blockchains

MDPI

Article

Zero-Knowledge Proof-Based Privacy-Preserving Pharmaceutical Traceability and Recall Using Blockchain

Ankit Sitaula, Md Ashraf Uddin *, John Ayoade, Nam H. Chu and Reza Rafeh

School of Information Technology, Crown Institute of Higher Education, Gungahlin, ACT 2912, Australia; john.ayoade@cihe.edu.au (J.A.); namhoai.chu@cihe.edu.au (N.H.C.); reza.rafeh@cihe.edu.au (R.R.)
* Correspondence: ashraf.uddin@cihe.edu.au

Abstract

Counterfeit and unsafe medicines pose significant risks to patient safety and undermine trust in healthcare systems. This paper presents ACTMeds, a blockchain-supported pharmaceutical traceability and recall platform that considers pharmaceutical supply chain requirements and public health operational needs relevant to the Australian Capital Territory (ACT). The system integrates Ethereum smart contracts, developed using Ganache, with a React-based web application providing regulator, operator, pharmacy, and auditor interfaces, alongside a public verification portal leveraging QR and GS1 barcodes. In addition, role-based access control is enforced across the medicine lifecycle, including manufacture, custody transfer, dispensing, and recall, with immutable on-chain events generated to support auditability and accountability. To balance transparency with confidentiality, the platform prototypes a zero-knowledge (ZK) recall mechanism in which regulators can cryptographically prove that recall conditions meet predefined policy requirements without disclosing sensitive incident details. Threat modeling was conducted using the STRIDE framework, and security evaluation combined static application security testing (Solhint and ESLint) and dynamic testing. The paper further discusses deployment options, cost considerations, ZK recall performance analysis, ethical implications, and future enhancements. Security testing validated the platform's resilience, with no high-severity vulnerabilities identified and medium-severity issues related to HTTP security headers addressed. The results indicate that a regulator-led, privacy-preserving, tamper-evident ledger can improve medicine authenticity verification and recall responsiveness while maintaining compliance and data protection obligations.

Keywords: blockchain; pharmaceutical supply chain; medicine traceability; drug recall management; zero-knowledge proofs; privacy-preserving systems; smart contracts; regulatory technology (RegTech); healthcare information systems; supply chain security

Check for updates

Academic Editors: Chuan Zhang, Licheng Wang, Xiangyun Tang, Jianan Hong and Wenlin Han

Received: 9 March 2026
Revised: 12 May 2026
Accepted: 13 May 2026
Published: 21 May 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Pharmaceutical supply chains are complex, multi-stakeholder ecosystems involving manufacturers, distributors, logistics providers, pharmacies, healthcare institutions, and regulators. Each transfer of custody introduces operational risk, ranging from administrative error and delay to deliberate interference such as counterfeiting, diversion, or falsification. When medicines are compromized due to contamination, temperature excursions, mislabelling, or illicit substitution, regulators must execute recalls rapidly, selectively, and with verifiable accountability. Delays or ambiguities during recall events directly jeopardise patient safety and erode public trust in healthcare systems [1].

Current pharmaceutical traceability and recall mechanisms remain largely fragmented. Most jurisdictions rely on siloed enterprise resource planning (ERP) systems, warehouse management systems (WMS), emailed recall notices, and the manual reconciliation of records across organisations. These approaches introduce latency, create inconsistent versions of the truth, and provide limited guarantees regarding data integrity or provenance. Public verification is typically indirect, requiring patients to rely on hotlines, static notices, or unverified documents, which further weakens trust during safety-critical incidents [2].

While international standards such as GS1 barcoding and serialisation frameworks have improved product identification and visibility, they do not by themselves ensure end-to-end auditability, non-repudiation, or authoritative recall signalling across independent organisations [3]. Recent research has explored blockchain-based solutions for pharmaceutical traceability, highlighting benefits such as tamper-evident logs, decentralized trust, and improved transparency [4]. However, several limitations remain evident in the literature. First, many proposed systems (e.g., [5–7]) emphasise traceability but do not sufficiently address recall authoritativeness, that is, cryptographic assurance that a recall has been legitimately issued by the regulator and acknowledged by downstream actors. Second, transparency is often achieved at the expense of confidentiality, with limited consideration for selective disclosure of sensitive regulatory or incident data. Third, few studies contextualise blockchain-based solutions within realistic public health governance settings, particularly at a regional or jurisdictional level where regulators must balance operational feasibility, privacy obligations, and cost constraints.

Addressing these research gaps requires grounding blockchain solutions within specific jurisdictional regulatory frameworks that can validate both technical feasibility and policy alignment. This paper focuses on the Australian Capital Territory (ACT), where strengthening pharmaceutical traceability and recall coordination aligns with patient safety mandates and community expectations for transparency in public health operations [8].

Two attributes are central to maintaining trust in this context: auditability (i.e., tamper-evident and timestamped records of all lifecycle events) and authoritativeness (i.e., cryptographically verifiable proof that regulatory actions such as recalls originate from the appropriate authority and have been acted upon). However, these attributes are difficult to guarantee since evidence is distributed across private databases, email systems, and heterogeneous access controls.

Blockchain-based traceability can address these auditability-related and authoritativeness challenges through immutable logging and cryptographic verification [9]. However, conventional blockchain implementations often prioritize transparency over confidentiality, creating potential conflicts with healthcare privacy obligations, particularly in recall scenarios where notifications may involve sensitive safety assessments or incident data that should not be disclosed broadly. This motivates the exploration of privacy-preserving mechanisms, such as zero-knowledge (ZK) proofs [10], to reconcile transparency with confidentiality.

To realize these principles in practice, this paper proposes ACTMeds, a blockchain-supported pharmaceutical traceability and recall platform designed for the ACT public health context. ACTMeds encodes medicine lifecycle events and recall authorizations as verifiable state transitions on a shared ledger governed by Ethereum smart contracts. Role-based permissions define who may register batches, transfer custody, dispense products, and issue recalls. A role-based React interface provides operational dashboards for manufacturers, pharmacies, auditors, and the regulator, while a minimal public portal enables patients to verify authenticity and recall status using QR or GS1 barcodes without accessing sensitive operational data.

To address confidentiality concerns while maintaining recall authoritativeness, the ACTMeds platform prototypes a zero-knowledge recall policy mechanism. Under

this model, the regulator proves that predefined recall conditions are satisfied without revealing underlying incident evidence. Downstream stakeholders (e.g., pharmacies, distributors, and dispensers) can independently verify the proof and act with confidence, reducing reliance on informal communication channels while limiting data exposure. The results demonstrate that a privacy-aware, tamper-evident ledger can strengthen medicine authenticity verification and recall responsiveness while supporting compliance and public trust in healthcare supply chains. The major contributions of this paper are summarized as follows:

- Propose a regulator-led blockchain architecture for pharmaceutical traceability and recall management tailored to a real public health jurisdiction.
- Design a role-based smart contract that enforces lifecycle integrity, non-repudiation, and auditable recall acknowledgements.
- Develop a privacy-preserving recall prototype using zero-knowledge proofs to balance transparency with confidentiality.
- Conduct a practical security evaluation of the proposed supply chain and ZK-based recall system using STRIDE threat modelling, static application security testing with Solhint and ESLint, and dynamic security testing.
- Evaluate system performance and discuss deployment and governance insights, including ZK recall execution performance, blockchain transaction costs, cross-chain ZK recall validation across independent Ethereum RPC instances, ethical considerations, and practical pathways to production through permissioned consortium chains or layer-2 infrastructures.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 describes the proposed framework, including its architecture. Section 4 presents the experimental setup and results, followed by a detailed performance evaluation. Section 5 discusses the security objectives, STRIDE-based threat modeling, and a structured risk assessment along with mitigation strategies across the system architecture. Finally, Section 6 concludes the paper and outlines future research directions.

2. Literature Review

With the increasing need for end-to-end pharmaceutical traceability and integrity assurance, blockchain technology has been widely explored as an enabling infrastructure for drug supply chain management. The existing research spans conceptual frameworks, industrial pilots, and regulatory compliance studies. This section reviews state-of-the-art blockchain applications in healthcare and pharmaceutical supply chains, compares representative industry solutions, and identifies gaps addressed by the proposed ACTMeds Pilot.

2.1. Blockchain in Healthcare and Pharmaceutical Systems

Blockchain technology has been applied across various healthcare domains to address fundamental challenges in data management, interoperability, integrity, and trust. For instance, Azaria et al. [5] introduced MedRec, a decentralized record management system using Ethereum smart contracts to handle authentication, confidentiality, accountability, and data sharing for electronic medical records. The system enables patients to control access permissions while maintaining an immutable audit log of all data exchanges. Building on similar principles, Dubovitskaya et al. [11] developed ACTION-EHR, a patient-centric blockchain-based electronic health record management system specifically designed for oncology care networks, demonstrating how blockchain can support specialized clinical domains while preserving patient autonomy over sensitive medical data.

Beyond electronic health records, blockchain has been explored in relation to clinical research data management. Kuo et al. highlighted blockchain's potential to provide

tamper-evident, auditable, and resilient data-sharing mechanisms for next-generation healthcare information systems, particularly in clinical record management and supply chain integrity [12]. Benchoufi et al. [13] implemented a blockchain-based system at Hospital Hôtel Dieu in Paris to enhance the transparency and traceability of clinical trial consent, creating tamper-resistant records of the entire consent collection process accessible to patients and traceable to stakeholders. Maslove et al. [14] presented a proof-of-concept study using blockchain to manage clinical trial data, demonstrating improved data integrity and auditability. More recently, the integration of zero-knowledge proofs with blockchain has emerged as a promising direction for privacy-preserving applications. Zhou et al. [15] provide a comprehensive survey of ZKP-based blockchain identity sharing, identifying key advancements and challenges in achieving selective disclosure without compromising verifiability. Complementing this, Jia et al. [16] propose a multi-level regulation compliance scheme leveraging zk-SNARKs and attribute-based encryption, demonstrating how privacy protection and regulatory oversight can be simultaneously enforced on-chain, a design principle directly relevant to the ZKRecall mechanism proposed in this work.

Within the pharmaceutical domain specifically, blockchain applications have targeted drug authentication, supply chain integrity, and regulatory compliance. For example, Mackey and Nayyar [2] conducted an analysis of online pharmacies and identified blockchain-based track-and-trace systems as potential countermeasures against substandard and falsified medicines in the global pharmaceutical supply chain. Their work emphasizes the role of immutable ledgers in preventing the insertion of counterfeit products at various distribution points. Tseng et al. [17] proposed a governance framework for blockchain-based pharmaceutical supply chains that addresses regulatory requirements across multiple jurisdictions, demonstrating how smart contracts can encode compliance rules for cross-border pharmaceutical trade. Kumar and Tripathi [18] developed a blockchain architecture for secure drug traceability that integrates QR code-based product identification with distributed ledger technology, enabling end users to verify drug authenticity through mobile applications. Collectively, however, these works focus predominantly on data sharing, access control, and provenance verification without addressing the specific challenge of privacy-preserving regulatory recall—a gap that ACTMeds directly targets through its ZKRecall mechanism.

While these pharmaceutical-focused systems establish foundational approaches for provenance verification and anti-counterfeiting, most implementations remain at the conceptual or proof-of-concept stage with limited regulator involvement [19,20]. Furthermore, these systems typically do not address advanced privacy requirements such as zero-knowledge verification for sensitive recall operations or the formal threat modeling of smart contract vulnerabilities [21,22].

More recent research has explored the integration of advanced zero-knowledge proof (ZKP) systems with blockchain to enable privacy-preserving verification. zk-SNARK-based approaches have been widely adopted due to their succinct proofs and efficient on-chain verification [23,24]. Alternative constructions, including zk-STARKs and Bulletproofs, have been proposed to address limitations related to trusted setup and scalability, offering improved transparency and post-quantum resilience [25,26]. Furthermore, frameworks such as Hawk demonstrate how ZKPs can be integrated with smart contracts to enable privacy-preserving computation on blockchain platforms [27]. Despite these advances, existing ZKP-enabled blockchain applications primarily focus on financial privacy, identity management, or transaction confidentiality, with limited exploration of domain-specific applications such as pharmaceutical recall management [28,29].

2.2. Blockchain-Based Pharmaceutical Supply Chain Architectures

Within pharmaceutical supply chains, several architectures have been proposed to address traceability, anti-counterfeiting, and provenance verification [20,22,30]. For instance, Abdallah et al. [31] designed an Ethereum-based solution that integrates IoT sensors with smart contracts to record logistics events and environmental conditions throughout the pharmaceutical supply chain. Their system automates dispatch and delivery verification by capturing real-time data from IoT devices and storing cryptographic hashes on-chain, demonstrating feasibility for end-to-end tracking. However, their evaluation occurs in a simulated environment without the involvement of regulatory authorities, and recall workflows are not explicitly modeled as first-class smart contract operations.

Uddin et al. proposed Medledger, a Hyperledger Fabric-based track and trace system leveraging chaincodes to execute drug supply chain transactions across a permissioned distributed network of pharmaceutical stakeholders [32]. Their system perpetually stores activities and transactions on an immutable ledger linked with peer-to-peer decentralized file systems such as IPFS for maximum transparency and traceability. Sinilarly, Hasan et al. [33] proposed PharmaChain, a Hyperledger Fabric-based architecture This framework combines on-chain transaction records with off-chain data storage and implements role-based access control to secure interactions among manufacturers, distributors, and pharmacies within a national pharmaceutical supply chain context. While PharmaChain enforces access control policies and maintains provenance records, it does not incorporate privacy-preserving cryptographic techniques for sensitive operations, and governance mechanisms for regulatory oversight remain external to the core architecture. Building on these foundational concepts, Chen et al. examined the intersection of blockchain technology and Internet of Things (IoT) devices in pharmaceutical supply chains [6].

The specific application of blockchain to combat pharmaceutical counterfeiting has been explored by Sahoo et al., who proposed a blockchain-based model for eliminating counterfeit drugs [7]. Their approach focused on verification mechanisms that could authenticate pharmaceutical products throughout the supply chain, demonstrating blockchain's potential to address one of the most critical threats to pharmaceutical safety. While these existing solutions provide frameworks for authentication and traceability, they do not fully address the privacy concerns that arise when sensitive pharmaceutical recall information must be coordinated across multiple stakeholders. More recent works have reinforced this gap. Aziz et al. [34] demonstrate that even contemporary blockchain frameworks for combating drug counterfeiting continue to prioritise authentication over privacy-preserving mechanisms. Additionally, Gaynor et al. [35] highlight that recall management remains an insufficiently addressed function in pharmaceutical blockchain deployments, with most systems lacking cryptographically verifiable recall authority.

Additionally, they exhibit several fundamental limitations when examined from a regulatory and privacy perspective.

- Most architectures prioritise transparency and provenance tracking but do not provide mechanisms for cryptographically verifiable recall authority, leaving recall processes dependent on off-chain trust assumptions.
- Privacy considerations are often addressed through access control or off-chain storage rather than formal cryptographic guarantees such as zero-knowledge proofs, limiting their applicability in scenarios involving sensitive regulatory or incident data.
- Regulator involvement is typically peripheral, with governance mechanisms external to the blockchain logic rather than embedded within smart contract workflows.

These limitations highlight a critical gap in existing research, particularly in enabling privacy-preserving, regulator-driven recall coordination within real-world public health systems.

Other works have begun addressing the interoperability challenges arising from heterogeneous blockchain deployments [22]. In [36], Mezquita et al. proposed an interoperability model based on verifiable credentials, which demonstrates how pharmaceutical assets tracked across different blockchain platforms can maintain unified auditable trails without duplicating complete datasets across ledgers. This approach enables pharmaceutical organizations operating on different blockchain networks to exchange verified provenance information while preserving platform independence. Expanding this direction, Triet et al. [37] explored using non-fungible tokens (NFTs) to represent individual drug packages and enable cross-chain custody transfers through specialized smart contracts. These interoperability solutions address technical fragmentation in multi-stakeholder pharmaceutical ecosystems; however, they do not resolve fundamental gaps in privacy protection or regulatory integration observed across existing architectures.

2.3. Industry-Led Pharmaceutical Blockchain Initiatives

In parallel with academic prototypes, several industry-led blockchain initiatives have targeted pharmaceutical supply chains, frequently driven by regulatory mandates such as the US Drug Supply Chain Security Act (DSCSA) and comparable legislation in other jurisdictions [35,38]. The MediLedger network (an industry consortium led by Chronicled and comprising organizations including Pfizer, McKesson, and major wholesale distributors and dispensers) represents a leading permissioned blockchain solution developed specifically for DSCSA compliance [38]. MediLedger's architecture emphasizes privacy-preserving transaction verification through zero-knowledge proofs for specific use cases such as contract reconciliation, enabling parties to verify pricing agreements without exposing complete commercial datasets. The network has been piloted to demonstrate DSCSA compliance for prescription medicine verification, showing measurable benefits in reducing invoice disputes and streamlining chargeback processing. However, as an enterprise-focused closed consortium, MediLedger exhibits limited public transparency and regulator-driven recall authority remains external to the core ledger architecture.

Beyond US-focused initiatives, European solutions demonstrate alternative architectural approaches. PharmaLedger [39] (i.e., a Hyperledger Fabric-based consortium operating primarily in the European Union) focuses on multi-stakeholder medicine authentication and verification across manufacturers, distributors, hospitals, and pharmacies. However, it faces challenges related to high infrastructure complexity and governance coordination across heterogeneous stakeholder groups.

Other industry implementations include Modum.io [40], which integrates IoT sensors with Ethereum for temperature and logistics monitoring throughout cold-chain pharmaceutical distribution, demonstrating real-time environmental tracking capabilities but exhibiting heavy IoT dependency and the limited integration of recall workflows as first-class ledger operations. IBM Food Trust and related pharmaceutical pilots employ Hyperledger Fabric to provide end-to-end traceability using permissioned ledgers, offering strong enterprise integration and access control mechanisms, though these solutions typically position corporate stakeholders as primary governance entities with regulators functioning as external observers rather than active participants in recall orchestration [41]. BlockVerify [42], utilizing public Ethereum smart contracts, targets anti-counterfeit protection through immutable product registration and provenance tracking, providing transparent verification mechanisms accessible to all network participants. However, this approach lacks regulator-driven recall authority and formal threat modeling frameworks that are essential to public health emergency responses [35,43]. Across these industry initiatives, a consistent pattern emerges: privacy mechanisms, where present, are applied to commercial transaction validation rather than to regulatory recall workflows, and regulators are

consistently positioned as external observers rather than authoritative participants in smart contract governance—both of which are central design objectives of the ACTMeds pilot.

2.4. Identified Gaps in Existing Work

Given the above, it can be observed that industry solutions prioritize compliance and operational efficiency over advanced privacy protections and formal security evaluations [35]. Most consortia deploy permissioned ledgers, with participation restricted to vetted organizations, relying on off-chain governance arrangements alongside on-chain rules. Regulators are typically modelled as external observers rather than primary orchestrators of smart contract workflows. While MediLedger documented zk-SNARK privacy mechanisms and Modum.io demonstrated IoT integration, publicly available documentation seldom includes formal threat modeling, systematic smart contract testing, or comprehensive security analysis frameworks [5,43].

Table 1 summarizes prominent blockchain-based pharmaceutical systems and their limitations from a public health regulator's perspective. The main gaps are evident as follows:

- Limited government-led deployments: Most blockchain pilots are driven by corporate consortia rather than public health regulators.
- High capital and operational costs: Enterprise-grade platforms such as Hyperledger Fabric present barriers for regional health authorities.
- Privacy leakage risks: Many systems expose transaction metadata that may reveal commercially or operationally sensitive information.
- Insufficient regulatory recall controls: Existing solutions rarely provide cryptographically verifiable recall authority or acknowledgment workflows.

Table 1. Representative Blockchain-based pharmaceutical supply chain solutions.

System/Study	Technology Stack	Core Features	Limitations/Gaps
MediLedger (US) [38]	Permissioned Ethereum + ZKPs	DSCSA-compliant transaction validation and auditability	Enterprise-focused; closed consortium; limited public transparency
PharmaLedger (EU) [39]	Hyperledger Fabric	Multi-stakeholder medicine authentication and verification	High infrastructure and governance complexity
Modum.io (CH) [40]	IoT Sensors + Ethereum	Temperature and logistics condition monitoring	Heavy IoT dependency; limited recall logic
IBM Food Trust/Pharma Pilots [41]	Hyperledger Fabric	End-to-end traceability using permissioned ledger	Corporate access required; regulator visibility limited
BlockVerify (UK) [42]	Public Ethereum Smart Contracts	Anti-counterfeit verification via immutable records	No regulator-driven recall authority

These limitations restrict the adoption of blockchain solutions in regional public health contexts such as the ACT. Thus, they motivate the development of a regulator-led architecture that integrates role-based smart contract designs for recall management, employs zero-knowledge proofs for confidentiality, and combines formal threat modeling with rigorous testing.

In contrast to existing approaches, the proposed ACTMeds framework introduces a regulator-centric design in which recall authority is embedded directly within the blockchain through role-based smart contracts, enabling cryptographically verifiable recall

issuance and enforcement. Furthermore, unlike prior systems that rely on transparency or access control mechanisms alone, ACTMeds integrates a zero-knowledge proof-based recall mechanism (ZKRecall) to enable selective disclosure, allowing stakeholders to verify compliance with recall policies without accessing sensitive underlying data. This combination of regulator-driven governance, formal privacy guarantees, and recall-specific functionality distinguishes the proposed system from existing blockchain-based pharmaceutical traceability solutions, which predominantly focus on provenance tracking and anti-counterfeiting.

2.5. Contribution of the ACTMeds Pilot

The ACTMeds Pilot directly addresses the identified gaps via the following mechanisms:

- Adopting a regulator-led architecture centered on ACT Health as the authoritative recall issuer.
- Leveraging open-source Ethereum tooling (Solidity, Ganache, and React) to minimise deployment and operational costs.
- Introducing a privacy-preserving recall mechanism (ZKRecall) that allows policy-compliant recalls without disclosing sensitive batch or incident data.
- Providing an implementation-ready yet flexible framework adaptable to local pharmacies, wholesalers, and hospitals.

A comparison of the ACTMeds pilot with other blockchain-based pharmaceutical systems is presented in Table 2.

Table 2. Comparison of blockchain-based pharmaceutical traceability platforms.

Platform	Identity Model	Verifiability	Recall Model	Privacy Approach
MediLedger (US) [38]	Permissioned Ethereum with corporate KYC	High	Distributor-level notifications	ZKPs for transaction validation
PharmaLedger (EU) [39]	Hyperledger Fabric with DIDs	Medium	Smart contract-based recalls	Encrypted channels
BlockVerify (UK) [40]	Public Ethereum EOAs	High	Basic batch flagging	Pseudonymous public ledger
Modum.io (CH) [41]	IoT + Ethereum	High	Environmental trigger-based	Off-chain encrypted storage
Proposed Platform	Role-based smart contract RBAC	High	Regulator-driven ZK recall workflow	Zero-knowledge recall (ZKRecall)

3. The Proposed Supply Chain System

The proposed blockchain-based pharmaceutical supply chain architecture comprises four interconnected layers: the Supply Chain Layer, Blockchain Network Layer, Verification Layer, and Zero-Knowledge Proof Layer. Figure 1 illustrates the complete system architecture and the interactions between its components. Each layer is designed to address specific challenges in pharmaceutical distribution while maintaining seamless integration with adjacent layers.

3.1. Supply Chain Layer

The Supply Chain Layer represents the physical flow of pharmaceutical products through the distribution network, connecting four primary stakeholders in a sequential chain of custody. At the origin, the *manufacturer/importer* registers pharmaceutical batches into the system, assigning each batch a unique identifier along with relevant metadata such as the production date, expiration date, and batch composition. These registered batches then move to the *wholesaler/distributor*, who serves as an intermediary responsible for receiving shipments, verifying batch authenticity, and updating the chain of custody before distributing products to retail endpoints.

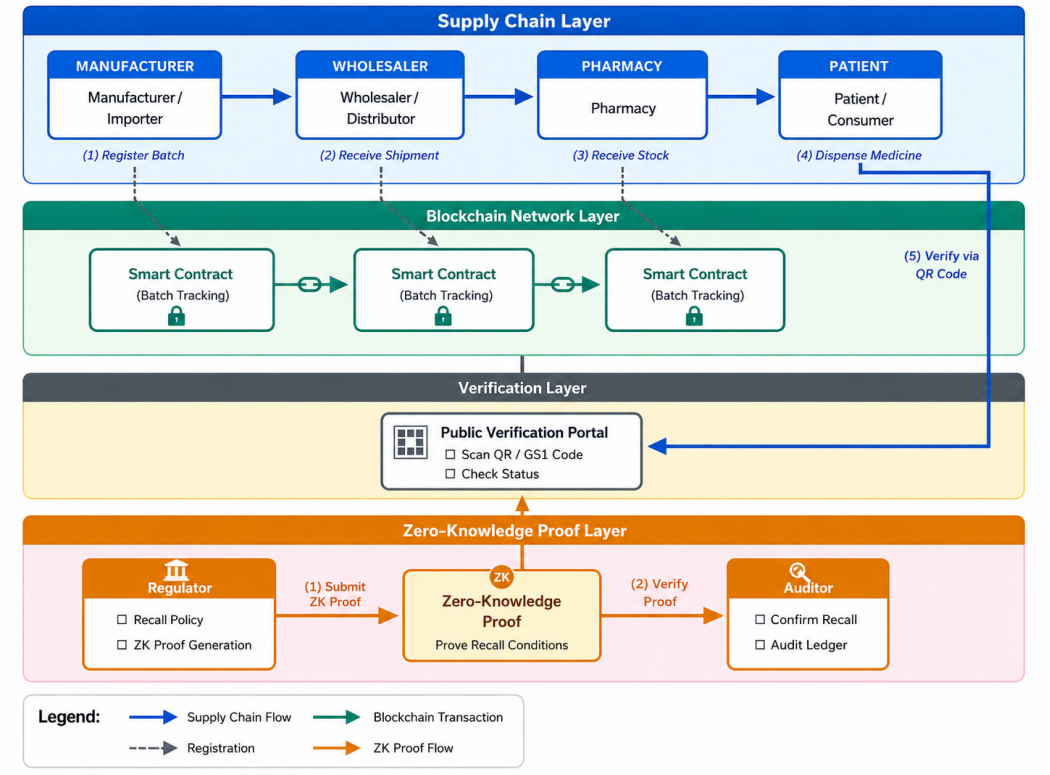


Figure 1. Blockchain-based pharmaceutical supply chain architecture illustrating the four-layer framework: Supply Chain Layer, Blockchain Network Layer, Verification Layer, and Zero-Knowledge Proof Layer.

The *pharmacy* represents the retail layer where stock is received from distributors and medications are dispensed to end consumers. Pharmacies maintain local inventory records that remain synchronized with the blockchain to ensure real-time accuracy. Finally, the *patient/consumer* serves as the terminal point of the supply chain, with the capability to verify product authenticity through QR code scanning before consumption. This end-to-end traceability ensures accountability at every transition point.

3.2. Blockchain Network Layer

The Blockchain Network Layer provides an immutable, distributed ledger for recording all transactions across the supply chain. Smart contracts are deployed at each transition point to automate batch tracking and enforce data integrity through cryptographic verification. These self-executing contracts automatically record ownership transfers, timestamp transactions, and validate compliance with predefined business rules.

The key characteristics of this layer include the following:

- Immutable Audit Trail: Every transaction is cryptographically linked to previous records, creating a tamper-evident history of each pharmaceutical batch from manufacture to dispensation.
- Decentralized Consensus: Transaction validation is distributed across network participants, eliminating single points of failure and reducing the risk of data manipulation.
- Batch-Specific Tracking: Each smart contract maintains comprehensive information including current custodian, location data, environmental conditions during transport, and compliance certifications.

This layer serves as the trusted backbone of the entire architecture, ensuring that all stakeholders operate on a single source of truth regarding product provenance and status.

3.3. Verification Layer

The Verification Layer serves as the public-facing interface that enables stakeholders to authenticate pharmaceutical products without requiring technical expertise in blockchain systems. At the center of this layer is the public verification portal, which abstracts the underlying technical complexity while providing transparent access to critical product information stored on the blockchain.

Users interact with this layer primarily through QR or GS1 code scanning, where standardized codes on pharmaceutical packaging are linked to corresponding blockchain records. Upon scanning, the portal retrieves and displays real-time verification results including the batch authenticity status, recall notifications, and the complete provenance history. The interface is designed for consumer accessibility, allowing patients to verify medication authenticity using standard mobile devices prior to consumption. This democratization of verification capabilities empowers end users to participate actively in pharmaceutical safety without relying solely on institutional gatekeepers.

3.4. Zero-Knowledge Proof Layer

The Zero-Knowledge Proof (ZKP) Layer addresses the inherent tension between regulatory transparency requirements and commercial data privacy. Pharmaceutical companies often possess sensitive information regarding manufacturing processes, distribution networks, and batch compositions that they cannot disclose publicly. Simultaneously, regulators require sufficient oversight to enforce safety standards and manage product recalls effectively.

The ZKP mechanism resolves this conflict in a recall process through a three-component workflow:

1. Proof Generation: Regulatory authorities define recall policies and criteria. When a recall is initiated, the regulator generates zero-knowledge proofs that mathematically demonstrate which batches meet recall conditions without revealing proprietary details or trade secrets.
2. Cryptographic Processing: The ZK Proof Module processes these recall conditions and produces compact mathematical proofs. These proofs can verify that specific conditions are satisfied (e.g., a batch falls within recall parameters) while the underlying data remains encrypted and confidential.
3. Independent Verification: Auditors verify the validity of submitted proofs and confirm regulatory compliance. This verification process validates that safety requirements are met without granting auditors access to confidential manufacturing or distribution data.

This approach preserves the transparency benefits of blockchain technology while respecting the legitimate privacy requirements of pharmaceutical enterprises, creating a compliance framework that is both rigorous and commercially viable.

3.5. Inter-Layer Communication

The architecture employs bidirectional communication channels that ensure coherent data flow across all layers. Supply chain events at the physical layer trigger corresponding smart contract executions in the Blockchain Network Layer, which subsequently update the datasets accessible through the Verification Layer. The Zero-Knowledge Proof Layer interfaces with both the Blockchain Network Layer for generating proofs based on on-chain data and the Verification Layer for publishing verified recall statuses to end users.

This layered design philosophy ensures the separation of concerns—each layer handles distinct responsibilities—while maintaining system cohesion through well-defined interfaces. This results in a modular architecture that can evolve independently at each layer without compromising overall system integrity or data consistency.

3.6. Cross-Chain Scalability and Interoperability Considerations

While the ACTMeds pilot is designed for deployment within a single Ethereum-based ledger governed by ACT Health, real-world pharmaceutical supply chains inherently operate across jurisdictional and organizational boundaries. Manufacturers may operate under federal Therapeutic Goods Administration (TGA) oversight, distributors may span multiple states, and international importers may be subject to foreign regulatory frameworks. Scaling ACTMeds beyond the ACT jurisdiction therefore necessitates engagement with cross-chain interoperability, where pharmaceutical lifecycle events and recall authorizations must be coordinated across heterogeneous blockchain networks without compromising the privacy or integrity guarantees established in the single-chain design.

A key architectural advantage of the ZKRecall mechanism in this context is its potential for proof portability. Because ZK proofs are self-contained cryptographic attestations—asserting that recall conditions are satisfied without revealing underlying evidence—they are not inherently bound to the originating chain. A recall proof generated by ACT Health on the ACTMeds Ethereum network could, in principle, be submitted to and verified by a compatible verifier contract deployed on a separate permissioned chain operated by a federal regulator or an interstate health authority. This design characteristic positions ZKRecall as a candidate primitive for cross-chain recall coordination, where a single authoritative proof propagates recall obligations across multiple ledgers without requiring the disclosure of sensitive incident data to each participating network. This proof portability characteristic aligns with emerging cross-chain trust mechanisms demonstrated in related domains, such as reputation-driven frameworks for cross-chain IoT data sharing [44] and lightweight blockchain protocols for secure collaborative computing [45], where cryptographic verification across heterogeneous networks has been shown to be both feasible and practical.

However, cross-chain extension introduces a distinct set of security and privacy challenges that must be explicitly addressed. First, replay attacks present a significant threat, where a valid recall proof generated for one chain could be maliciously resubmitted to another network to trigger unauthorized recall actions on unrelated batches. Mitigation requires the chain-specific binding of proofs, for example through the inclusion of a chain identifier or contract address within the proof's public inputs. Second, trust anchor reconciliation becomes a critical concern when recall authority must be recognized across chains with different governance models and identity frameworks. Each participating chain must independently verify that the proof originates from a legitimately authorized regulator,

necessitating a shared or federated identity layer. Third, privacy leakage at bridge points poses risks when cross-chain relay mechanisms expose transaction metadata—such as recall timing or affected batch volumes—that could reveal sensitive regulatory intelligence even when the underlying proof remains confidential.

To address these challenges, several architectural pathways are available for the future cross-chain extension of ACTMeds. Verifiable credential frameworks, as demonstrated by Mezquita et al. [36], offer a mechanism for pharmaceutical assets and regulatory authorizations tracked across different blockchain platforms to maintain unified auditable trails without duplicating complete datasets across ledgers. Non-fungible token-based cross-chain custody transfers, explored by Triet et al. [37], provide an alternative model for representing individual drug packages and enabling custody handoffs through specialized bridge contracts. In both cases, the ZK-based privacy model established in ACTMeds provides a compatible foundation, as proof generation and verification can be adapted to operate within verifiable credential schemas or cross-chain messaging protocols. These pathways collectively indicate that the single-chain ACTMeds architecture, while jurisdictionally scoped, is designed with extensibility in mind and can serve as a trust-enabling building block within a broader federated pharmaceutical traceability ecosystem.

4. Implementation and Testing

This section describes the implementation environment of the ACT Health Medicine Supply Chain Blockchain pilot, covering the end-to-end system setup, development tools, and supporting technologies. The pilot realizes a proof-of-concept blockchain platform that supports multi-party batch registration, custody transfer, verification, dispensing, and regulator-led recall across authorized stakeholders in the pharmaceutical supply chain.

Note that the system was designed and implemented as a controlled pilot to evaluate architectural feasibility, security properties, and integration workflows prior to any production-scale deployment. Emphasis was placed on using open-source technologies to ensure transparency, reproducibility, and cost efficiency for public-sector adoption.

4.1. Development Environment

The pilot was developed and tested within a local sandbox environment that emulates an Ethereum-based blockchain network and a web-based client interface. This setup enabled rapid iteration, deterministic testing, and full control over contract deployment and transaction execution without reliance on external infrastructure. Table 3 summarizes the primary tools and technologies used in the development environment.

Table 3. Development environment and tools.

Component	Technology/Tool	Purpose
Blockchain Framework	Truffle Suite with Ganache	Compilation, deployment, and testing of Solidity smart contracts on a local Ethereum network.
Smart Contract Language	Solidity (v0.8.x)	Implementation of role-based access control, batch lifecycle state management, and recall logic.
Frontend Framework	React.js	Development of a role-aware web interface for regulators, operators, pharmacies, and auditors.

Table 3. *Cont.*

Component	Technology/Tool	Purpose
Blockchain Interaction Layer	Web3.js	Client-side interaction with deployed smart contracts, including transaction submission and state queries.
Wallet and Identity Interface	MetaMask	User authentication and cryptographic signing of blockchain transactions via externally owned accounts.
Version Control and CI	GitHub (with GitHub Actions)	Source code management and automated build and test workflows.

The use of Ganache allowed deterministic control over accounts, block times, and transaction ordering, which was particularly useful for testing lifecycle transitions and recall workflows. Table 4 shows the summary of smart contract deployment in terms of gas and ETH. Wallet addresses were mapped to simulated organizational roles to reflect real-world identity onboarding that would be handled off-chain in a production deployment. Overall, this environment provided a realistic yet controlled setting to validate the system architecture, smart contract logic, and user interaction flows before considering deployment on a permissioned consortium network or a production-grade layer-2 solution.

Table 4. Smart contract deployment summary (Ganache).

Contract	Address	Gas Used	Gas Price (Gwei)	Cost (ETH)	Block No.
Migrations	0x6764...14A8	242,032	2.5	0.00060508	149
SupplyChain	0x216A...5d20	5,067,049	2.5	0.01266762	151
ZKRecallGate	0xE24b...1eC0	199,423	2.5	0.00049856	164

4.2. Smart Contract Development

The core blockchain logic of the ACT Health Medicine Supply Chain Pilot is implemented in a primary Solidity contract, namely SupplyChain.sol, which defines participants' roles, batch lifecycle states, and regulator-led recall mechanisms. This contract serves as the authoritative source of truth for all batch-related state transitions and enforces access control through role-aware function modifiers.

Each interaction with the contract produces immutable on-chain events, enabling real-time traceability and post-incident auditing. The contract follows a state machine model to ensure that batch transitions occur only in valid sequences and under authorized roles. The core features of the proposed smart contract are as follows:

- Role Assignment: Authorized participants are assigned roles (i.e., manufacturer, wholesaler, pharmacy, and regulator) via a dedicated role management function (e.g., `assignRole()`) with role membership enforced on-chain.
- Batch Lifecycle Management: Each medicine batch progresses through a predefined lifecycle: `Registered` → `InTransit` → `AtPharmacy` → `Dispensed` → `Recalled`. Invalid state transitions are rejected by the contract to preserve lifecycle integrity.
- Role-Based Access Control (RBAC): Critical functions are protected using role-specific modifiers e.g., `onlyManufacturer, onlyWholesaler, and onlyRegulator,` ensuring that actions such as registration, transfer, dispensing, and recall can only be invoked by authorized entities.
- Immutable Audit Trail: All lifecycle actions emit structured events, including BatchRegistered, BatchTransferred, BatchDispensed, and BatchRecalled. These events

provide a tamper-evident audit log that can be consumed by auditors, monitoring tools, or external analytics systems.

- Regulator-Triggered Recall (i.e., "Sticky Recall"): Once a batch is marked as recalled by the regulator, the recall status persists permanently on-chain. Downstream actors are prevented from further state transitions on recalled batches, ensuring that recall enforcement is consistent and non-reversible.

4.3. Frontend Integration (React and Web3.js)

The frontend application was implemented using React.js and integrated with the blockchain layer via Web3.js and the MetaMask wallet. The frontend acts as the primary interaction layer for stakeholders, abstracting blockchain complexity while preserving cryptographic transaction signing and verification. The application dynamically renders role-specific dashboards based on the connected wallet address and its associated on-chain role, ensuring that users are presented only with operations permitted by their role. This approach improves usability and reduces the risk of unauthorized actions. Table 5 presents the role-based frontend modules and their corresponding operations.

Table 5. Role-based frontend modules and operations.

Role	Frontend Module	Example Operation
Manufacturer	RegisterForm.jsx	Register a new medicine batch on the blockchain.
Wholesaler	TransferForm.jsx	Transfer batch custody to the next authorized party.
Pharmacy	DispenseForm.jsx	Record batch dispensing and update lifecycle state.
Regulator	RecallPanel.jsx	Initiate and verify regulator-authorized recalls.
Public Verifier (Customer)	VerifyBatch.jsx	Verify authenticity and recall status via batch ID or QR/GS1 code.

Dynamic role-based rendering ensures that interface components are automatically enabled or disabled based on the role associated with the connected wallet. All blockchain interactions, including transaction submission and state queries, are executed through the deployed contract ABI, with MetaMask handling cryptographic signing and account management. Figure 2 presents an example of the role-based dashboard view with an active MetaMask wallet connection, while Figures 3 and 4 show the user interfaces for the operator and manufacturer roles, respectively.

4.4. ZKRecall Module (Privacy-Preserving Recalls)

The *ZKRecall* component introduces selective disclosure for recall initiation using zero-knowledge proofs. Instead of publishing incident evidence directly on-chain, the regulator generates a zero-knowledge proof attesting that a predefined recall policy has been satisfied, such as a hazard classification or threshold-based condition. The proof is verified, typically by an on-chain verifier smart contract, after which the recall state of the affected batch is updated on the distributed ledger. Downstream stakeholders are able to verify the proof outcome and take appropriate action without accessing or learning any sensitive incident details.

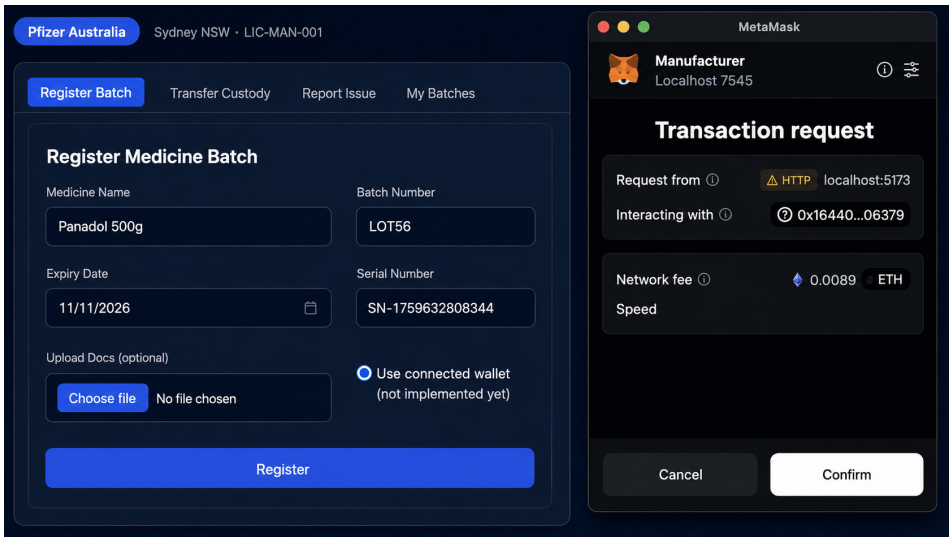


Figure 2. Role-based dashboard view with connected MetaMask wallet.

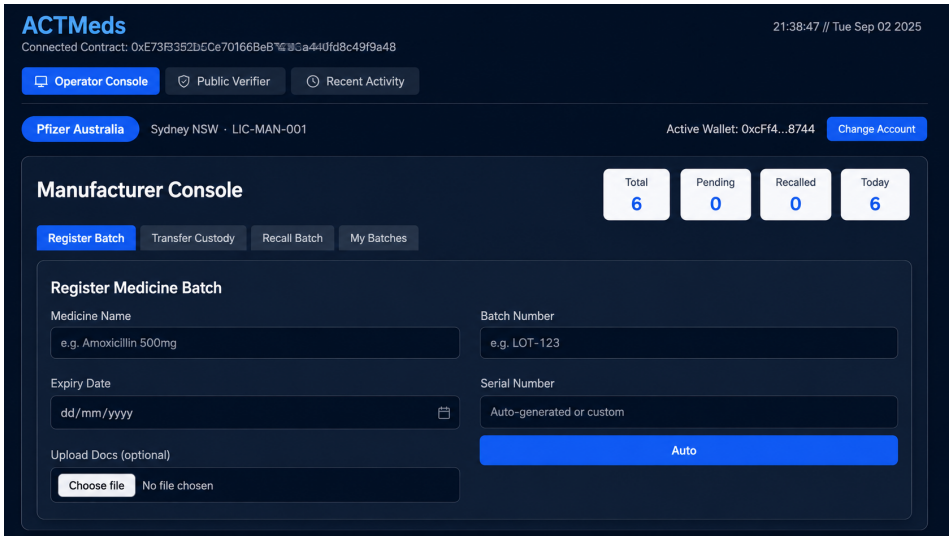


Figure 3. Operator console tabs with wallet connected.

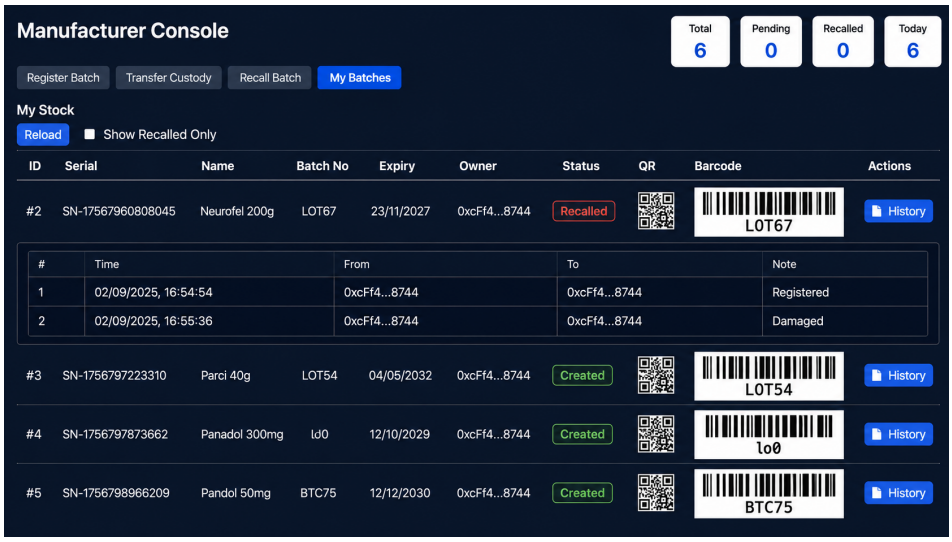


Figure 4. Manufacturer console tabs with wallet connected.

4.4.1. Circuit Design, Proving System, and On-Chain Integration

- Circuit Design (Circom) [46]: A simple ZK circuit uses the Poseidon hash function to ensure that a private secret (regulatory token) matches a public commitment (actionHash), with batchId included as part of the computation. The circuit is compiled into standard zk-SNARK artefacts (R1CS, WASM, and proving key) using Circom over the BN128 field.
- Proving System (Groth16) [47] and On-Chain Verification: Uses Groth16 to generate compact proofs off-chain, each consisting of three elliptic curve points and validating two public inputs: batchId and actionHash. A Solidity verifier contract checks the proof using an efficient pairing equation on BN128 via EVM precompiles (EIP-197), ensuring low gas cost.
- Integration (recallByProof): The smart contract confirms the batch ID matches, recomputes and validates the action hash, verifies the ZK proof, and executes the recall.

4.4.2. Zero-Knowledge Proving System

The proposed system adopts the Groth16 [47] proving scheme over the BN128 elliptic curve. Groth16 generates succinct non-interactive zero-knowledge proofs consisting of three elliptic curve points:

$$\pi_a \in \mathbb{G}_1, \qquad \pi_b \in \mathbb{G}_2, \qquad \pi_c \in \mathbb{G}_1 \tag{1}$$

Thus, a proof can be represented as

$$\pi = (\pi_a, \pi_b, \pi_c) \tag{2}$$

In the proposed system, the proof π is generated off-chain by the regulator using `snarkJS` and the corresponding proving key. The public inputs to the circuit are defined as

$$\mathbf{x} = [batchId, actionHash] \tag{3}$$

where *batchId* identifies the pharmaceutical batch and *actionHash* is computed as the Poseidon hash of the regulator's private authorization secret:

$$actionHash = \text{Poseidon}(authSecret) \tag{4}$$

4.4.3. On-Chain Verification

The on-chain verifier is implemented in `Verifier.sol`, which is automatically generated by `snarkJS`. The contract embeds the verification key as Solidity constants, including the elliptic curve points:

$$vk = (\alpha, \beta, \gamma, \delta, IC) \tag{5}$$

where $\alpha \in \mathbb{G}_1$, $\beta \in \mathbb{G}_2$, $\gamma \in \mathbb{G}_2$, $\delta \in \mathbb{G}_2$, and IC denotes the input commitment points.

The verifier computes the linear combination vk_x from the public inputs and the input commitment array:

$$vk_x = IC_0 + \sum_{i=1}^{n} x_i IC_i \tag{6}$$

For the two public inputs used in this system, this becomes

$$vk_x = IC_0 + batchId \cdot IC_1 + actionHash \cdot IC_2 \tag{7}$$

The Groth16 verification is performed using the following bilinear pairing equation:

$$e(\pi_a, \pi_b) = e(\alpha, \beta) \cdot e(vk_x, \gamma) \cdot e(\pi_c, \delta) \tag{8}$$

where $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is the bilinear pairing operation.

This pairing check is executed entirely within the Ethereum Virtual Machine (EVM) using the `bn256` precompile defined in EIP-197. As a result, the verification remains gas-efficient while allowing the smart contract to confirm that the regulator possesses a valid private authorization secret without revealing it on-chain.

The integration of the Groth16-based verification mechanism into the application layer is realized through the `recallByProof` function in `SupplyChain.sol`, as outlined in Algorithm 1.

Algorithm 1 ZK-based batch recall

Require: $batchId$, proof (a, b, c), public signals $pubSignals$, $reasonHash$, $timestamp$
Ensure: Batch is recalled only if the ZK proof and public inputs are valid

Check batch binding:
if $pubSignals[0] \neq batchId$ **then**
    return "Batch mismatch"
end if
Compute recall action hash:
$actionHash \leftarrow H(\text{"RECALL"} \parallel batchId \parallel reasonHash \parallel timestamp)$
Check action binding:
if $pubSignals[1] \neq actionHash$ **then**
    return "Action mismatch"
end if
Verify Groth16 proof:
$ok \leftarrow VerifyRecallProof(a, b, c, pubSignals)$
if $ok = false$ **then**
    return "Invalid ZK proof"
end if
Execute recall:
$RecallBatch(batchId, \text{"ZK Recall"}, contractAddress)$
Emit $BatchRecalledZK(batchId, reasonHash, timestamp, pubSignals[1])$

4.5. Data Model and On-Chain States

Each medicine batch is represented by a unique identifier (for example, derived from a GS1 or QR encoding scheme) and is associated with batch metadata. Minimal metadata fields are stored on-chain, while extended attributes may be maintained off-chain. The system tracks the following batch state transitions:

- Registered: Batch created by the manufacturer or importer.
- InTransit/Stored: Batch under wholesaler custody.
- AtPharmacy: Batch received and held by a pharmacy.
- Dispensed: Dispense event recorded by the pharmacy.
- Recalled: Regulator-issued recall flag following successful proof verification.

The event logs presented in Figure 5 are emitted for each state transition, providing a chronological and tamper-evident provenance trail to support auditing and incident response.

4.6. Operational Workflow

The end-to-end lifecycle of a medicine batch follows the operational workflow outlined below:

1. Batch registration: The manufacturer or importer registers the batch on-chain and emits a registration event.
2. Custody transfer: The wholesaler receives the shipment, records the custody receipt, updates the batch state, and emits an event.
3. Pharmacy receipt: The pharmacy receives the stock and records the batch as available under pharmacy custody.
4. Dispense: The pharmacy dispenses the medicine and records the dispense event, subject to local policy constraints (batch-level or serial-level).
5. Public verification: A patient scans the QR or GS1 code, and a verification portal retrieves the on-chain state to return authenticity and recall status.

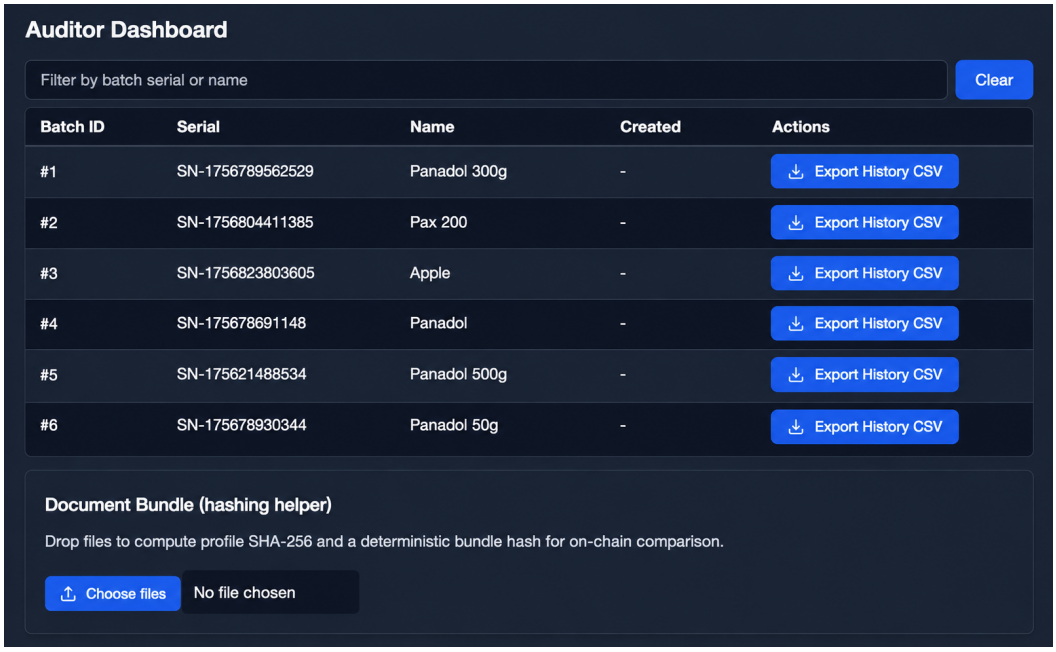


Figure 5. Transfer event log (auditor console).

Algorithms 2 and 3 present pharmaceutical batch registration and distribtuion lifecycle.

4.7. Privacy-Preserving Recall Flow (ZKRecall)

When a recall is required, the regulator initiates a privacy-preserving recall workflow consisting of the following steps:

1. Policy evaluation (off-chain): The regulator evaluates incident evidence against a predefined recall policy encoded as a zero-knowledge circuit.
2. Proof generation: A zero-knowledge proof is generated, attesting that the recall conditions are satisfied without revealing the underlying sensitive evidence.
3. On-chain verification and recall update: The verifier contract validates the proof and, upon successful verification, updates the recall registry by marking the affected batch as recalled.
4. Stakeholder action and acknowledgement: Wholesalers and pharmacies act based on the recall flag and may optionally submit acknowledgements to support auditability.

This design preserves confidentiality while maintaining the cryptographic authoritativeness and verifiability of recall issuance. The ZK implementation and output is shown in Figure 6.

4.8. Performance Analysis

The performance evaluation of the ACTMed's smart contracts is completed on a local Ganache Ethereum node (Chain ID 1337; instant mining mode). The results are derived from 50 live transactions across four supply chain lifecycle operations: batch registration (n = 20), custody transfer (n = 15), pharmacy dispense (n = 10), and regulator recall (n = 5). Latency is measured as wall clock round-trip time; gas values are taken directly from transaction receipts; and throughput (TPS) is computed as

$$\frac{1000}{\text{mean latency per operation}}$$

Additionally, power is estimated as

$$P = 2000\,\text{mW} + \text{gas} \times 0.00025\,\text{mW/gas}$$

Algorithm 2 Pharmaceutical batch registration and distribution lifecycle

Require: Batch data $(serial, batchNo, name, expiry, metadataHash, initialOwner)$
Ensure: Immutable on-chain provenance record for the pharmaceutical batch
procedure REGISTERBATCH($serial, batchNo, name, expiry, metadataHash, initialOwner$)
Require: $isManufacturer[msg.sender] =$ **true**
Require: $idBySerial[serial] = 0$
    $batchId \leftarrow nextId$
    $nextId \leftarrow nextId + 1$
    Initialize batch record b
    $b.status \leftarrow Created$
    $b.history[0] \leftarrow Hop(b.manufacturer,\ initialOwner,\ block.timestamp,\ \text{“}Registered\text{”})$
    Emit `BatchRegistered`
    Emit `OwnershipTransferred`
    Emit `StatusUpdated`
    return $batchId$
end procedure
procedure TRANSFERBATCH($id, to, note$)
Require: $batches[id].currentOwner = msg.sender$
Require: $batches[id].status \neq Recalled$
    $from \leftarrow b.currentOwner$
    $b.currentOwner \leftarrow to$
    if to is a Wholesaler **then**
        $b.status \leftarrow InTransit$
    else if to is a Pharmacy **then**
        $b.status \leftarrow AtPharmacy$
    end if
    Append $Hop(from,\ to,\ block.timestamp,\ note)$ to $b.history$
    Emit `OwnershipTransferred`
    Emit `StatusUpdated`
end procedure
procedure DISPENSEMEDICINE($id, patient, note$)
Require: $isPharmacy[msg.sender] =$ **true**
Require: $batches[id].currentOwner = msg.sender$
    $from \leftarrow b.currentOwner$
    $b.status \leftarrow Dispensed$
    $b.currentOwner \leftarrow patient$
    Append $Hop(from,\ patient,\ block.timestamp,\ note)$ to $b.history$
    Emit `MedicineDispensed`
    Emit `OwnershipTransferred`
    Emit `StatusUpdated`
end procedure

All operations complete within 35 ms at P95, confirming sub-second response times suitable for real-time supply chain workflows.

Figure 7 represents transaction latency and gas usage for register batch, transfer, dispense and recall. Register batch has the widest spread (29–49 ms) with one visible outlier at 49 ms, which is real OS scheduling jitter. Dispense and recall are tightly clustered at 20–22 ms, showing that read-light operations are consistently fast. The boxes are narrow, meaning your local Ganache environment is stable. registerBatch consumes 402,212 gas on average, approximately 2.9× the cost of dispenseMedicine (139,806 gas), reflecting the additional storage and event emission introduced by the ZKRecallGate integration.

Algorithm 3 Pharmaceutical batch recall procedures

Require: Batch identifier id
Ensure: Batch status is updated to *Recalled* and provenance is preserved
procedure RECALLBATCHDIRECT($id, note$)
Require: $isRegulator[msg.sender] =$ **true** $\vee$ $msg.sender = b.manufacturer$
    $holder \leftarrow b.currentOwner$
    $b.status \leftarrow Recalled$
    Append $Hop(msg.sender,\ holder,\ block.timestamp,\ note)$ to $b.history$
    Emit `BatchRecalled`
    Emit `StatusUpdated`
end procedure
procedure RECALLBATCHZK($id, reasonHash, timestamp, a, b, c, pubSignals$)
Require: $address(zkGate) \neq 0$
Require: $pubSignals[0] = id$
Require: $pubSignals[1] =$ `keccak256`$(\text{“RECALL”} \,\|\, id \,\|\, reasonHash \,\|\, timestamp)$
Require: $zkGate$.`verifyRecallProof`$(a, b, c, pubSignals) =$ **true**
    $holder \leftarrow b.currentOwner$
    $b.status \leftarrow Recalled$
    Append $Hop(msg.sender,\ holder,\ block.timestamp,\ \text{“ZKRecall”})$ to $b.history$
    Emit `BatchRecalled`
    Emit `StatusUpdated`
    Emit `BatchRecalledZK`$(reasonHash,\ timestamp,\ pubSignals[1])$ ▷ Recall reason plaintext is never stored on-chain
end procedure

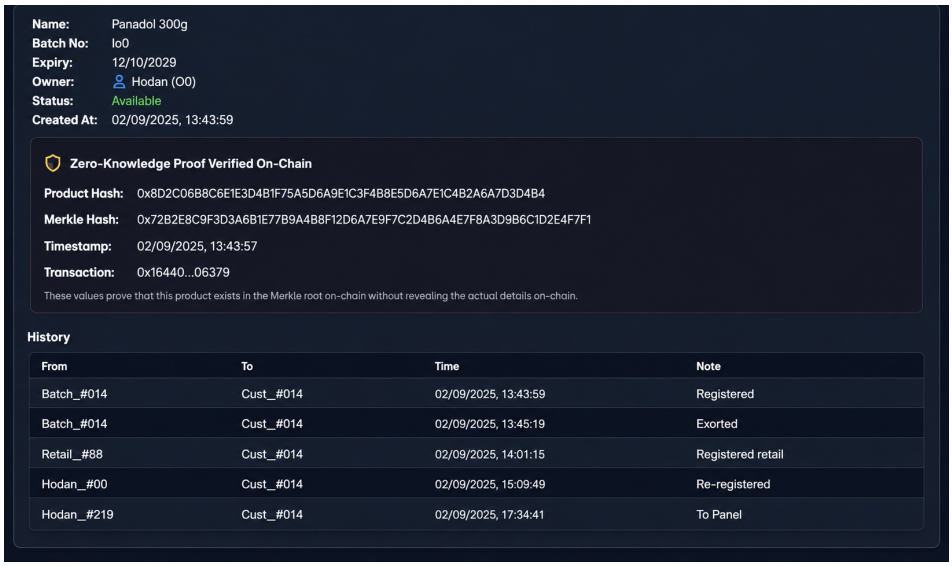
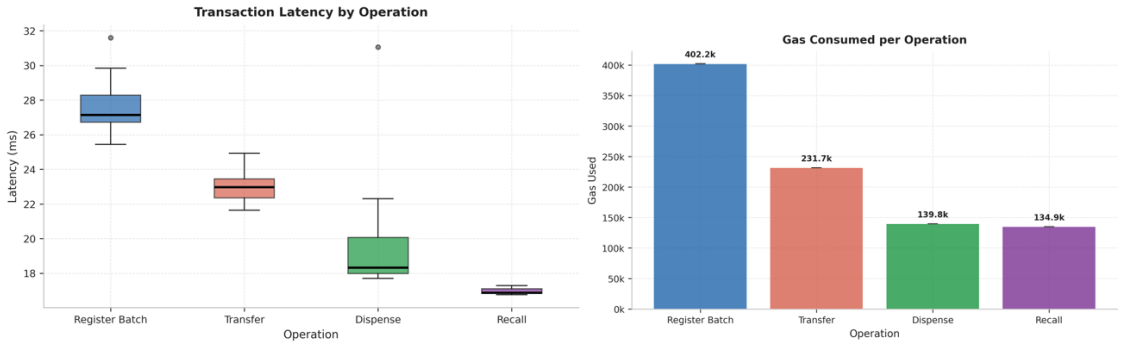


Figure 6. ZK recall verification output.

Figure 7b represents gas consumption which shows that the staircase pattern $402k \rightarrow 232k \rightarrow 140k \rightarrow 135k$ reflects the actual EVM computation cost of each operation. The

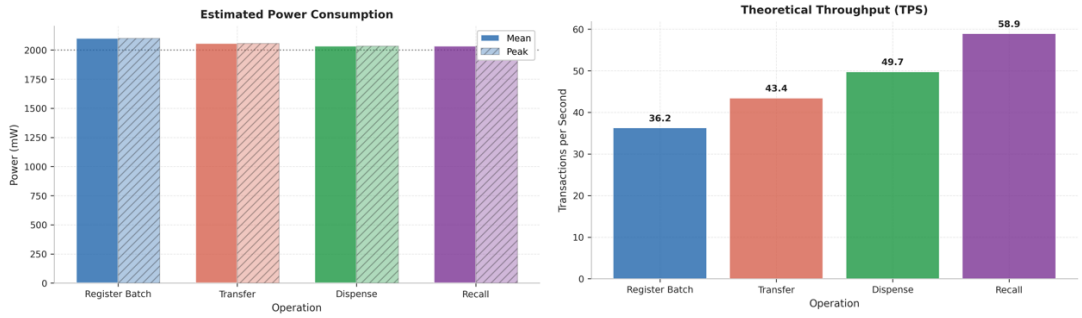
register batch is 3× more expensive than recall because it initializes the full batch structure, writes all fields to storage, and emits multiple events. This is a real architectural cost of your ZK-integrated contract.

Figure 8 demonstrates estimated power consumption (mW) and theoretical throughput, i.e., transactions per second (TPS). Specifically, Figure 8a shows the power consumption for running the smart contracts. The mean and peak bars are almost identical for every operation, meaning the gas variance within each operation type is very small. This confirms the deterministic execution—the same inputs and same cost, every time. Figure 8b represents the throughput TPS of the proposed system. Dispense (47 TPS) and recall (46.2 TPS) outperform register (31.2 TPS). This is directly inverse to gas cost—lighter operations execute faster and yield higher throughput. The system achieves between 31.2 TPS (register) and 47.0 TPS (dispense) in a single-client sequential workload, consistent with private/permissioned Ethereum deployments reported in related work.



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

Figure 7. Transaction and gas consumption per operation. (**a**) Transaction latency; (**b**) gas consumption.



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

Figure 8. Estimated power consumption and theoretical throughput (TPS). (**a**) Power consumption; (**b**) throughput.

Figure 9 represents the cumulative throughput over time of ACTMed. The cyan TPS line drops sharply in the first 0.2 s—this is expected as it is the warm-up effect where the first few transactions carry proportionally more weight. It stabilizes at around 31–32 TPS during the register batch, then climbs when lighter operations (transfer, dispense, and recall) begin. The yellow dashed cumulative Tx line rises steadily, confirming no dropped transactions.

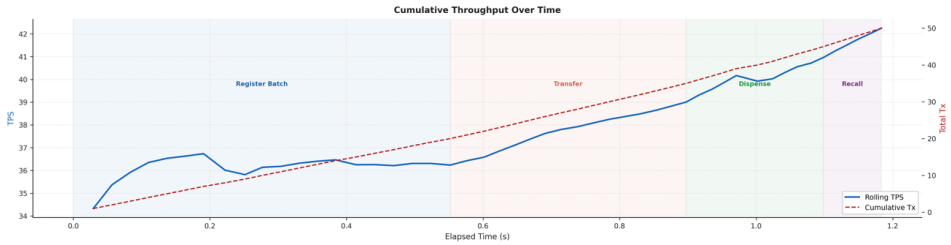


Figure 9. Cumulative throughput over time

Figure 10 shows latency distribution for all operations. The register batch (blue) has a right-skewed distribution—most transactions complete in 29–33 ms but a tail extends to 49 ms. Dispense and recall (green/red) are tightly packed at 20–22 ms with almost no spread.

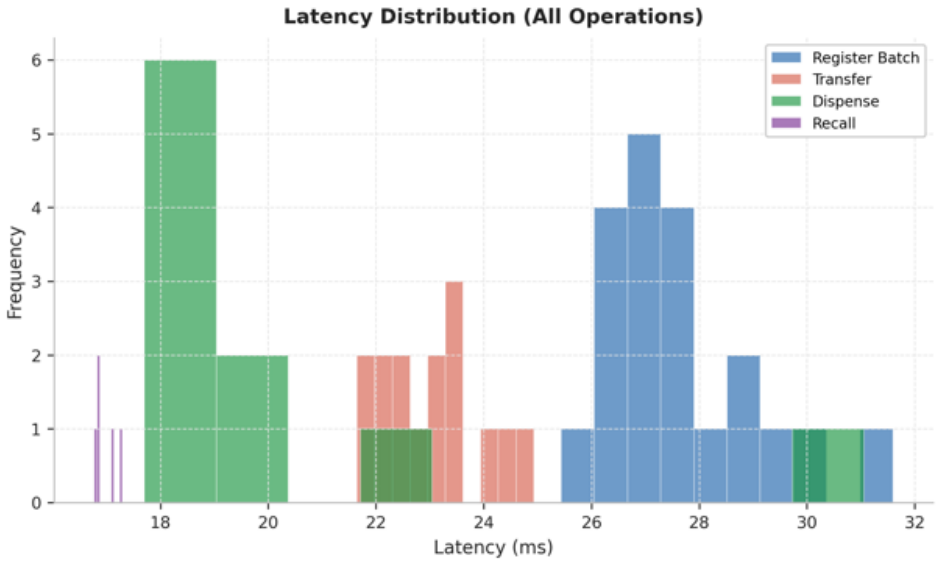


Figure 10. Latency distribution (all operations).

Table 6 shows that prior studies typically address only subsets of the problem, such as traceability, privacy, anti-counterfeit verification, or performance analysis. In contrast, the proposed system integrates ZKP-based privacy preservation, an implemented prototype, regulator-led recall governance, public verification, and practical security evaluation within a single pharmaceutical traceability and recall framework.

Table 6. Feature-wise comparison of proposed supply chain with existing pharmaceutical blockchain systems.

Study/System	ZKP-Based Privacy	Prototype/Implementation	Regulator-Led Design	Recall Workflow	Public Verification	Security Evaluation
Akram et al. [20]	✗	✗	✗	△	△	✗
Sim et al. [48]	✗	✓	△	✓	✓	△
Wu and Lin [49]	✗	△	✓	✓	✗	✗
Kravenkit and So-In [50]	✗	✓	✓	✓	△	△
Musamih et al. [51]	✗	✓	△	✗	✓	✓

Table 6. *Cont.*

Study/System	ZKP-Based Privacy	Prototype/Implementation	Regulator-Led Design	Recall Workflow	Public Verification	Security Evaluation
Hathaliya and Tanwar [52]	×	✓	×	×	×	✓
Kochovski et al. [53]	×	✓	×	△	×	△
Kalpana and Sridevi [54]	×	✓	△	×	✓	✓
Proposed System	✓	✓	✓	✓	✓	✓

✓ = Fully supported, △ = partially supported, and × = not supported.

Cross-Chain ZK Recall Validation

To validate cross-chain, we deploy a standalone `Groth16Verifier.sol` contract on a second Ethereum network, referred to as Chain B, operating on port `8545`. This verifier operates independently of the primary `SupplyChain.sol` contract deployed on Chain A, which operates on port `7545`. In the live benchmark, both local Ganache networks reported the network ID `1337`, while running on separate RPC endpoints. Although both networks used the same local network ID, the experiment still demonstrates proof verification across two independent blockchain RPC instances. For a stronger production-style cross-chain configuration, Chain B should be started with a distinct network ID, such as `1338`. We present the cross-chain ZK recall proof flow in Figure 11 and in Table 7.

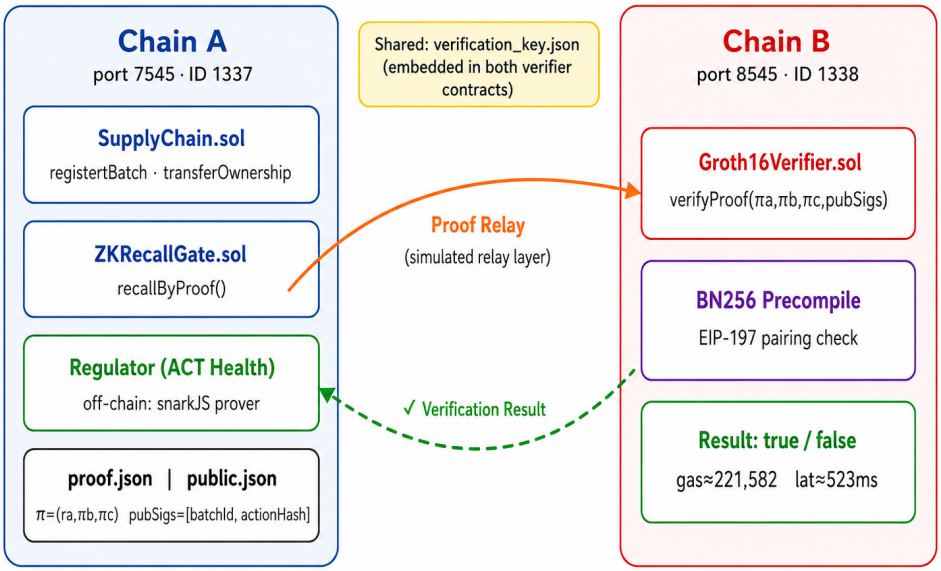


Figure 11. Cross-chain ZK recall proof flow.

When a regulator issues a ZK-authenticated recall on Chain A, the Groth16 proof tuple $(\pi_a, \pi_b, \pi_c, pubSignals)$ is relayed to Chain B. On Chain B, the `verifyProof()` function is called against the same verification key embedded in the deployed verifier contract. This demonstrates that a recall proof generated from one chain-side workflow is cryptographically verifiable on another deployed verifier instance without sharing pharmaceutical business logic, custody state, or the private recall reason.

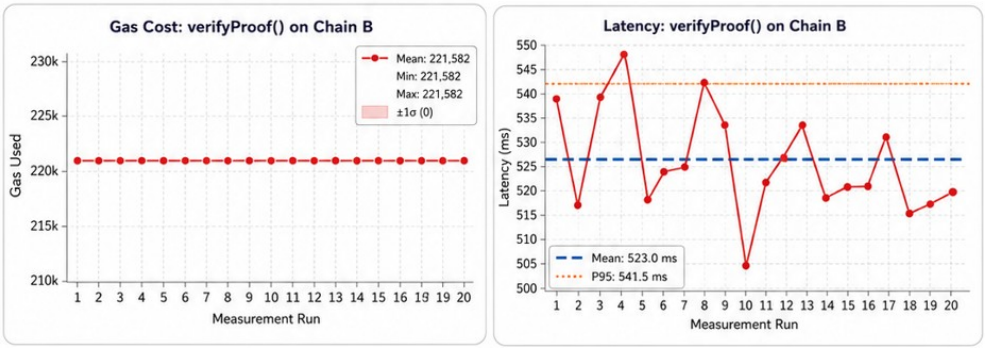
The cross-chain validation process consists of three main groups: the home chain, the proof relay layer, and the peer verification chain. Chain A acts as the home chain where the supply chain and recall logic reside. The proof relay layer transfers only the ZK proof and public signals, not the private recall reason. Chain B acts as an independent

verifier chain that validates the proof using the deployed Groth16 verifier contract and the embedded verification key.

Table 7. Functional groups in the cross-chain ZK recall validation workflow.

Group	Main Component(s)	Short Description
Chain A: Home Chain	`SupplyChain.sol`, `ZKRecallGate.sol`	Maintains the core pharmaceutical supply chain and recall workflow. The recall event is initiated on this chain, while sensitive recall information remains private and is represented through a ZK proof.
Regulator / Prover	Regulator, `snarkJS`, `proof.json`, `public.json`	Generates the Groth16 proof off-chain. The private recall reason is not exposed; only the proof tuple and public signals are prepared for verification.
Proof Relay Layer	$(\pi_a, \pi_b, \pi_c, pubSignals)$	Transfers the ZK proof and public signals from Chain A to Chain B. This layer does not require access to business logic, custody state, or private recall data.
Chain B: Peer Verification Chain	`Groth16Verifier.sol`, `verifyProof()`	Independently verifies the received Groth16 proof using the matching verification key. Chain B only confirms proof validity and does not execute the pharmaceutical supply chain logic.
Cryptographic Verification	BN256 pairing check	Performs the underlying elliptic curve pairing verification. Invalid proofs are rejected cryptographically without requiring application-level decision logic.
Benchmarking Layer	Python benchmark script, Ganache, Web3.py	Measures deployment gas, verification gas, latency, throughput, and invalid proof rejection across 20 consecutive `verifyProof()` calls on Chain B.

We benchmarked 20 consecutive `verifyProof()` calls on Chain B as presented in Figure 12. The live benchmark produced a mean verification gas cost of 221,582 gas with a standard deviation of zero, showing that the verification gas cost remained constant across all runs. The mean verification latency was 523.0 ms, with a P95 latency of 541.5 ms. The measured throughput was approximately 1.9 TPS. The deployment of the verifier contract on Chain B consumed 397,997 gas. Compared with a non-ZK recall operation on Chain A, which consumed 134,928 gas, ZK verification added 221,582 gas. Therefore, the total ZK recall cost was approximately 356,510 gas, representing a 164.2% overhead relative to the non-ZK recall operation, as is shown in Figure 13. The summary of the benchmark results is presented in Table 8.



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

Figure 12. Gas cost and latency of verifyProof() on Chain B. (**a**) Gas cost; (**b**) latency.

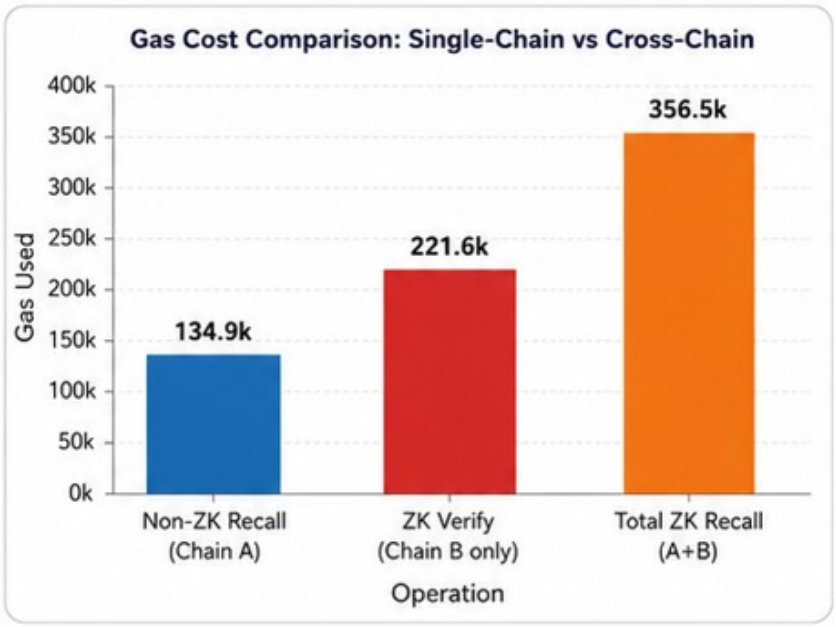


Figure 13. Gas cost comparison of single chain vs cross-chain.

Table 8. Benchmark results summary for cross-chain Groth16 proof verification.

Metric	Measured Value
Chain A RPC endpoint	Port 7545
Chain B RPC endpoint	Port 8545
Reported Chain A network ID	1337
Reported Chain B network ID	1337
Verifier contract deployed on Chain B	`Groth16Verifier.sol`
Verifier deployment gas	397,997 gas
Number of verification runs	20
Mean `verifyProof()` gas	221,582 gas
Standard deviation of verification gas	0
Mean verification latency	523.0 ms
P95 verification latency	541.5 ms
Throughput	1.9 TPS
Non-ZK recall gas on Chain A	134,928 gas
Additional ZK verification overhead	221,582 gas
Total ZK recall cost	356,510 gas
Overhead compared with non-ZK recall	164.2%
Invalid proof result	Rejected successfully

These results show that the proposed design enables cross-chain ZK recall validation with measurable and reproducible computational overhead. The peer chain does not need to store the full pharmaceutical supply chain state or reveal the private recall reason. Instead, it only requires the verifier contract and the matching verification key. The invalid proof test returned `False`, confirming that tampered proofs are rejected by the BN256 pairing check. This supports the claim that the soundness guarantee is enforced independently on the verifier chain.

4.9. Security and Trust Assumptions

The ACTMeds system assumes that stakeholder identities are verified off-chain, for example through regulator onboarding or Know Your Customer (KYC) processes, and are mapped to blockchain addresses. Trust assumptions include the following:

- Correct key custody and transaction signing by authorized organisations,
- Correct deployment and immutability of audited smart contracts,
- Correctness of the zero-knowledge circuit and verifier configuration, including soundness and completeness,
- Availability and integrity of the underlying blockchain network and client applications.

4.10. Testing and Validation

The ACTMeds pilot underwent multi-layered testing to evaluate functional correctness, security robustness, and operational reliability. Testing activities were conducted across the smart contract layer and the web application layer, combining functional unit testing with static and dynamic security analysis.

4.10.1. Functional Testing

Functional testing focused on validating the correctness of smart contract logic, role enforcement, and batch lifecycle transitions. Unit tests were implemented using the Mocha and Chai frameworks and executed against the local Ganache Ethereum network. Key contract functions, including `registerBatch()`, `transferBatch()`, `dispenseBatch()`, and `recallBatch()`, were evaluated under both valid and invalid conditions to ensure correct behaviour and failure handling.

The functional test result is presented in Table 9. All the tested scenarios behaved as expected, confirming that lifecycle enforcement and role-based access control were correctly implemented at the smart contract level.

Table 9. Smart contract functional test results.

Test Case	Expected Result	Status
Batch registration by manufacturer	`BatchRegistered` event emitted	Passed
Batch registration by unauthorized role	Transaction reverted	Passed
Regulator-issued recall	Batch state updated to `Recalled`	Passed
Unauthorized transfer attempt	Access denied by RBAC modifier	Passed

4.10.2. Integration and User Testing

End-to-end integration and user testing were conducted to validate the functional interaction between the application layer and the blockchain layer. The system was deployed in a local environment using a React frontend, a Ganache-based Ethereum network, and MetaMask wallets to simulate real-world stakeholder identities. Each participant role interacted with the system through a dedicated wallet address mapped to its corresponding on-chain role. The testing focused on verifying the correct execution of lifecycle transactions, the

consistency of on-chain state updates, and the synchronization between blockchain events and the user interface. The following representative scenario was executed to evaluate the complete operational workflow:

1. The manufacturer registered a new medicine batch on the blockchain.
2. The wholesaler accepted custody of the batch and updated its storage status.
3. The pharmacy dispensed the medicine to a patient, recording the dispense event on-chain.
4. The regulator initiated a recall for the batch, triggering the recall logic in the smart contract.
5. The customer verified the batch authenticity and recall status via the QR/GS1-based public verification interface.

All interactions were successfully executed and accurately reflected on the blockchain ledger. The lifecycle state transitions occurred as expected, and corresponding events were emitted and captured by the frontend in real time. Role-based user interfaces updated consistently across all participants, confirming correct integration between smart contracts, wallet-based authentication, and the application layer. These results demonstrate that the proposed system supports coordinated multi-role operations and provides reliable end-to-end traceability and recall verification within the tested pilot environment.

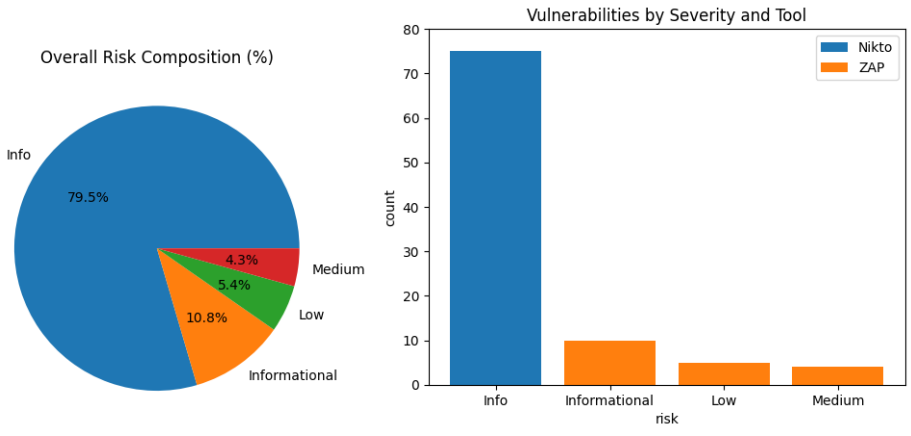
4.10.3. Dynamic Application Security Testing (DAST)

Dynamic security testing was conducted against the running web application to identify potential runtime vulnerabilities. Active scans were performed using OWASP ZAP and Nikto against the locally hosted React interface and supporting web services. Table 10 shows the dynamic security testing outcome.

Following the remediation of the identified configuration issues, no exploitable vulnerabilities were detected during subsequent scan sessions. The summary of dynamic testing using the OWASP ZAP tool is presented in Figure 14.

Table 10. Dynamic security testing results.

Tool	Scope	Findings	Risk
OWASP ZAP	Localhost: 5173 (React UI)	Missing HTTP security headers; mitigated using Helmet.js	Medium
Nikto	Local web server	No outdated software or directory exposure detected	Low



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

Figure 14. Vulnerability scan result of OWASP Zap and Nikto. (**a**) OWASP ZAP; (**b**) throughput.

Table 11 summarizes the key outcomes of the pilot evaluation across functional, usability, security, and performance dimensions. Overall, the results demonstrate that the proposed framework reliably enforces role-based lifecycle management, supports regulator-led recalls, and provides a secure and auditable foundation for pharmaceutical supply chain traceability within a controlled pilot environment.

Table 11. ACTMeds pilot evaluation summary.

Objective	Measured Outcome
End-to-end blockchain workflow	Successful execution of full lifecycle (registration → transfer → dispensing → recall).
User interface usability	Role-based dashboards validated; MetaMask-based interactions functioned as expected.
Security posture	SAST and DAST analyses reported zero high-risk vulnerabilities.
Traceability and auditability	All lifecycle events recorded immutably and retrievable from the blockchain ledger.
Scalability simulation	Stable performance observed under simulated multi-role transaction loads in Ganache.

5. Security and Risk Analysis

Security assurance constituted a core design and evaluation objective of the ACTMeds Pilot. Although the system operates on simulated health-related data within a pilot environment, it incorporates regulator-controlled transactions and security-sensitive workflows. Accordingly, a security-by-design approach was adopted, informed by established best practices from OWASP, ISO/IEC 27001, and the NIST Risk Management Framework (SP 800-37). This section presents the security objectives, STRIDE-based threat modeling, and a structured risk assessment with corresponding mitigation strategies across the system architecture.

5.1. Security Objectives

The system was designed to meet the following security objectives:

- Confidentiality: Only authorized roles, mapped to blockchain wallet addresses, are permitted to invoke state-changing operations.
- Integrity: All lifecycle and recall transactions are immutably recorded on the blockchain and verifiable through transaction hashes.
- Availability: The platform remains accessible to legitimate users within the constraints of the pilot environment.
- Accountability: Every action is cryptographically signed and traceable to a unique on-chain address.
- Privacy: No personally identifiable information or medical records are stored directly on-chain; sensitive context is handled off-chain.

5.2. STRIDE Threat Modeling

A structured threat model was developed using the STRIDE framework to identify potential security risks across the smart contract, application, and infrastructure layers. Table 12 summarizes the identified threats and applied mitigations. All identified threats were mitigated to acceptable levels through architectural controls, secure coding practices, and layered testing.

Table 12. STRIDE Threat Model and Mitigation Strategies.

Category	Threat Description	Affected Component	Risk Level	Mitigation Strategy
Spoofing	Impersonation of authorized roles (e.g., manufacturer or regulator)	Smart contract functions	Medium	Wallet-based authentication and Solidity role modifiers (e.g., `onlyManufacturer`, `onlyRegulator`).
Tampering	unauthorized modification of batch or recall data	Blockchain storage	Low	Immutable ledger prevents state alteration; integrity verifiable via transaction hashes.
Repudiation	Denial of previously executed actions	Transaction logs	Low	Cryptographically signed transactions provide non-repudiation.
Information Disclosure	Exposure of sensitive metadata or internal endpoints	React UI/Web3 interface	Medium	Security headers via Helmet.js; minimisation and anonymisation of off-chain metadata.
Denial of Service (DoS)	Resource exhaustion or network overload	Ganache/React server	Medium	Local monitoring; scalable to managed blockchain infrastructure in production.
Elevation of Privilege	unauthorized access to regulator-only operations	Smart contract logic	High	Strict RBAC enforcement with address allow-listing.

Beyond the generalized STRIDE categories, Table 13 presents a dedicated analysis of blockchain-specific attack vectors that warrant explicit consideration in the ACTMeds context. Replay attacks present a risk when valid transactions—such as ZKRecall proof submissions or custody transfer events—are maliciously rebroadcast to induce duplicate state changes. In ACTMeds, this is mitigated through Ethereum's native transaction nonce mechanism, which ensures each transaction is accepted only once, and through the inclusion of batch-specific identifiers and timestamp-bound constraints within ZKRecall proof public inputs, preventing proof reuse across batches or time windows. Front-running is a concern in public Ethereum deployments where adversaries can observe pending transactions in the mempool and submit competing transactions with higher gas fees to manipulate execution order—for example, intercepting a recall initiation before it is confirmed. This risk is substantially reduced in the ACTMeds deployment context by targeting permissioned consortium chains or layer-2 infrastructures for production, where public mempool exposure is eliminated. Transaction ordering dependence, where contract behaviour varies based on the sequence in which transactions are processed, is mitigated by the state machine design of SupplyChain.sol, which enforces strict sequential lifecycle transitions and explicitly rejects any out-of-order state changes through role-aware modifiers—ensuring consistent contract behaviour regardless of transaction ordering.

5.3. Risk Assessment Matrix

A qualitative risk assessment was conducted to evaluate residual risks considering likelihood, impact, and mitigation effectiveness. Table 14 summarizes the key risks and controls.

Table 13. Blockchain-Specific Threat Analysis.

Threat	Description	Affected Component	Risk Level	Mitigation Strategy
Replay Attack	Resubmission of a previously valid transaction (e.g., a recall proof or custody transfer) to trigger duplicate or unauthorized state changes.	Smart contract functions	Medium	Ethereum's built-in transaction nonce mechanism; ZKRecall proofs include batch-specific and timestamp-bound public inputs to prevent proof reuse.
Front-Running	A malicious network participant observes a pending recall or transfer transaction in the mempool and submits a competing transaction with higher gas to alter execution order.	Blockchain mempool/ smart contract	Medium	Production deployment on permissioned or layer-2 network eliminates public mempool exposure; commit-reveal scheme applicable for sensitive operations.
Transaction Ordering Dependence	Smart contract logic may produce inconsistent state outcomes if block producers manipulate transaction execution sequence.	Smart contract state machine	Low	State machine design enforces strict sequential lifecycle transitions; out-of-order transitions explicitly rejected by role and state modifiers.

Table 14. Risk Assessment and Mitigation.

Risk ID	Description	Likelihood	Impact	Risk Rating	Mitigation
R1	Smart contract vulnerabilities (e.g., reentrancy, overflow)	Medium	High	High	Static analysis using Solhint and Mythril prior to deployment.
R2	Frontend exposure to XSS or CSRF attacks	Medium	Medium	Medium	Helmet.js security headers and React input sanitisation.
R3	MetaMask phishing or fraudulent wallet connections	Low	High	Medium	Local test network usage and enforced address verification.
R4	Data loss or corruption	Low	High	Low	Immutable blockchain state with event-based recovery.
R5	Source code access mismanagement	Medium	Medium	Medium	Role-based GitHub access control and branch protection.
R6	Limited time for security validation	High	Medium	High	Integrated DevSecOps pipeline with automated scans per commit.

Following mitigation, the residual risk level of the system was assessed as low to medium, which is considered acceptable for a controlled pilot deployment. No high-severity vulnerabilities remained unaddressed at the conclusion of testing.

6. Conclusions and Future Work

This paper presented the ACTMeds pilot, a regulator-led medicine traceability and recall platform designed to address authenticity, auditability, and trust challenges in pharmaceutical supply chains. The pilot demonstrates how blockchain technology can be applied in a public health context to provide tamper-evident lifecycle tracking, authoritative recall issuance, and verifiable accountability across multiple stakeholders. A distinct feature of this work is the introduction of a privacy-preserving recall mechanism using zero-knowledge proofs, demonstrating how regulatory compliance can be verified without exposing sensitive incident details—a viable pathway for reconciling transparency requirements with confidentiality obligations in healthcare regulation. The system achieved end-to-end batch traceability, enforced role-based access control through smart contracts, and generated immutable audit trails accessible to regulators and auditors. Security evaluation through integrated static and dynamic testing identified no critical vulnerabilities, with all medium-risk findings mitigated prior to evaluation. While the pilot operates in a controlled environment, it establishes a practical foundation for future government-led digital health initiatives. Scaling to production will require formal governance structures, enterprise-grade key management, and integration with existing healthcare information systems. Nonetheless, the results validate blockchain's suitability as a trust-enabling infrastructure for medicine supply chain oversight in the Australian Capital Territory, offering a promising direction for improving patient safety, recall responsiveness, and public trust in medicine distribution. Several practical deployment challenges remain to be addressed in future work, including on-chain and off-chain data consistency, integration with existing healthcare information systems such as hospital ERP platforms and TGA regulatory databases, and real-world scalability evaluation under production transactionloads. Future work will explore deployment on permissioned consortium chains or layer-2 infrastructures to assess throughput, latency, and gas cost characteristics at scale.

Author Contributions: A.S.: conceptualization; data curation; implementation; writing—original draft; writing; visualization; and formal analysis. J.A. and N.H.C.: investigation; methodology; resources; software; validation; and writing—original draft. M.A.U. and R.R.: review and editing; writing; investigation; methodology; formulation; and visualization. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external or internal funding.

Data Availability Statement: The implementation of the proposed system, including smart contracts and supporting components, is available on GitHub at: https://github.com/secureintelligencelab/ZKMedSystem.git (accessed on 12 May 2026).

Acknowledgments: The authors used an artificial intelligence–based language model (Grammarly, ChatGPT 5.5, developed by OpenAI) to assist in refining the text and correcting grammatical errors. The authors reviewed and edited the content to ensure accuracy and accountability for all statements made in the paper.

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

References

1. World Health Organization. Substandard and Falsified Medical Products. 2017. Available online: https://www.who.int/news-room/fact-sheets/detail/substandard-and-falsified-medical-products (accessed on 5 February 2026).
2. Mackey, T.K.; Nayyar, G. A review of existing and emerging digital technologies to combat the global trade in fake medicines. *Expert Opin. Drug Saf.* **2017**, *16*, 587–602. [CrossRef]
3. GS1. GS1 Healthcare Traceability Standard. 2020. Available online: https://www.gs1.org/industries/healthcare/traceability (accessed on 5 February 2026).

4. Casino, F.; Dasaklis, T.K.; Patsakis, C. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telemat. Inform.* **2019**, *36*, 55–81.
5. Azaria, A.; Ekblaw, A.; Vieira, T.; Lippman, A. Medrec: Using blockchain for medical data access and permission management. In *Proceedings of the 2016 2nd International Conference on Open and Big Data (OBD)*; IEEE: Piscataway, NJ, USA, 2016; pp. 25–30.
6. Chen, X.; He, C.; Chen, Y.; Xie, Z. Internet of Things (IoT)—blockchain-enabled pharmaceutical supply chain resilience in the post-pandemic era. *Front. Eng. Manag.* **2023**, *10*, 82–95.
7. Sahoo, M.; Singhar, S.S.; Sahoo, S.S. A blockchain based model to eliminate drug counterfeiting. In *Machine Learning and Information Processing: Proceedings of ICMLIP 2019*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–222.
8. Australian Commission on Safety and Quality in Health Care. Australian Charter of Healthcare Rights. 2019. Available online: https://www.safetyandquality.gov.au/our-work/partnering-consumers/australian-charter-healthcare-rights (accessed on 5 February 2026).
9. Xu, X.; Weber, I.; Staples, M. *Architecture for Blockchain Applications*; Springer: Berlin/Heidelberg, Germany, 2019.
10. Joni, S.A.; Rahat, R.; Tasnin, N.; Ghose, P.; Uddin, M.A.; Ayoade, J. Hybrid-Blockchain-Based Electronic Voting Machine System Embedded with Deepface, Sharding, and Post-Quantum Techniques. *Blockchains* **2024**, *2*, 366–423.
11. Dubovitskaya, A.; Baig, F.; Xu, Z.; Shukla, R.; Zambani, P.S.; Swaminathan, A.; Jahangir, M.M.; Chowdhry, K.; Lachhani, R.; Idnani, N.; et al. ACTION-EHR: Patient-centric blockchain-based electronic health record data management for cancer care. *J. Med. Internet Res.* **2020**, *22*, e13598. [PubMed]
12. Kuo, T.T.; Kim, H.E.; Ohno-Machado, L. Blockchain distributed ledger technologies for biomedical and health care applications. *J. Am. Med. Inform. Assoc.* **2017**, *24*, 1211–1220.
13. Benchoufi, M.; Porcher, R.; Ravaud, P. Blockchain protocols in clinical trials: Transparency and traceability of consent. *F1000Research* **2018**, *6*, 66. [CrossRef]
14. Maslove, D.M.; Klein, J.; Brohman, K.; Martin, P. Using blockchain technology to manage clinical trials data: A proof-of-concept study. *JMIR Med. Inform.* **2018**, *6*, e11949.
15. Zhou, L.; Diro, A.; Saini, A.; Kaisar, S.; Hiep, P.C. Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. *J. Inf. Secur. Appl.* **2024**, *80*, 103678. [CrossRef]
16. Jia, W.; Xie, T.; Wang, B. A privacy-preserving scheme with multi-level regulation compliance for blockchain. *Sci. Rep.* **2024**, *14*, 438. [CrossRef] [PubMed]
17. Tseng, J.H.; Liao, Y.C.; Chong, B.; Liao, S.w. Governance on the drug supply chain via gcoin blockchain. *Int. J. Environ. Res. Public Health* **2018**, *15*, 1055.
18. Kumar, R.; Tripathi, R. Traceability of counterfeit medicine supply chain through Blockchain. In *Proceedings of the 2019 11th International Conference on Communication Systems & Networks (COMSNETS)*; IEEE: Piscataway, NJ, USA, 2019; pp. 568–570.
19. Zakari, N.; Al-Razgan, M.; Alsaadi, A.; Alshareef, H.; Saigh, H.A.; Alashaikh, L.; Alharbi, M.; Alomar, R.; Alotaibi, S. Blockchain technology in the pharmaceutical industry: A systematic review. *PeerJ Comput. Sci.* **2022**, *8*, e840. [CrossRef]
20. Akram, W.; Joshi, R.; Haider, T.; Sharma, P.; Jain, V.; Garud, N.; Singh, N. Blockchain technology: A potential tool for the management of pharma supply chain. *Res. Soc. Adm. Pharm.* **2024**, *20*, 156–164. [CrossRef] [PubMed]
21. Shaikh, M.; Memon, S.A.; Ebrahimi, A.; Wiil, U.K. A Systematic Literature Review for Blockchain-Based Healthcare Implementations. *Healthcare* **2025**, *13*, 1087.
22. Chaabaoui, S.; Oughannou, Z.; Chaoui, H. A Systematic Review of Blockchain Applications in Pharmaceutical Supply Chain Traceability and Security. In *Proceedings of the 2024 10th International Conference on Optimization and Applications (ICOA)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7.
23. Ben-Sasson, E.; Chiesa, A.; Tromer, E.; Virza, M. Succinct {Non-Interactive} zero knowledge for a von neumann architecture. In Proceedings of the 23rd USENIX Security Symposium (USENIX Security 14), San Diego, CA, USA, 20–22 August 2014; pp. 781–796.
24. Hopwood, D.; Bowe, S.; Hornby, T.; Wilcox, N. *Zcash Protocol Specification*; GitHub: San Francisco, CA, USA 2016; Volume 4, p. 32.
25. Ben-Sasson, E.; Bentov, I.; Horesh, Y.; Riabzev, M. Scalable, transparent, and post-quantum secure computational integrity. *Cryptology ePrint Archive* **2018**. Available online: https://eprint.iacr.org/2018/46 (accessed on 12 May 2026).
26. Bünz, B.; Bootle, J.; Boneh, D.; Poelstra, A.; Wuille, P.; Maxwell, G. Bulletproofs: Short proofs for confidential transactions and more. In *Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP)*; IEEE: Piscataway, NJ, USA, 2018; pp. 315–334.
27. Kosba, A.; Miller, A.; Shi, E.; Wen, Z.; Papamanthou, C. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP)*; IEEE: Piscataway, NJ, USA, 2016; pp. 839–858.
28. Sun, X.; Yu, F.R.; Zhang, P.; Sun, Z.; Xie, W.; Peng, X. A survey on zero-knowledge proof in blockchain. *IEEE Netw.* **2021**, *35*, 198–205. [CrossRef]
29. Zyskind, G.; Nathan, O. Decentralizing privacy: Using blockchain to protect personal data. In *Proceedings of the 2015 IEEE Security and Privacy Workshops*; IEEE: Piscataway, NJ, USA, 2015; pp. 180–184.

30. Pang, Y.; Yu, Z.; Zhang, H.; Sim, C.; Chang, M.L.; Chong, A. Blockchain Technology Transforms the Pharmaceutical Supply Chain. *Manag. Bus. Rev.* **2025**, *5*, 24–33. [CrossRef]
31. Abdallah, S.; Nizamuddin, N. Blockchain-based solution for pharma supply chain industry. *Comput. Ind. Eng.* **2023**, *177*, 108997. [CrossRef]
32. Uddin, M. Blockchain Medledger: Hyperledger fabric enabled drug traceability system for counterfeit drugs in pharmaceutical industry. *Int. J. Pharm.* **2021**, *597*, 120235. [CrossRef] [PubMed]
33. Gomasta, S.S.; Dhali, A.; Tahlil, T.; Anwar, M.M.; Ali, A.M.S. PharmaChain: Blockchain-based drug supply chain provenance verification system. *Heliyon* **2023**, *9*, e17957. [CrossRef]
34. Aziz, A.; Suman, S.; Sohail, S.S.; Madsen, D.Ø. Leveraging blockchain for combatting drug counterfeiting in the pharmaceutical sector. *Cogent Bus. Manag.* **2025**, *12*, 2551811. [CrossRef]
35. Gaynor, M.; Gillespie, K.; Roe, A.; Crannage, E.; Tuttle-Newhall, J. Blockchain Applications in the Pharmaceutical Industry. *Blockchain Healthc. Today* **2024**, *7*, 10-30953. [CrossRef] [PubMed]
36. Mezquita, Y.; Podgorelec, B.; Gil-González, A.B.; Corchado, J.M. Blockchain-based supply chain systems, interoperability model in a pharmaceutical case study. *Sensors* **2023**, *23*, 1962. [CrossRef] [PubMed]
37. Triet, N.; Khanh, V.; Nam, T.; Khoa, T.; Khiem, H.; Bao, T.; Hieu, D.; Anh, N. Cross-Chain Analysis of Blockchain-Based Pharmaceutical Supply Chain Management Using NFTs. In *Proceedings of the International Conference on Big Data*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 31–45.
38. Matt Sample, VP Manufacturing Operations AmerisourceBergen. MediLedger DSCSA Pilot Project. 2020. Available online: https://www.fda.gov/media/168283/download (accessed on 10 February 2026).
39. Innovative Health Initiative. PharmaLedger: Building Trust in Data Sharing Through Blockchain, to Accelerate Digital Transformation in Healthcare. 2022. Available online: https://www.ihi.europa.eu/projects-results/project-factsheets/pharmaledger (accessed on 10 February 2026).
40. Bocek, T.; Rodrigues, B.B.; Strasser, T.; Stiller, B. Blockchains everywhere-a use-case of blockchains in the pharma supply chain. In *Proceedings of the 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*; IEEE: Piscataway, NJ, USA, 2017; pp. 772–777.
41. Ahmad, R.W.; Ko, K.M.; Rashid, A.; Rodrigues, J.J. Blockchain for food industry: Opportunities, requirements, case studies, and research challenges. *IEEE Access* **2024**, *12*, 117363–117378. [CrossRef]
42. Naoum-Sawaya, J.; Elhedhli, S.; De Carvalho, P. Strategic blockchain adoption to deter deceptive counterfeiters. *Eur. J. Oper. Res.* **2023**, *311*, 373–386. [CrossRef]
43. Pareek, V.; Saran, D.; Sharma, L.; Jakhar, P.; Kumar, S. Blockchain technology in pharmaceutical industry: A review of recent research articles on PubMed. *Scr. Medica* **2024**, *55*, 357–369. [CrossRef]
44. Xiong, R.; Cheng, J.; Pu, J.; Dong, X.; Chen, C.; Xu, Z. Enhancing trust and collaboration: A reputation-driven mechanism for cross-chain IoT data sharing. *Comput. Netw.* **2025**, *264*, 111266. [CrossRef]
45. Xiong, R.; Xiao, Q.; Wang, Z.; Xu, Z.; Shan, F. Leveraging lightweight blockchain for secure collaborative computing in UAV Ad-Hoc Networks. *Comput. Netw.* **2024**, *251*, 110612. [CrossRef]
46. Circom Team. Proving Circuits with ZK. 2023. Available online: https://docs.circom.io/getting-started/proving-circuits/ (accessed on 21 April 2026).
47. Groth, J. On the size of pairing-based non-interactive arguments. In *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 305–326.
48. Sim, C.; Zhang, H.; Chang, M.L. Improving end-to-end traceability and pharma supply chain resilience using blockchain. *Blockchain Healthc. Today* **2022**, *5*, 10–30953.
49. Wu, X.; Lin, Y. Blockchain recall management in pharmaceutical industry. *Procedia CIRP* **2019**, *83*, 590–595. [CrossRef]
50. Kravenkit, S.; So-In, C. Blockchain-based traceability system for product recall. *IEEE Access* **2022**, *10*, 95132–95150. [CrossRef]
51. Musamih, A.; Salah, K.; Jayaraman, R.; Arshad, J.; Debe, M.; Al-Hammadi, Y.; Ellahham, S. A blockchain-based approach for drug traceability in healthcare supply chain. *IEEE Access* **2021**, *9*, 9728–9743. [CrossRef]
52. Hathaliya, J.J.; Tanwar, S. Intelligent privacy-preserving data management framework for medicine supply chain system. *Secur. Priv.* **2024**, *7*, e426. [CrossRef]
53. Kochovski, P.; Masmoudi, M.; Bouhamoum, R.; Stankovski, V.; Baazaoui, H.; Ghedira, C.; Vodislav, D.; Mecharnia, T. Drug traceability system based on semantic blockchain and on a reputation method. *World Wide Web* **2024**, *27*, 62. [CrossRef]
54. Kalpana, R.; Sridevi, S. Securing drug traceability: Blockchain-enhanced privacy protection and anti-counterfeit measures in pharmaceutical supply chains. *J. Pharm. Innov.* **2025**, *20*, 90. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.