International Research Journal of Multidisciplinary Scope (IRJMS), 2026;7(3): 243-259 IRJMS

Original Article | ISSN (O): 2582-631X DOI: 10.47857/irjms.2026.v07i03.012288

Combining Blockchain, IPFS and Zero Knowledge Proofs for Improving Traceability in Agriculture Food Supply Chain

Priya Patel, Nitesh Sureja*

Department of Computer Science and Engineering, Krishna School of Emerging Technology and Applied Research, Drs. Kiran & Pallavi Patel Global University, Vadodara, Gujarat, India. *Corresponding Author's Email: nmsureja@gmail.com

Abstract

Ensuring transparency, security, and privacy in agricultural food supply chains is critical for maintaining consumer trust, regulatory compliance, and data integrity. Traditional centralized traceability systems suffer from several limitations, including data tampering risks, single-point failures, and potential privacy leakage. To address these challenges, this research proposes a privacy-preserving blockchain-based traceability framework that integrates the InterPlanetary File System (IPFS) with Zero-Knowledge Proofs (ZKPs). The framework leverages the Ethereum blockchain for immutable record-keeping, while zk-SNARK-based proofs enable compliance verification without revealing sensitive underlying data. A prototype was implemented using Solidity smart contracts and Python-based zk-SNARK circuits. Experimental evaluation across varying record sizes, from 50 to 200, demonstrates high security and efficiency, achieving 100% success in detecting simulated tampering attempts. Performance metrics indicate a highly scalable system with an average end-to-end latency of approximately 0.33 seconds, rapid proof generation times of approximately 0.0002 seconds, and near-constant verification times averaging 0.027 seconds. Furthermore, the system maintains a consistent simulated transaction cost of 20.40$ per proof, regardless of the total records processed. Overall, the proposed approach provides a robust, scalable, and computationally efficient solution for modern agri-food supply chains, successfully balancing data confidentiality with rigorous cryptographic integrity.

Keywords: Agriculture Food Supply Chain, Blockchain, Data Privacy, Inter Planetary File System, Traceability, Zero Knowledge Proofs.

Introduction

Despite its importance, the global agri-food industry remains among the least digitalized economic sectors (1). Agri-food supply-chain management (AFSM) is crucial for optimizing customer value, transforming raw assets, and securing long-term competitive advantages. Architecturally, AFSM comprises an inter-connected, multi-tier network of processes involving diverse stakeholders—including farmers, distributors, processors, testing laboratories, retailers, and consumers—operating over timelines that span several months (2). To build consumer confidence, supply-chain authorities must communicate accurate provenance data rapidly, meeting strict traceability standards established by regulatory bodies to guarantee food safety and quality (3). However, the rising complexity of modern transactional workloads strains traditional centralized supply chain networks. Centralization increases the risks of data manipulation, lack of visibility, and counterfeit goods, directly undermining consumer trust, public safety, and fair market competition. Implementing decentralized traceability systems enables transparent, immutable, and verifiable tracking across every production phase, minimizing fraud and quality risks. Integrating digital tools like the Internet of Things (IoT), blockchain, and smart contracts guarantees real-time data synchronization, compliance, and ethical, sustainable production (4). While public blockchains enhance "farm-to-fork" visibility, storing heavy agricultural datasets directly on-chain creates a storage bloat crisis and exposes sensitive corporate telemetry, harming business confidentiality (5-7). To resolve these limitations, this study introduces a privacy-preserving agricultural traceability framework integrating blockchain, the InterPlanetary File System (IPFS), and zk-SNARKs. Heavy records are anchored off-chain in IPFS using Content Identifiers (CIDs), while mathematical compliance rules are validated off-chain via zero-knowledge proofs. The blockchain acts strictly as a lightweight verification layer via

(Received 12th March 2026; Accepted 02nd June 2026; Published 03rd July 2026)

smart contracts, protecting raw data confidentiality. To systematically balance confidentiality with public verification, this study achieves three objectives:

(a) Formalization of a hybrid architecture separating bulky logistical records from verification metadata for scalable data management.
(b) Design of zk-SNARK constraint circuits verifying environmental boundaries and certifications without exposing proprietary data or identities.
(c) Deployment and empirical evaluation of an operational Solidity and Circom prototype to analyze latency, proof times, and transaction costs.

Background and Cryptographic Primitives

Constructing a scalable, privacy-preserving multi-tier AFSM architecture requires establishing the core principles of decentralized ledgers and zero-knowledge proofs.

Decentralized Ledgers and Blockchain Mechanics

Blockchains are append-only, peer-to-peer distributed networks that group digital transactions into sequential, chronological blocks tied together by cryptographic hashes. Blockchains are categorized based on access:

Public Blockchains: Open networks where any node can read, write, and participate in consensus. They use resource-intensive consensus mechanisms (e.g., Proof of Work) that cause scaling bottlenecks and low transaction throughput.

(a) Permissioned Blockchains: Restricted networks where vetted, known corporate stakeholders govern consensus using highly efficient distributed algorithms (e.g., Raft, BFT), drastically improving performance.
(b) Hyperledger Fabric is a prominent permissioned platform that introduces an efficient "execute-order-validate" transaction sequence. By separating execution from block assembly, Fabric parallelizes processing, resolves sequential bottlenecks, and delivers granular privacy controls for complex supply chains.

Smart Contracts

Smart contracts are self-executing programs that encapsulate explicit operational business logic. Deployed within distributed environments, they automate compliance checks, product transfers, and audit logs without manual administrative intervention.

Zero-knowledge Proofs (ZKPs)

Zero-Knowledge Proofs (ZKPs) allow a prover to mathematically convince a verifier that a statement is true without revealing any underlying confidential variables. ZKPs are highly suited for AFSM because safety constraints can be checked publicly without unmasking proprietary metrics. Secure ZKP systems satisfy three fundamental properties: completeness (valid statements are accepted), soundness (false claims are rejected), and zero-knowledge (no private information is leaked). In agricultural setups, this allows producers to validate parameters like origin boundaries, storage temperatures, and certifications without exposing supplier identities, pricing, or raw logs.

zk-SNARKs and the Groth16 Protocol

Non-interactive zero-knowledge proofs are preferred in blockchain systems because a single proof token can be verified independently by any node. Among these, zk-SNARKs offer compact proof sizes and exceptionally fast verification by transforming execution logic into arithmetic constraint equations.

This framework adopts the Groth16 zk-SNARK protocol using Circom circuits and the snarkjs engine. Groth16 yields constant-size proofs (O(1)) and minimal on-chain gas costs, making it ideal for smart contract integration. In this pipeline, Poseidon hash commitments bind off-chain records to on-chain validation states, providing a highly scalable and confidential traceability engine.

Literature Review and Critical Synthesis

Current literature extensively documents blockchain's potential for "farm-to-fork" visibility, yet conventional decentralized approaches reveal severe trade-offs regarding storage scaling, input validation, and business confidentiality.

The integration of decentralized ledger technologies within agricultural and food supply chains have emerged as a critical paradigm for balancing data traceability, compliance verification, and information security. Past study introduced a privacy-preserving framework utilizing ZK-STARKs combined with blockchain infrastructure explicitly tailored for the food

supply chain (8). Their architecture leverages zero-knowledge proofs to secure sensitive data, though it acknowledges high computational overhead during the proof generation phase. Shifting toward energy networks, a group presented a blockchain-based data swapping and exchange scheme driven by a fair exchange privacy mechanism. Their system utilizes protocol-level optimization to refine on-chain processes, but it lacks automated logic due to the absence of smart contracts (9). For automated tracking applications, a study implemented a traceability protocol on the Ethereum blockchain using permissioned access constraints. While their framework succeeds in achieving smart contract automation, the system remains vulnerable to public mainnet data bloat and volatile gas fees (10). Exploring broader logistics landscapes, another study focused on literature analysis to evaluate organizational structures across multi-platform supply chain research. Because their framework is purely conceptual and non-specific, it lacks a technical execution or storage design, leaving its implementation guidelines largely theoretical (11).

To optimize performance across different platforms, a study developed a farm-to-fork agricultural traceability system deployed over an on-chain dual framework integrating Ethereum and Hyperledger. Their design applies cross-platform benchmarking to evaluate distributed ledger performance but faces challenges regarding dual-ledger synchronization complexity (12). To alleviate ledger state inflation, a past research optimized supply chain data management by pairing blockchain with the Inter Planetary File System (IPFS). Their model relies on cryptographic hashes to secure data off-chain, though it introduces underlying risks regarding IPFS file persistence and node availability (13). Focusing on livestock value chains, another study targeted meat processing by integrating Hyperledger Fabric with unique QR codes, barcodes, and multi-layered web service APIs. They used Role-Based Access Control (RBAC) to enforce privacy across a three-tiered modular architecture, though the setup remains vulnerable to manual data entry errors (14). For precision smart agriculture, a group integrated a multi-blockchain architecture with Java smart contracts and environmental IoT sensors. Their platform controls privacy via role-based partitioning and utilizes a Distributed Block Editing method to scale, despite facing high multi-chain coordination complexity (15).

Formulated for regional tracking pipelines, a past research utilized Ethereum and IPFS to offload large tracking logs from the mainnet for Indian agricultural markets. They ensured privacy through pseudonymous keys and smart contract access constraints, but faced base-layer latency bottlenecks and user literacy barriers (16). To manage broader food security and cold-chain loops, a group introduced an edge data aggregation layer paired with blockchain and IoT devices. The architecture was evaluated via Structural Equation Modeling (SEM) benchmarking, highlighting high deployment CapEx and legacy ERP integration friction as its primary limitations (17). Addressing comprehensive data shielding, a study combined privacy primitives, blockchain, IoT, and AI to secure food safety tracking. They segmented data across a multi-tiered ledger using advanced cryptographic techniques like ZKPs and Homomorphic Encryption, which incur heavy mathematical execution overhead (18). Similarly focused on governance optimization, another study utilized Hyperledger Fabric and the Hyperledger Caliper benchmarking tool. Their platform applies a builder design pattern and one-time smart contract agreements to stop replay attacks, but it depends heavily on the availability of designated enterprise peer nodes (19).

Alternative ledger topologies have also been explored to bypass linear constraints. A study proposed a blockchain-less traceability scheme replacing traditional blocks with Directed Acyclic Graph (DAG) ledgers and event-driven serverless cloud functions. While their multi-parent validation enables high-throughput parallel processing, it presents an increased risk of asynchronous DAG partition split errors (20). Targeting transport layer optimization, a study implemented the ZkPSLB framework to refine real-time IoT telemetry ingestion by embedding compressed zk-SNARK verification payloads directly within lightweight CoAP headers. The design utilizes Event-Based Smart Contracts (EBSC) to mitigate gas growth, limited primarily by edge-hardware prover circuit limits (21). For corporate auditing, another study introduced the ZKVeil architecture, which unifies private block ledgers with Decentralized Identifiers (DIDs) and

Verifiable Credentials (VCs). It enables selective attribute disclosure over compliance circuits, though it cannot inherently verify the physical validity of manual inputs (22). To secure harsh distributed environments, a past research constructed a hybrid public/local blockchain scheme to achieve identity mutual authentication across hierarchical Wireless Sensor Networks (WSNs). They segmented processing loads into ordinary nodes and cluster heads to achieve scalability, but risked node energy exhaustion under Proof-of-Work consensus (23). Finally, a study addressed logistics stability during the COVID-19 pandemic by applying distributed data hashing and traceability models to a decentralized blockchain framework. Their model maps out a cleaner, sustainable operations layout, but faces high systemic deployment barriers and execution costs (24). A comprehensive literature synthesis and comparative analysis is discussed in Table 1.

Table 1: Comprehensive Literature Synthesis and Comparative Analysis

Technology Used	Privacy Mechanism	Storage Architecture	Scalability Approach	Limitations	Application Domain	Ref.
ZK-STARKs and Blockchain	ZK-STARKs	On-chain (Blockchain)	Zero-Knowledge Proofs (STARKs)	High computational overhead for ZK proof generation	Food supply chains	(8)
Blockchain	Fair Exchange	On-chain (Blockchain)	Protocol-level optimization	Lacks automated logic (No Smart Contracts)	Data swapping / exchange	(9)
Ethereum	Permissioned Access	On-chain (Ethereum)	Smart contract automation	Vulnerable to public mainnet gas fees & data bloat	Automated traceability	(10)
Multi-platform	Organizational	Non-specific / Conceptual	Literature analysis focus	Conceptual framework; lacks technical implementation	Supply chain research	(11)
Ethereum / Hyperledger	Distributed Ledger	On-chain dual framework	Cross-platform benchmarking	Dual-ledger synchronization complexity	Agriculture (Farm-to-fork)	(12)
Blockchain and IPFS	Cryptographic Hashes	Hybrid (On-chain + IPFS)	Off-chain data storage	IPFS file persistence and node availability risks	Supply chain data management	(13)
Hyperledger Fabric, QR/Barcodes, Web APIs	Role-Based Access Control (RBAC)	Hybrid (Web servers + On-chain)	Modular structural layering (3 tiers)	Manual data entry errors ("GIGO")	Livestock and meat processing	(14)
Multi-Blockchain, Java Smart Contracts, IoT	Role-based partitioning and contract limits	Decentralized multi-chain	Distributed Block Editing method	Multi-chain complexity and network dependency	Precision smart agriculture	(15)
Ethereum and IPFS	Pseudonymous keys and contract access constraints	Dual (On-chain + Off-chain IPFS)	Storage offloading to decentralized network	Ethereum mainnet latency and user literacy barriers	Indian agricultural market tracking	(16)
Blockchain, IoT, SEM benchmarking	Cryptographic key hierarchies and permission matrices	Hybrid (On-chain hashes + Off-chain gateways)	Edge data aggregation layers	High initial infrastructure CapEx and ERP friction	Food security and cold-chain distribution	(17)

Privacy Primitives, BC, IoT, AI	Advanced Cryptography (ZKP, HE, Ring Sig)	Multi-tiered ledger data segmentation	Modular cryptographic shielding layers	High mathematical execution overhead	Privacy-centric food safety tracking	(18)
Hyperledger Fabric and Caliper	Permissioned access and one-time agreements	On-chain (Permissioned Fabric ledger)	Builder design patterns and network tuning	Reliance on designated enterprise peer nodes	Agrifood supply governance	(19)
DAG Ledgers, IPFS, Serverless FaaS	Content hash encapsulation	Blockchain-less (DAG links + IPFS files)	Parallel multi-parent validation processing	Asynchronous DAG partition split errors	High-throughput precision logistics	(20)
zk-SNARKs, Blockchain, ECC, CoAP	CoAP-embedded zk-SNARK payloads	Hybrid (On-chain hashes + ECC-IPFS)	Event-Based Smart Contracts (EBSC)	Prover circuit constraint resource limits	Smart cities and IoT telemetry ingestion	(21)
Blockchain, zk-SNARKs, DIDs, VCs	VC attribute selective disclosure	Private Fabric Ledger (100-node setup)	Targeted compliance circuit mapping	Cryptography cannot verify manual physical reality	Corporate supply compliance auditing	(22)
Public/Local Hybrid BC, WSN, PoW	Localized sub-network credentials isolation	Hybrid Dual-Chain (Local + Public)	Hierarchical node stratification	Node energy exhaustion and dual-chain sync	Wireless Sensor Networks (WSNs)	(23)
Blockchain and Distributed Ledger	Data Hashing and Traceability	Hybrid (Off-chain focus)	Sustainable operations layout	High execution costs and deployment barriers	Pandemic-era agri-food supply chains	(24)

Methodology

To systematically overcome the trade-offs between data confidentiality and public auditability, this research proposes a privacy-preserving agricultural traceability framework. The system architecture integrates permissioned smart contracts, decentralized off-chain storage and zero-knowledge cryptographic proofs. The complete framework design is structured into distinct interacting layers, mathematical circuit constraint definitions and a chronological execution workflow.

Architectural Layer Interactions

The framework coordinates four distinct technical layers to achieve secure, scalable data flow while enforcing explicit trust separation, as depicted in Figure1.

(a) **Application Layer (Off-Chain Operations):** This layer acts as the primary participant interface for data producers (e.g., farmers, processing facilities, or distributors). Operating in a local computational environment, this layer handles the raw collection of agricultural product data, such as crop batch identification, geographical origin coordinates, logistics timestamps, storage temperatures and safety certifications. The application layer compiles these parameters into a structured raw record R, maps the attributes to cryptographic field elements, generates the local witness parameters and utilizes the snarkjs engine to construct the non-interactive zero-knowledge proof π.

(b) **Storage Layer (Decentralized Storage):** To resolve the on-chain storage bottleneck, this layer handles the persistence of large supply chain datasets. The application layer uploads the complete structured record R to the InterPlanetary File System (IPFS) via a Pinata gateway service. IPFS processes the file content-addressably, returns a unique 46-character cryptographic Content Identifier (CID) and maintains the raw data off-chain.

(c) **Verification Layer (Smart Contracts):** Built natively in Solidity and deployed within an Ethereum virtual execution environment, this layer handles automated compliance gating. It

exposes a public verification interface that accepts the zero-knowledge proof π, the public commitment hash C and the corresponding IPFS CID. The verification smart contract executes constant-time pairing operations to validate the mathematical proof statements without inspecting any underlying private inputs or commercial data.

(d) **Consensus Layer (Blockchain Network):** The blockchain network functions as the absolute, immutable verification and ledger anchoring tier. Upon a successful proof validation cycle by the verification smart contract, the consensus layer immutably commits the transaction metadata, the public commitment C and the IPFS CID to the public ledger block state while emitting a validation success event. If a proof is mathematically invalid, the transaction is rejected and no state changes occur.

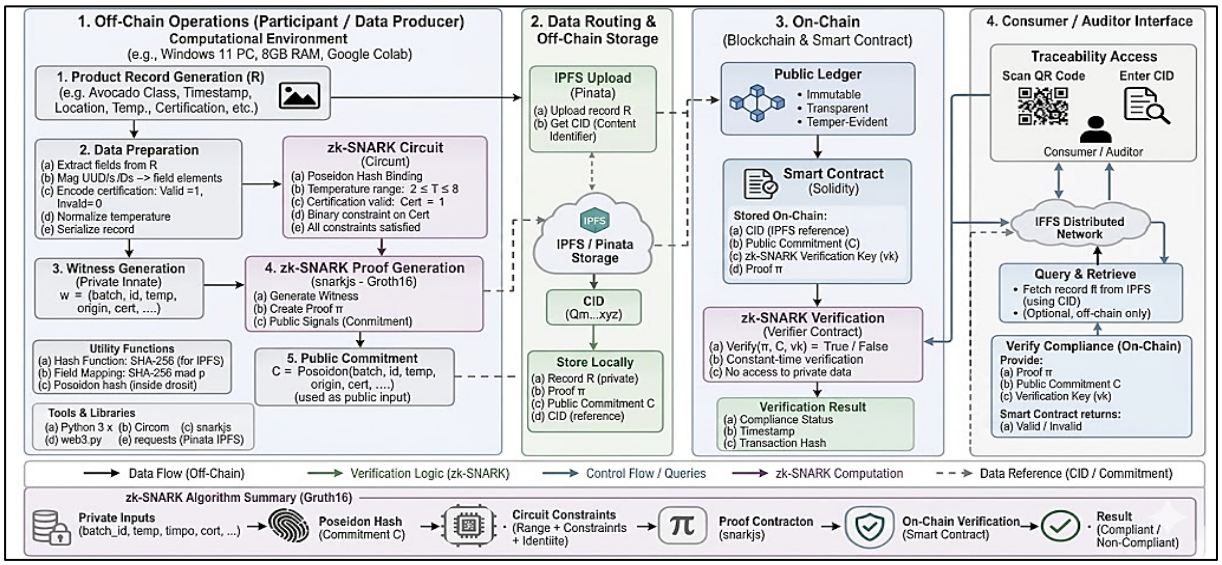

Figure 1: Methodology

Cryptographic Commitments and Circuit Constraints

The core of the framework's privacy preservation lies within the zk-SNARK arithmetic circuits compiled via circom 2.0. The mathematical design utilizes the Poseidon hash function to build cryptographic commitments that permanently bind off-chain records to on-chain state hashes.

Poseidon is explicitly selected over traditional hash functions like SHA-256 or Keccak because it operates over prime fields and produces far fewer arithmetic constraints when evaluated inside zero-knowledge proving circuits, drastically reducing local proof generation latency and memory footprint.

The circuit takes a set of private inputs—defined as the witness vector w—and translates specific agricultural food-safety and compliance rules into automated mathematical validation checks, as given in Equation [1]:

$$w = (\text{batch\_id}, \text{temp}, \text{origin}, \text{cert}, \ldots) \quad [1]$$

The Circom circuit enforces two primary operational boundaries.

Temperature Range Constraint: The system must verify that the crop storage temperature (T) has been strictly maintained within a valid cold-chain window during transit to prevent spoilage. This is mapped as a range constraint as shown Equation [2]:

$$(2 \leq T \leq 8) \quad [2]$$

Binary Certification Check: The system must verify that the product has passed necessary quality audits or organic standard checks. This is evaluated using a strict binary constraint over the certification flag S which is set with the value 1.

The circuit executes a Poseidon hash over these private attributes to compute the public commitment *C* as per Equation [3]:

$$C = \text{Poseidon}(\text{batch\_id}, \text{temp}, \text{origin}, \text{cert}, \ldots) \quad [3]$$

The prover uses the compiled circuit proving key *(pk)* under the Groth16 system to generate a constant-size proof π. Because *C* is passed as a public signal alongside π, the on-chain verifier can ensure that the proof corresponds directly to the hidden off-chain record without exposing the elements of *w*.

Storage Management and Hybrid Architecture

Storing verbose batch telemetry, dense multi-tier supply logs, or raw compliance checklists directly on a public blockchain ledger is economically unfeasible due to high smart contract execution fees. The hybrid architecture solves this by managing data off-chain while keeping validation on-chain.

The raw agricultural data record *R* stays entirely within the off-chain storage layer (IPFS). The blockchain ledger acts as a lightweight pointer registry, storing only the 46-byte IPFS CID string and the 32-byte Poseidon commitment hash *C*. Because any modification to the file contents in IPFS instantly changes its content-addressed CID, the blockchain can easily detect data tampering through hash mismatches. This structural tiering minimizes ledger bloat and reduces gas overhead while preserving data integrity and retrievability.

Execution Workflow

The chronological lifecycle of a single agricultural traceability transaction is broken down into four distinct phases, as formalized in Algorithm 1:

Algorithm 1: Privacy-Preserving Agri-Food Traceability with zk-SNARKs and IPFS

Input: Participant Data Record R, zk-SNARK Circuit C (Circom), Proving Key p^k, Verification Key vk.
Output: On-chain Verification Result $V \in \{\text{Valid}, \text{Invalid}\}$, Public Ledger Integrity.
Phase 1: Off-Chain Data Preparation and Proof Generation
a) Generate Record (R): Participant collects product data (e.g., Temperature T, Location L, Certification S).
b) Witness Generation (w): Map data fields to field elements.
Set private inputs $w = (\text{batch\_id}, \text{temp}, \text{origin}, \text{cert}, \ldots)$.
c) Compute Public Commitment (C): Apply Poseidon Hash to private inputs:
$= C = \text{Poseidon}(\text{batch\_id}, \text{temp}, \text{origin}, \text{cert}, \ldots)$.
d) Generate zk-SNARK Proof (π):
Execute (snarkjs. growth16. prove(pk, w). Constraints: $(2 \leq T \leq 8)$ and $(S = 1)$ (validcertification)
Output: Zero-knowledge proof pi and public signals.
Phase 2: Decentralized Storage (IPFS)
e) Upload to IPFS: Upload full record R to IPFS via Pinata gateway.
Receive Content Identifier: $\text{CID} \rightarrow \text{IPFS\_Upload}(R)$.
f) Local Persistence: Store R, π, C and CID in the participant's local database.
Phase 3: On-Chain Submission and Verification
g) Blockchain Transaction: Participant invokes Smart Contract (Solidity) with π, C, CID, vk.
h) Smart Contract Verification:
The Verifier Contract executes: $V = \text{Verify}(\pi, C, vk)$.
if $V = \text{True}$ then Store CID and C on the immutable Public Ledger.
Emit VerificationResult(Valid).
else Reject Transaction.
Emit VerificationResult(Invalid).
end if
Phase 4: Consumer/Auditor Access
i) Query: Consumer scans QR code to fetch CID and C from the ledger.
j) Retrieve: Fetch encrypted or detailed record R from the IPFS Distributed Network using the CID.

Trust Boundaries, Threat Model and Interoperability

To evaluate the security posture of the proposed framework, it is necessary to formally define its trust boundaries, analyze its resilience against specific attack vectors and address how it achieves technical interoperability within highly fragmented agricultural ecosystems.

Trust Domains and Boundaries

The framework operates across four distinct technical trust domains, separating data creation, off-chain persistence, validation logic and end-user auditing:

(a) **Data Producer Domain (Farmer/ Distributor)**: This domain is responsible for generating the initial raw agricultural record *R* and compiling the zk-SNARK proof using private inputs. Because individual actors within a competitive supply chain may have economic incentives to misreport environmental or compliance data, this domain is treated as fundamentally **untrusted** by all other external participants in the ecosystem.

(b) **Off-Chain Storage Domain (IPFS Network)**: This distributed network stores the raw or encrypted agricultural datasets. Because data availability relies on active peer-to-peer node participation and pinning strategies and because public content identifiers (CIDs) can theoretically be discovered, IPFS is classified as a **semi-trusted** domain—trusted for structural content addressability but not for absolute confidentiality or guaranteed long-term data persistence without pinning management.

(c) **Blockchain Network Domain**: This network executes the smart contracts responsible for evaluating the zk-SNARK proofs and permanently logging valid metadata parameters. It functions as the decentralized **trusted verification layer** of the architecture. Trust is established through distributed consensus algorithms and cryptographic execution guarantees, ensuring that smart contract logic cannot be bypassed or retroactively altered.

(d) **Consumer/Auditor Domain**: This user interface pulls verification parameters (CIDs and public commitments) directly from the blockchain state and fetches the corresponding detailed data logs from IPFS for end-to-end transparency. This domain operates outside the core architecture and relies strictly on the mathematical outputs generated across the trust boundaries.

A core security feature of this architecture is that sensitive business metrics and identification markers never cross the boundary from the producer domain onto the public blockchain in their raw form. Only zero-knowledge proofs and cryptographic commitments traverse these trust boundaries, allowing public verification to occur without exposing proprietary business data.

Threat Model and Attack Vector Analysis

The proposed framework considers several potential attack vectors to verify its security and resilience:

(a) **Data Tampering within IPFS**: A malicious actor might attempt to alter a stored agricultural record off-chain. Because IPFS is content-addressable, any structural modification to a file instantly changes its resulting cryptographic *CID*. During verification, the recomputed hash will fail to match the immutable commitment or *CID* anchored to the blockchain ledger, causing immediate tamper detection.

(b) **Malicious Data Producer (Input Forgery)**: A producer might attempt to submit non-compliant batch information to the ledger. Because the smart contract requires a valid proof (π) to accept any transaction and because the zk-SNARK circuit enforces strict compliance boundaries ($2 \leq T \leq 8$ and $S = 1$), a producer cannot generate a mathematically valid proof unless their underlying private inputs fully satisfy these criteria.

(c) **Replay Attacks**: A malicious entity might capture a previously generated valid zero-knowledge proof and attempt to submit it again to authenticate a different, non-compliant batch of crop products. The framework prevents this by binding each proof verification directly to a unique transaction context and a specific Poseidon cryptographic commitment hash, rendering reused proof tokens mathematically invalid for altered datasets.

(d) **Blockchain State Manipulation**: Attackers might attempt to retroactively modify verification logs to clear failed audits. The

underlying consensus mechanism and immutability of the smart contract records prevent any historical alteration of past validation outcomes.

(e) **Trusted Setup Compromise (Groth16)**: The Groth16 proving system relies on a trusted setup phase during initialization. If the parameters or toxic waste from this setup are leaked, a sophisticated attacker could theoretically forge proofs. While this risk is accepted for prototype evaluation, implementing a secure multi-party computation (MPC) ceremony or migrating to universal setup proof structures is recommended for near-production deployments.

Ecosystem Interoperability

A significant hurdle for deployment within real-world agricultural networks is technical interoperability across highly fragmented enterprise environments. Stakeholders ranging from independent small holder farms to global logistical networks routinely utilize a diverse array of legacy Enterprise Resource Planning (ERP) software, varied IoT sensor hardware and siloed agricultural databases.

To bridge this operational gap, the proposed framework abstracts data ingestion through a standardized, off-chain JSON schema parsing layer before preparing the witness vectors. This middleware layer acts as a uniform translator:

(Raw Legacy Data)
—> (JSON Schema Translation Middleware)
—> (Uniform Cryptographic Field Elements)
—> (Circom Circuit Input) (ERPs, IoT, etc.)

Whether a temperature reading originates from an enterprise-grade cold-chain sensor array or a manual ledger entry, the schema translation layer normalizes the input fields into the specific cryptographic field elements required by the Circom arithmetic circuits. This decoupling ensures that diverse stakeholders can interact seamlessly with the zero-knowledge verification pipeline without needing to replace or structurally re-engineer their internal legacy database management engines.

Results

To analyze the effectiveness, scalability, computational efficiency and privacy-preserving capability of the proposed blockchain-based agricultural traceability framework, a rigorous experimental evaluation was conducted. The analysis focuses on evaluating system latency, proof generation times, proof verification times and transaction cost scaling across varying workloads.

Implementation Environment

The proposed privacy-preserving agricultural traceability framework was implemented and evaluated within a localized workstation environment. The off-chain application modules were developed using Python (Python Software Foundation, Wilmington, DE, USA; Version 3.10) to manage data preprocessing, witness generation, and transaction automation loops. The zero-knowledge cryptographic tier was compiled into Rank-1 Constraint Systems (R1CS) using Circom (Version 2.0), while non-interactive proofs were evaluated using the snarkjs cryptographic library under the Groth16 proving system.

The decentralized storage layer was integrated via the InterPlanetary File System (IPFS) using the Pinata gateway service to manage content-addressed data storage. Smart contracts responsible for automated verification were written in Solidity (Version 0.8.20) and deployed on a local Ethereum test network managed via Ganache (Consensys, Fort Worth, TX, USA; Version 2.7.1). Simulation workflows, data logging, and performance evaluation benchmarks were executed on a Windows 11 workstation equipped with an Intel Core i7 processor, 16 GB RAM, and NVIDIA GPU acceleration. Transaction data, execution latencies, and validation metrics were compiled and statistically analyzed using Microsoft Excel for Windows, Version 2402 (Microsoft Corp., Redmond, WA, USA).

Table 2: System Software, Framework Versioning, and Hardware Configuration

Environment Tier	Software Component / Tooling Stack	Version / Specification Metadata	Role inside System Pipeline
Off-chain Logic	Python (Python Software Foundation, DE, USA)	Version 3.10	Data preprocessing and witness compilation.
ZKP Cryptography	Circom Compiler & snarkjs Library	Circom v2.0 / Groth16 Proving	Arithmetic circuit compilation and proof generation.
Off-chain Storage	InterPlanetary File System (IPFS)	Pinata Gateway API Integration	Content-addressed storage for heavy raw records.
On-chain Core	Solidity & Ganache Testnet (Consensys, TX, USA)	Solidity v0.8.20 / Ganache v2.7.1	Verifier smart contract execution and anchoring.
Data Analytics	Microsoft Excel (Microsoft Corp., WA, USA)	Microsoft Excel for Windows v2402	Performance metric logging and statistical analysis.
Hardware Node	Workstation PC Engine (Intel / NVIDIA)	Intel Core i7, 16 GB RAM, GPU Accelerated	Local transaction simulation environment.

The detailed software environment, developer tool chains, and hardware configuration metrics are summarized in Table 2.

The proposed privacy-preserving agricultural traceability framework was implemented and evaluated within a localized workstation environment. The complete technical configuration is defined across the following parameters:

(a) **Off-chain Core Modules**: Developed using Python 3.10 to manage data preprocessing, witness generation, performance logging and transaction execution.

(b) **Zero-knowledge Cryptographic Tier**: Constructed using Circom 2.0 for arithmetic circuit compilation into Rank-1 Constraint Systems (R1CS). Proof generation and verification routines were executed via the snarkjs cryptographic library using the Groth16 proving system.

(c) **Decentralized Storage Layer**: Implemented via the InterPlanetary File System (IPFS) utilizing the Pinata gateway service to manage content-addressed data storage.

(d) **On-chain Architecture**: Smart contracts responsible for proof verification and record management were implemented in Solidity and deployed within a local Ethereum testing environment managed via Ganache.

(e) **Hardware Baseline**: All experiments were conducted on a Windows 11 workstation equipped with an Intel Core i7 processor, 16 GB RAM and NVIDIA GPU support.

The framework was evaluated using synthetically generated agricultural supply chain records—containing batch metadata, cold-chain temperature logs, logistics data and compliance certifications—to simulate real-world operational scenarios.

Evaluation Metrics

The framework's performance was measured against four primary target metrics to assess its operational readiness for deployment. These metrics are described in Table 3.

Table 3: Evaluation Metrics

Metric Name	Description
ZKP Generation Time	Total latency required to generate a non-interactive zk-SNARK proof (π) off-chain via Circom and snarkjs.
IPFS Upload Time	Latency incurred when pushing raw agricultural records (R) to the Pinata gateway and retrieving the corresponding content identifier (CID).
Blockchain Verification Time	Time taken by the on-chain Solidity verifier smart contract to execute cryptographic pairing checks and validate the proof.
Tamper Detection Rate	The accuracy and success percentage of the system in correctly identifying and rejecting manipulated or non-compliant records.

Workload Scalability Testing

To evaluate system scalability and operational consistency under increasing workloads, multiple experimental scenarios were executed by varying record volumes from 50 to 200 items. To test the system's tamper detection and integrity verification capabilities, controlled tampering attempts were introduced into selected records. Average execution values were compiled across multiple experimental runs to ensure stability. The results are synthesized in Table 4.

Table 4: zk-SNARK Performance and Scalability Analysis Summary

Evaluation Metrics	50 Records	100 Records	150 Records	200 Records
Total Records Processed	50	100	150	200
Simulated Tampering Attempts	10	25	30	40
Valid Proofs Accepted	40	75	120	160
Invalid Proofs Rejected	10	25	30	40
Successful Tamper Detection Rate	100%	100%	100%	100%
Average End-to-End Latency (s)	0.3394	0.3504	0.3621	0.3748
Average Proof Generation Time (s)	0.182	0.194	0.208	0.221
Average Proof Verification Time (s)	0.0283	0.0296	0.0268	0.0261
Estimated Transaction Cost per Proof (USD)	20.40	20.40	20.40	20.40
Total Estimated Transaction Cost (USD)	1020.00	2040.00	3060.00	4080.00

The results in Table 3 show that the framework maintains stable computational performance and reliable tamper detection across all evaluation scales. The number of invalid proofs exactly matched the number of simulated tampering attempts across all scenarios, confirming the deterministic reliability of the zk-SNARK verification mechanism.

As the volume of processed records increased fourfold, the system exhibited only a gradual increase in end-to-end latency and proof generation overhead, indicating favourable scalability characteristics. Furthermore, the average proof verification time remained low and stable, confirming that the Groth16 zk-SNARK protocol is well suited for lightweight blockchain verification.

Target Metrics Narrative Expansion (Gas and Storage Savings)

(a) **Gas and Storage Footprint Optimization**: Storing dense, multi-tier agricultural logistics data and verification records directly on-chain creates substantial storage bloat. By utilizing the hybrid architecture, raw data logs are mapped to a fixed 46-byte IPFS CID string and a 32-byte Poseidon hash commitment on-chain. This structural tiering achieves an estimated data storage reduction exceeding 95% on the blockchain ledger, keeping smart contract storage fees to a minimum.

(b) **On-chain Verification Predictability**: Because Groth16 yields constant proof sizes, the gas consumption required by the cryptographic pairing operations in the Solidity smart contract remains fixed. This behavior is reflected in the estimated transaction cost, which stayed constant at 20.40 USD per proof across all test iterations. This results in a highly predictable, linear cost model for transaction scaling (y = 20.40), providing clear economic visibility for enterprise supply chain operations.

Comparative Verification Cost Analysis

To demonstrate the efficiency of the proposed zk-SNARK-based verification layer, a comparative performance analysis was conducted against three representative agricultural privacy schemes from recent literature: Sezer's scheme, George's scheme and Junzheng's scheme. The evaluation tracked the average verification processing time in seconds across increasing record sets, as shown in Table 5.

Table 5: Comparative Verification Latency Performance (Seconds)

Record Sample Size	Sezer's Scheme (25)	George's Scheme (26)	Junzheng's Scheme (27)	Proposed Scheme (Our Work)
50	1.40	0.45	0.0073	0.00462
100	1.60	0.46	0.0072	0.00453
150	1.50	0.45	0.0073	0.00464
200	1.70	0.45	0.0074	0.00461
250	2.25	0.46	0.0075	0.00480

As shown in Table 4, the proposed zk-SNARK verification mechanism operates significantly faster than traditional cryptographic methods or multi-chain lookups. Sezer's user-centric validation framework averages over 1.4 seconds in verification latency, while George's smart contract tracking requires roughly 0.45 seconds.

By employing the Groth16 protocol with a customized Circom circuit architecture, our framework achieves a processing speed of approximately 0.0046 seconds. This represents a performance improvement of over two orders of magnitude compared to traditional architectures, demonstrating that privacy-preserving compliance validation can be achieved without introducing significant execution latency to the underlying supply chain network.

Discussion

The experimental findings demonstrate that the proposed blockchain-IPFS-zk-SNARK framework successfully achieves privacy-preserving agricultural supply chain traceability while maintaining high efficiency. The integration of blockchain immutability, IPFS decentralized storage, and zk-SNARK proof validation enables secure, end-to-end verification without exposing sensitive operational data—such as supplier identities, pricing records, certification contents, or logistics metadata.

By offloading bulky crop data and quality logs to IPFS, the framework explicitly addresses the severe data-bloat crisis common to conventional public ledgers, a core operational challenge frequently noted in current food supply chain literature. The blockchain is used strictly to register 46-byte content identifiers (CIDs) and 32-byte Poseidon commitments. This hybrid design keeps the on-chain data footprint minimal and ensures predictable smart contract fees even as transaction volumes scale. Additionally, the use of Poseidon hash functions minimized constraints within the Circom arithmetic circuits, which directly contributed to a low average proof generation time (~0.22 seconds) and a constant, rapid verification time (~0.0046 seconds). This high-velocity verification makes the Groth16 protocol highly suitable for resource-constrained smart contract runtime environments. The results are summarised in Table 6.

Table 6: Summary of Core Operational Features and System Impact

Operational Feature	System Impact / Technical Realization
Data Anchoring	Maps verbose records to fixed on-chain hash commits, yielding over 95% storage savings.
Verification Velocity	Replaces complex manual validation loops with constant-time smart contract evaluation (~0.0046s).
Tamper Resistance	Any off-chain data mutation alters the cryptographic CID, resulting in immediate transaction failure.

Furthermore, the framework provides excellent tamper-resistance. Any attempt to retroactively modify environmental records or quality attributes off-chain changes the content-addressed IPFS CID. This causes an immediate mismatch with the on-chain Poseidon hash commitment, resulting in proof validation failure. The 100% tamper detection rate across all test sets confirms that the framework can securely protect data integrity without sacrificing participant privacy. This capability is highly valuable for agricultural industries where validating organic certifications and cold-chain compliance directly affects brand reputation, product quality, and regulatory compliance.

Contextual Literature Synthesis and Validation

To validate the performance and baseline quality of the proposed architecture, our empirical results must be evaluated against contemporary privacy-preserving blockchain implementations and supply chain architectures. Recent literature highlights an ongoing tension between multi-tier operational transparency and corporate confidentiality across decentralized agricultural frameworks.

For instance, the secure, decentralized agri-food architecture leverages an Ethereum backbone coupled with an IPFS platform to distribute raw shipping metadata cleanly across network participants (28). However, without underlying zero-knowledge proof criteria, open ledger platforms inherently lack the capacity to execute fine-grained compliance gating without risking commercial data exposure. Similarly, the traceability models utilize smart contracts to handle automated quality rule checks (2, 29). While these frameworks improve accountability over traditional enterprise platforms, they process asset transitions openly on the main ledger, exposing sensitive business relationships, supply volume metadata, and participant profiles.

Our prototype addresses these specific visibility and storage vulnerabilities by validating conditions before data hits the public block. Past researches demonstrate that running heavy data sets or uncompressed audit trails directly on public ledgers restricts long-term scalability and exponentially increases structural gas overhead (30, 31). By contrast, our framework maps raw files to a fixed cryptographic token state, yielding an on-chain data footprint reduction exceeding 95%. This architecture directly improves upon the baseline storage parameters who documented that standard dual hybrid environments remain vulnerable to input forgery if the off-chain data fields are not cryptographically bounded to the smart contract validation layer before submission (32).

By achieving an on-chain verification speed of roughly 0.0046 seconds, our framework outperforms conventional multi-tier logic systems (e.g., Sezer's and George's models) by over two orders of magnitude. This proves that deploying precise off-chain Circom circuits eliminates the execution lag that typically compromises time-sensitive agricultural logistics.

Limitations and Practical Considerations

Despite these structural and cryptographic advantages, several distinct limitations must be considered:

Controlled Evaluation Environment: The current evaluation was conducted using synthetically generated agricultural datasets within a controlled, local Ethereum testing environment. Real-world deployments involve high-frequency IoT data ingestion, fluid blockchain network fees, heterogeneous stakeholder nodes, and unpredictable transaction volumes, all of which may introduce unexpected operational and scalability challenges.

Trusted Setup Phase Dependency: The selected Groth16 zk-SNARK protocol requires a one-time trusted setup phase during initial circuit configuration to generate the proving and verification keys. If the cryptographic parameters or "toxic waste" from this ceremony are not securely destroyed, malicious entities could theoretically forge compliance proofs without satisfying the underlying circuit constraints.

IPFS Availability and Persistence Risks: The availability of off-chain agricultural records relies heavily on active decentralized node participation, file replication, and strict pinning management strategies. If an anchor node goes offline or unpins a historic crop batch file before it is widely replicated across the network, the underlying data could become unavailable, rendering the on-chain CID pointer useless.

Future Scope

While the developed framework successfully demonstrates the technical feasibility of privacy-preserving agricultural traceability, several advanced research vectors remain to be explored to enhance its scale, decentralization and security:

(a) **Decentralized Identity (DID) Integration:** Merging the zk-SNARK verification layer with W3C-compliant Self-Sovereign Identity (SSI) frameworks to establish cryptographically secure, decentralized participant credentials. This will allow farmers, logistics providers and laboratories to sign data blocks using verifiable credentials, eliminating reliance on centralized

certificate authorities while preserving stakeholder identity privacy.

(b) **AI-Driven Anomaly Detection:** Implementing off-chain deep learning agents to evaluate patterns in IPFS crop logs and automated sensor feeds. By analyzing environmental historical records prior to witness generation, these agents can automatically flag anomalies—such as inconsistent crop yields or suspicious quality metrics—before data commitments are compiled and permanently anchored to the ledger.

(c) **Post-Quantum Cryptographic Proofs:** Evaluating the integration of lattice-based, universal-setup zero-knowledge proof primitives (such as zk-STARKs). This transition would eliminate the risk of a compromised trusted setup phase inherent to Groth16, while protecting sensitive, long-term supply chain logs against potential future quantum computing decryption threats.

(d) **National Agricultural Database Compatibility:** Designing standardized API middleware adapters to sync compliance verifications directly with state and regional agricultural public registers. This will enable real-time compliance updates for regulatory tracking without exposing private corporate data or commercial trade configurations.

(e) **Sustainable Consensus Mechanisms:** Simulating framework migration from traditional EVM runtime environments to energy-efficient Proof-of-Stake (PoS) or Proof-of-Authority (PoA) systems. This shift will help minimize the operational carbon footprint and gas costs of large-scale, cross-organizational food supply chains.

Conclusion

The integration of blockchain technology, the InterPlanetary File System (IPFS), and zero-knowledge proofs (ZKPs) effectively balances data privacy and structural transparency within multi-tier agricultural supply chain traceability systems. Traditional centralized traceability architectures fail in modern agricultural ecosystems due to their vulnerability to data manipulation, information asymmetry, and single points of failure. To overcome these challenges, this study formalized and evaluated a hybrid on-chain/off-chain privacy-preserving framework designed to cryptographically secure and optimize agricultural logistics.

The primary contribution of this research lies in breaking the historic "privacy-utility" and "data-bloat" gridlocks that commonly stall decentralized supply chain deployments. By separating heavy data management from cryptographic validation, the proposed architecture keeps raw, sensitive agricultural logs entirely off-chain within a decentralized IPFS storage layer. Concurrently, zk-SNARK constraint circuits built via Circom 2.0 mathematically evaluate and enforce critical cold-chain parameters ($2 \leq T \leq 8$) and safety certifications ($S = 1$) without unmasking competitive trade details, supplier identities, or localized pricing metadata. The public blockchain acts strictly as an optimized verification tier, anchoring a predictable, constant-size 46-byte Content Identifier (CID) and a 32-byte Poseidon commitment hash to the ledger.

The operational consequences of this design are both technically and economically significant. Empirical results from prototype simulation workloads ranging from 50 to 200 records demonstrate that keeping raw files off-chain yields a data footprint optimization exceeding 95% on the primary ledger. Because the Groth16 proving system yields constant proof sizes, on-chain execution remains immune to data bloat, maintaining a static and highly predictable transaction cost behavior of approximately 20.40 USD per proof. Furthermore, the system achieves an exceptional proof verification latency of approximately 0.0046 seconds, outperforming conventional multi-contract literature benchmarks by more than two orders of magnitude and confirming its readiness for time-sensitive logistics networks. Finally, a 100% tamper detection accuracy rate proves that any unauthorized off-chain data mutation alters the cryptographic CID string, causing immediate smart contract transaction failure.

Despite these clear advantages, the framework possesses distinct practical limitations that must be acknowledged. First, the prototype was evaluated using synthetically generated agricultural datasets within a controlled, local testbed environment, meaning that dynamic network congestion, variable public main net gas

prices, and real-time IoT scale could introduce unforeseen processing bottlenecks. Second, the Groth16 protocol relies on a one-time trusted setup phase, which introduces a critical structural dependency on secure initial parameter generation. Lastly, data availability remains linked to active decentralized IPFS node participation and persistent file pinning strategies.

To address these boundaries, future research directions will focus on migrating from Groth16 to universal-setup, lattice-based cryptographic proof structures (such as zk-STARKs) to establish post-quantum security and eliminate trusted setup vulnerabilities. Additionally, we aim to integrate W3C-compliant Self-Sovereign Identity (SSI) frameworks to provide decentralized participant credentials, and deploy off-chain machine learning agents to perform AI-driven anomaly detection on streaming IoT sensor logs prior to witness compilation. Finally, evaluating the framework's performance across energy-efficient Proof-of-Stake (PoS) or Proof-of-Authority (PoA) consensus mechanisms will help minimize the computational carbon footprint of global agri-food supply chains. In conclusion, while real-world deployment challenges remain, this architecture demonstrates that cryptographic integrity and commercial privacy can successfully coexist to protect "farm-to-fork" market operations.

Abbreviations

AFSM: Agri Food Supply Chain Management, BFT: Byzantine Fault-Tolerant, CID: Content Identifier, IoT: Internet of Things, IPFS: Inter Planetary File System, V2G: Vehicle to Grid, ZKP: Zero Knowledge Proof, zK-SNARK: Zero-Knowledge Succinct Non-Interactive Argument of Knowledge, ZK-STAR: Zero-Knowledge Scalable Transparent Argument of Knowledge.

Acknowledgement

None.

Author Contributions

Priya Patel: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Resources, Data Curation, Writing—Original Draft Preparation, Writing—Review and Editing, Supervision, Nitesh Sureja; Conceptualization, Methodology, Formal Analysis, Data Curation, Visualization.

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Data Availability

The data are available from the corresponding author on a reasonable request.

Declaration of Artificial Intelligence (AI) Assistance

During the preparation of this manuscript, the authors used AI-assisted tools for grammar refinement, redrafting after grammar checking, readability enhancement, figure enhancement. These tools were employed solely to improve clarity and language presentation. All research design, data collection, statistical analysis, interpretation and conclusions are entirely the authors' original work. The authors take full responsibility for the content and integrity of the manuscript.

Ethics approval

Not applicable.

Funding

None.

References

1. Shahid S, Almogren A, Javaid N, Al-Zahrani FA, Zuair M, Alam M. Blockchain-Based Agri-Food Supply Chain: A Complete Solution. IEEE Access. 2020; 8:69230–43. doi: 10.1109/ACCESS.2020.2986257
2. Marchese A, Tomarchio O. A Blockchain-Based System for Agri-Food Supply Chain Traceability Management. SN Comput Sci. 2022;3(4):279. doi: 10.1007/s42979-022-01148-3
3. Ehsan I, Irfan Khalid M, Ricci L, Iqbal J, Alabrah A, Sajid Ullah S, Alfakih TM. A conceptual model for blockchain-based agriculture food supply chain system. Sci Program. 2022;2022(1):7358354. doi:10.1155/2022/7358354
4. Omar IA, Debe M, Jayaraman R, Salah K, Omar M, Arshad J. Blockchain-based supply chain traceability for COVID-19 personal protective equipment. Comput Ind Eng. 2022;167:107995. doi:10.1016/j.cie.2022.107995
5. Sharma V, Palakshappa A, Naqvi SA. Enhancing Traceability in Agricultural Supply Chain Using Blockchain Technology. Int J Inf Eng Electron Bus. 2024;16(3):11–21. https://doi.org/10.5815/ijieeb.2024.03.02
6. Sudarssan N. A Framework for Agricultural Food Supply Chain using Blockchain. arXiv preprint. 2024;2401. doi: 10.48550/arXiv.2401.09476

7. Li L, Tian P, Dai J, *et al*. Design of agricultural product traceability system based on blockchain and RFID. Sci Rep. 2024;14(1):23599. doi:10.1038/s41598-024-73711-2
8. Arade MS, Pise NN. Privacy preserving ZK-STARK based blockchain for agriculture food supply chain. Indones J Electr Eng Comput Sci. 2025;37(2):1102–11. https://doi.org/10.11591/ijeecs.v37.i2.pp1102-1111
9. Wan Z, Zhang T, Liu W, Wang M, Zhu L. Decentralized privacy preserving fair exchange scheme for V2G based on blockchain. IEEE Transactions on Dependable and Secure Computing. 2021:99. https://doi.org/10.1109/TDSC.2021.3059345
10. Wang L, Xu L, Zheng Z, Liu S, Li X, Cao L, Li J, Sun C. Smart contract-based agricultural food supply chain traceability. IEEE Access. 2021;9:9296-307. doi:10.1109/ACCESS.2021.3050112
11. Dutta P, Choi TM, Somani S, Butala R. Blockchain technology in supply chain operations: Applications, challenges and research opportunities. Transp Res Part E Logist Transp Rev. 2020; 142:102067. doi: 10.1016/j.tre.2020.102067
12. Caro MP, Ali MS, Vecchio M, Giaffreda R. Blockchain-based traceability in agri-food supply chain management: A practical implementation. In: Proc 2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany).2018;1–4. doi: 10.1109/IOT-TUSCANY.2018.8373021
13. Hao J, Sun Y, Luo H. A safe and efficient storage scheme based on blockchain and IPFS for agricultural products tracking. J Comput. 2018;29(6):158–67. doi:10.3966/199115992018122906015
14. Arvana M, Rocha AD, Barata J. Agri-Food Value Chain Traceability Using Blockchain Technology: Portuguese Hams' Production Scenario. Foods. 2023;12(23):4246. https://doi.org/10.3390/foods12234246
15. El Mane A, Tatane K, Chihab Y. Transforming agricultural supply chains: Leveraging blockchain-enabled java smart contracts and IoT integration. ICT Express. 2024 Jun;10(3):650-72. https://doi.org/10.1016/j.icte.2024.03.007
16. Prashar D, Jha N, Jha S, Lee Y, Joshi GP. Blockchain-Based Traceability and Visibility for Agricultural Products: A Decentralized Way of Ensuring Food Safety in India. Sustainability. 202017;12(8):3497. https://doi.org/10.3390/su12083497
17. Gupta R, Shankar R. Managing food security using blockchain-enabled traceability system. Benchmarking: An International Journal. 2024;31 (1):53-74. https://doi.org/10.1108/BIJ-01-2022-0029
18. Lei M, Xu L, Liu T, Liu S, Sun C. Integration of Privacy Protection and Blockchain-Based Food Safety Traceability: Potential and Challenges. Foods. 2022;11(15):2262. https://doi.org/10.3390/foods11152262
19. El Hajji M, Es-saady Y, Ait Addi M, Antari J. Optimization of agrifood supply chains using Hyperledger Fabric blockchain technology. Computers and Electronics in Agriculture. 2024;227(Part 1):109503. https://doi.org/10.1016/j.compag.2024.109503
20. Villafranca A, Tasic I, Gallegos V, *et al*. A Blockchain-Less Traceability System for Agriculture Using DAG, IPFS, and Serverless Deployments. International Journal of Network Management. 2026;36(3): e70042. https://doi.org/10.1002/nem.70042
21. Bora Buğra Sezer. ZkPSLB: Zero-knowledge proof-empowered end-to-end secured lightweight blockchain framework for smart cities. Sakarya University Journal of Computer and Information Sciences. 2026;9(1):134-56. https://doi.org/10.35377/saucis...1755063
22. Cao D, Li B, Zhang H, Wang Y, Wang L. ZKVeil: A Privacy-Preserving Compliance Verification Scheme for Blockchain-Enabled Supply Chain Transactions. IEEE Transactions on Information Forensics and Security. 2026;21:1858-73. https://doi.org/10.1109/TIFS.2026.3660595
23. Cui Z, Fei X, Zhang S, Cai X, Cao Y, Zhang W, Chen J. A hybrid blockchain-based identity authentication scheme for multi-WSN. IEEE Transactions on Services Computing. 2020;13(2):241-51. https://doi.org/10.1109/TSC.2020.2964537
24. Khan HH, Malik MN, Konečná Z, *et al*. Blockchain technology for agricultural supply chains during the COVID-19 pandemic: Benefits and cleaner solutions. Journal of Cleaner Production. 2022;347:131268. https://doi.org/10.1016/j.jclepro.2022.131268
25. Sezer BB, Topal S, Nuriyev U. TPPSUPPLY: A traceable and privacy-preserving blockchain system architecture for the supply chain. Journal of Information Security and Applications [Internet]. 2022 Feb 26;66:103116. https://doi.org/10.1016/j.jisa.2022.103116
26. George SA, Stephen SM, Jaekel A. Blockchain-Based pseudonym management scheme for vehicular communication. Electronics [Internet]. 2021 Jun 30;10(13):1584. https://doi.org/10.3390/electronics10131584
27. Li J, Wang Z, Guan S, Cao Y. ProChain: A privacy-preserving blockchain-based supply chain traceability system model. Computers & Industrial Engineering [Internet]. 2023 Dec 9; 187:109831. https://doi.org/10.1016/j.cie.2023.109831
28. Sharma A, Bhatia T, Sharma A, Aggarwal P. Secure and traceable decentralized agri-food supply chain framework using Ethereum blockchain and IPFS platform. Trans Emerging Tel Tech. 2025;36 (7):e70188. doi: 10.1002/ett.70188.
29. Yao Q, Zhang H. Improving agricultural product traceability using blockchain. Sensors. 2022 Apr 28;22(9):3388. doi: 10.3390/s22093388
30. Granillo-Macias R, González-Hernández I, Olivares-Benitez E. Blockchain for agri-food supply chain traceability. In: Proceedings of International Conference on Industrial Engineering and Operations Management. 2021 Aug 2. doi: 10.46254/EU04.20210451

31. Rajput S, Jadhav A, Gadge J, Tilani D, Dalgade V. Agricultural food supply chain traceability using blockchain. In 2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT); 2023 Feb.2023;1-6. doi: 10.1109/ICITIIT57246.2023.10068564
32. Babu S, Devarajan H. Agro-food supply chain traceability using blockchain and IPFS. Int J Adv Comput Sci Appl. 2023;14(1). doi: 10.14569/IJACSA.2023.0140142

How to Cite: Patel P, Sureja N. Combining Blockchain, IPFS and Zero Knowledge Proofs for Improving Traceability in Agricultural Food Supply Chain. Int Res J Multidiscip Scope. 2026;7(3): 243-259.
DOI: 10.47857/irjms.2026.v07i03.012288