2023)* (38.78 ms). In the Signature Verification phase, our scheme performs best at 22.36 ms, substantially outperforming *Cai et al. (2023)* (104.24 ms) and Zeng et al. (2024) (114.60 ms). For Membership Revocation, our scheme and *Zeng et al. (2024)* exhibit comparable time costs, indicating efficient support for dynamic membership management, whereas Cai et al. (2023) does not implement revocation. Aggregating the three phases, the total measured overhead of our scheme is 83.14 ms, markedly below *Cai et al. (2023)* (143.02 ms) and *Zeng et al. (2024)* (191.50 ms), while maintaining equivalent security guarantees.

(4) Communication Overhead

In blockchain-based agricultural traceability, communication overhead is dominated by signature transmission. Signature sizes are quantified for our scheme and are compared with *Cai et al. (2023)* and *Zeng et al. (2024)*. Results are shown in Table 5.

Table 5

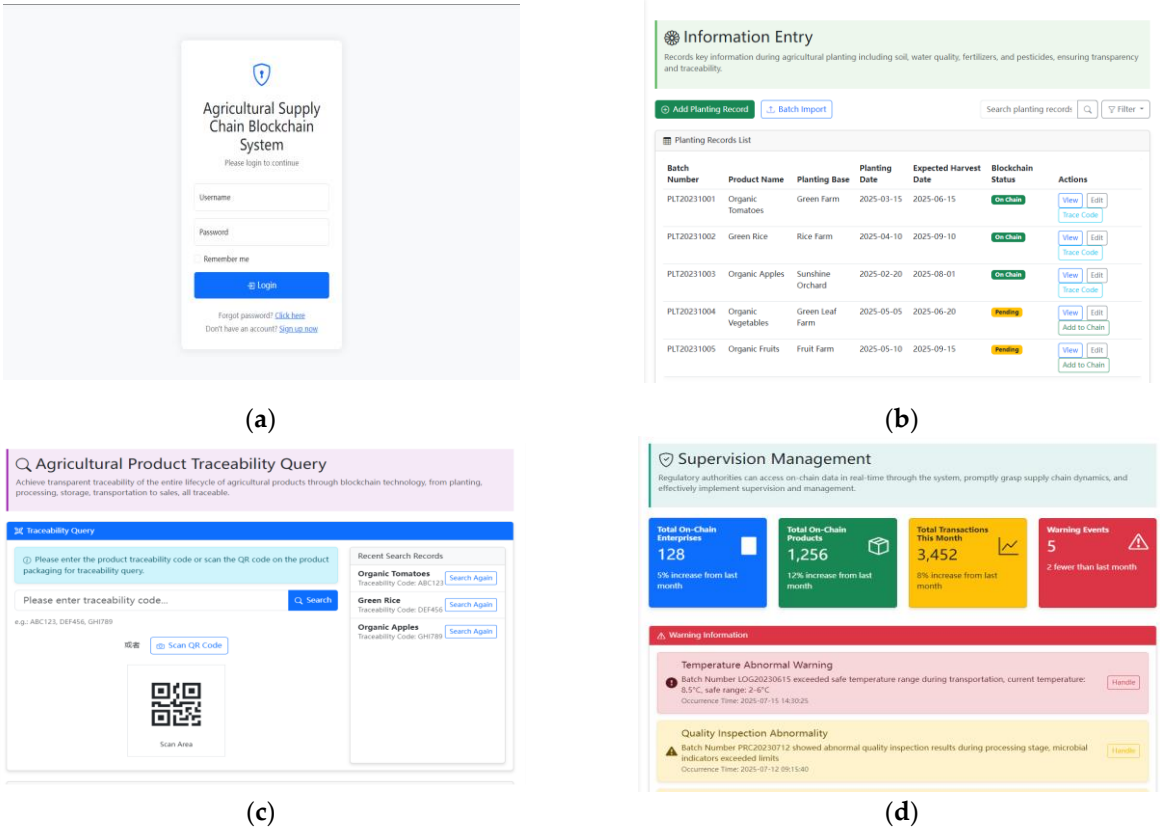
Signature communication overhead comparison table

Programme	Signature size	Byte size
Cai et al., 2023	4|G|+|H|	532B
Zeng et al., 2024	5|G|+|H|+2|Zr|	700B
Ours	4|G|+|H|+2|Zr|	572B

From Table 5, it can be seen that the signature size of this paper's scheme is 572 bytes, 128 bytes less than that of the *Zeng et al. (2024)*, and is more advantageous in transmission efficiency. However, there is a slight increase compared with the *Cai et al. (2023);* overall, it is still in the acceptable range. The difference mainly stems from the additional parameters necessary to introduce a revocable mechanism in the signature structure in this paper. It is worth noting that the scheme in this paper achieves an effective trade-off between security and efficiency by integrating anonymity, traceability and revocation mechanisms in the signature structure, and keeping the communication burden moderate while expanding the functionality. Comprehensively, this paper effectively controls the communication overhead to ensure the integrity of the signature function and demonstrates better practicality and system performance.

(5) Subsubsection

The prototype system, which embodies the proposed group-signature-based blockchain privacy-protection scheme, was developed with the open-source Hyperledger Fabric 2.1 framework. It covers the four pivotal stages of the agricultural supply chain—cultivation, processing, storage and transport, and retail—each corresponding to a distinct stakeholder group. The system significantly accelerates consumer queries for traceability information, strengthening trust in agricultural products. Regulatory bodies gain real-time access to on-chain data, enabling them to monitor supply-chain dynamics and enforce oversight effectively. Agricultural enterprises, meanwhile, can exploit the scheme's authorisation and privacy mechanisms to achieve secure, efficient data exchange and sharing, thus promoting the digitalisation and intelligent upgrading of the sector. The implementation and user interface of the system are illustrated in Fig. 8.



(a) (b) (c) (d)

Fig. 8 - Selected system function interfaces

(a) login page (b) enterprise information entry (c) consumer traceability query and verification (d) regulatory authority maintenance

Figure 8(a) shows the system's login interface: enterprise users and regulatory authorities sign in with account credentials, whereas consumers can proceed directly to the traceability query module. Figure 8(b) depicts the enterprise data-entry interface, where stakeholders at each supply-chain stage upload traceability data to guarantee the authenticity and traceability of product information. Figure 8(c) presents the consumer query interface, enabling users to retrieve product traceability details by scanning a QR code or entering a traceability code. Figure 8(d) illustrates the supervisory-authority management interface, through which regulators monitor and manage traceability data across the entire process, ensuring secure and stable system operation.

CONCLUSIONS

This study addresses the critical issues of privacy leakage, inflexible identity management, and inadequate auditability in agricultural-product traceability systems by introducing a blockchain-based privacy-protection scheme grounded in group signatures. The proposed design integrates a revocable accumulator, pseudonym mapping, and dynamic membership management to maintain signature anonymity while ensuring regulatory traceability and identity consistency amid frequent node changes. Six polynomial-time algorithms and a corresponding security-property model collectively support user registration, signing, verification, de-anonymisation, and revocation, forming a comprehensive privacy-enhanced authentication framework for traceability. Formal analysis confirms that the scheme guarantees anonymity, signature unforgeability, traceability, and forward and backwards privacy. Experimental results demonstrate superior computational efficiency, lower communication overhead, and better blockchain performance than several existing benchmarks—most notably in verification latency and revocation responsiveness. The scheme offers a viable technical route and theoretical foundation for secure, accountable data sharing in multi-stakeholder agricultural supply chains.

Although the prototype performs well in small- to medium-scale networks, the communication complexity and latency of the classical PBFT consensus protocol become bottlenecks as network size grows or conditions fluctuate. Future work will focus on tailored optimisations of PBFT, specifically: (i) compressing communication costs through threshold signatures and batch verification; (ii) employing hierarchical or partitioned consensus to enable parallel execution and reduce overall complexity; and (iii) designing adaptive view-change mechanisms that leverage latency monitoring and reputation evaluation to swiftly exclude faulty nodes, thereby enhancing scalability and robustness in large, heterogeneous networks.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all authors for their support and contributions to this manuscript. This research is supported by the Shanxi Province Basic Research Program for Young Scientists (NO. 202303021222039).

REFERENCES

[1] Bellare, Namprempre, Pointcheval, & Semanko. (2003). The one-more-RSA-inversion problems and the security of Chaum's blind signature scheme. *Journal of Cryptology*,*16*(3), 185-215.

[2] Bermeo-Almeida, O., Cardenas-Rodriguez, M., Samaniego-Cobo, T., Ferruzola-Gómez, E., Cabezas-Cabezas, R., & Bazán-Vera, W. (2018). Blockchain in agriculture: A systematic literature review. In *International conference on technologies and innovation.* pp. 44-56. Springer, Cham.

[3] Boneh, D., & Franklin, M. (2001, August). Identity-based encryption from the Weil pairing. In *Annual international cryptology conference.* pp. 213-229. Berlin, Heidelberg: Springer Berlin Heidelberg.

[4] Cai, J., Tao, X., & Wang, C. (2023). Cooperative authentication scheme for heterogeneous networks based on identity group signature and blockchain. *IEEE Transactions on Vehicular Technology*, *73*(1), 1394-1399.

[5] Demestichas, K., Peppes, N., Alexakis, T., & Adamopoulou, E. (2020). Blockchain in agriculture traceability systems: A review. *Applied Sciences* ,*10*(12), 4113.

[6] Gong, Q., Zhang, J., Wei, Z., Wang, X., Zhang, X., Yan, X., Liu, Y & Dong, L. (2024). Sdacs: Blockchain-based secure and dynamic access control scheme for internet of things. *Sensors*, *24*(7), 2267.

[7] Li, J., Wang, Z., Guan, S., & Cao, Y. (2024). ProChain: A privacy-preserving blockchain-based supply chain traceability system model. *Computers & Industrial Engineering*, *187*, 109831.

[8] Li, L., Tian, P., Dai, J., & Miao, F. (2024). Design of agricultural product traceability system based on blockchain and RFID. *Scientific Reports*, *14*(1), 23599.

[9] Peng, X., Zhang, X., Wang, X., Li, H., Xu, J., & Zhao, Z. (2022). Construction of rice supply chain supervision model driven by blockchain smart contract. *Scientific Reports*, *12*(1), 20984.

[10] Sharma, M. G. (2023). Supply chain, geographical indicator and blockchain: provenance model for commodity. *International Journal of Productivity and Performance Management*, *72*(1), 92-108.

[11] Soy, A., & Balkrishna, S. M. (2025). Blockchain Integration in Agriculture for Transparent Farm-to-Fork Supply Chains: Leveraging IoT and Decentralized Identity for Enhanced Traceability and Security. In *SHS Web of Conferences* (Vol. 216, p. 01073). EDP Sciences.

[12] Stranieri, S., Riccardi, F., Meuwissen, M. P., & Soregaroli, C. (2021). Exploring the impact of blockchain on the performance of agri-food supply chains. *Food control*, *119*, 107495.

[13] Sun, L., Zhou, D., Liu, D., Tang, J., & Li, Y. (2023). BPDAC: A Blockchain Based and Provenance Enabled Dynamic Access Control Scheme. *IEEE Access*,*11*, 142552-142568.
[14] Wang, L., Peng, C., & Tan, W. (2023). Secure ring signature scheme for privacy-preserving blockchain. *Entropy*, *25*(9), 1334.
[15] Wang, S., Luo, N., Xing, B., Sun, Z., Zhang, H., & Sun, C. (2024). Blockchain-based proxy re-encryption access control method for biological risk privacy protection of agricultural products. *Scientific Reports*, *14*(1), 20048.
[16] Xu, Y., Li, X., Zeng, X., Cao, J., & Jiang, W. (2022). Application of blockchain technology in food safety control: current trends and future prospects. *Critical reviews in food science and nutrition*, *62*(10), 2800-2819.
[17] Yang, S., Li, S., Chen, W., & Zhao, Y. (2024). A redactable blockchain-based data management scheme for agricultural product traceability. *Sensors*, *24*(5), 1667.
[18] Zeng, M., Cui, J., Zhang, Q., Zhong, H., & He, D. (2024). Efficient revocable cross-domain anonymous authentication scheme for IIoT. *IEEE Transactions on Information Forensics and Security*.
[19] Zhang, B., Xu, J., Wang, X., Zhao, Z., Chen, S., & Zhang, X. (2023). Research on the construction of grain food multi-chain blockchain based on zero-knowledge proof. *Foods*, *12*(8), 1600.
[20] Zhang, G., Chen, X., Zhang, L., Feng, B., Guo, X., Ling, J., & Zhang, Y. (2022). STAIBT: Blockchain and CP-ABE empowered secure and trusted agricultural IoT blockchain terminal. *IJIMAI*, *7*(5), 66-75.
[21] Zhang, S., Ye, J., & Li, G. (2020). Research and implementation of blockchain technology scheme for cold chain logistics. *Computer Engineering and Applications*, *56*(3), 19-27.