International Journal of Convergence Research in Engineering, Technology & Management (IJCRETM)
Volume 1, Issue 1 | July–September 2026 | *Peer Reviewed (Double-Blind)* • *Open Access* • *Quarterly*

A Unified Framework for Blockchain-Based Cryptographic Cybersecurity

Tanishka Pahwa[1], **Geet Bawa**[2,3*]

[1]*B.Tech. Student, Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India*
[2]*School of Computer Science, Khalsa University, Amritsar, Punjab, India*
[3]*Department of Computer Applications, CT University, Ludhiana, Punjab, India*

***Corresponding Author:** geetbawa_mca@yahoo.co.in | **ORCID:** 0009-0006-3028-7219

ARTICLE INFORMATION

Manuscript ID:	IJCRETM-20260101098
How to Cite:	Pahwa, T., & Bawa, G. (2026). A unified framework for blockchain-based cryptographic cybersecurity. *International Journal of Convergence Research in Engineering, Technology & Management, 1*(1 IJCRETM-20260101098.
Copyright & License:	© 2026 The Author(s). Published by A2Z EduLearningHub, India. This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0).

Abstract

As digital systems increasingly rely on decentralized trust, cryptographic guarantees, and active threat monitoring simultaneously, blockchain, cryptography, and cybersecurity continue to be researched and engineered as largely separate domains, with little work modeling how they function together as one security ecosystem. This paper addresses that gap by proposing a converged, three-layer framework: cryptographic, blockchain, and cybersecurity; connected through explicit, bidirectional feedback rather than one-directional dependency. The framework is evaluated against a defined adversary model, mapped against nine common attack vectors, and supported by a semi-formal justification of the integrity, non-repudiation, and resilience properties it claims to provide. A working Python-based implementation, applied to an IoT security use case, demonstrates the framework in practice: across 200 transactions among 20 simulated devices and 7 consensus nodes, cryptographic and cybersecurity overhead remained under one millisecond per transaction, while a replay attack was detected and blocked before reaching consensus. A direct comparison against a blockchain-only baseline shows this is not incidental; the baseline accepted the same replayed transaction a second time, while the converged framework did not. The framework's applicability is further illustrated across financial systems, healthcare data sharing, digital identity, and supply chain integrity. Scalability, post-quantum migration, regulatory tensions, and the absence of standardized cross-layer interfaces remain open challenges. This work contributes a conceptual architecture, a semi-formal security analysis, and empirical evidence that meaningful security value arises from the interaction of these three domains, not from any one in isolation.

Keywords: Blockchain; Cryptography; Cybersecurity; Converged Security Framework; Threat Model

1. Introduction

As the digitalization of processes is increasing, the shortcomings of traditionally centralized security systems are becoming apparent. These include the risks of single points of failure, insider attacks, and data breaches. Blockchain technology offers decentralization, transparency, and immutability of data as well as a trustworthy environment for the stakeholders [1-2]. Cryptography provides confidentiality, integrity, and authentication [3-4]. Cybersecurity provides monitoring, access control, and solutions for attacks such as Sybil, 51%, smart-contract, and man-in-the-middle [5]. Figure 1 below represents the convergence of these three areas. Table 1 demonstrates their interrelation, implementation mechanisms, threats, and limitations. Although they are considered separate fields of secure digital systems, many attacks target more than one area. The convergence of these three pillars is especially prominent in emerging platforms such as Web3, IoT, Central Bank Digital Currencies (CBDCs), and digital identities, as presented in Figure 2.

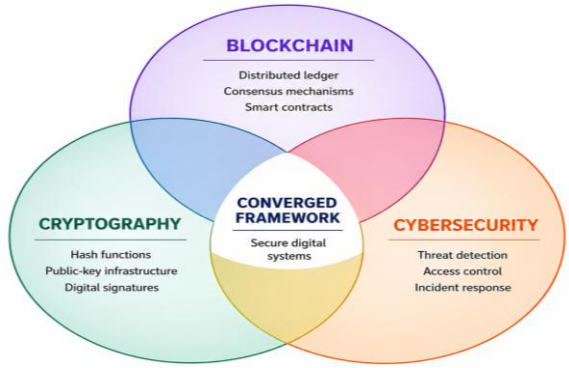


Figure 1. Convergence of Blockchain, Cryptography and Cybersecurity

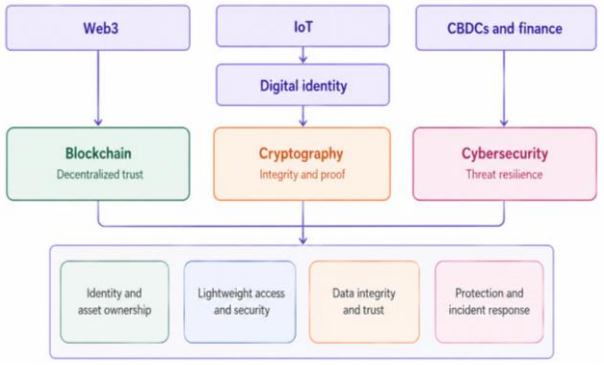


Figure 2. Application Areas of the Converged Blockchain–Cryptography–Cybersecurity Framework

Table 1. Comparison of the Security Role of Blockchain, Cryptography and Cybersecurity

Domain	Role & Mechanisms	Threats	Limitation
Blockchain	Decentralized trust, immutability, transparency; PoW, PoS, PBFT	51% attacks, double-spending, single-point failures	Off-chain data are not safe; centralization of consensus
Cryptography	Integrity, authentication, confidentiality; hashing, digital signatures, PKI	Tampering, spoofing, disclosure	Inadequate key management; quantum computing
Cybersecurity	Monitoring, access control, incident response; IDS, firewalls, IAM	MITM, insider threats, phishing	Response time effectiveness

1.1 Problem Statement

The existing body of research rarely integrates blockchain technology, cryptography, and cybersecurity for eliminating threats in digital systems. Blockchain, cryptography, and cybersecurity are treated as separate areas of research, and the interconnections between them are rarely considered. Additionally, the potential negative impact of decisions made in one area on the implementation of solutions in another area is rarely focused on. The paper develops a conceptual framework that will incorporate blockchain technology, associated cryptographic protocols and mechanisms, and cybersecurity aspects to achieve a secure digital system.

1.2 Objectives and Contributions

The objectives of this paper include:

- Identifying and discussing the basic concepts, principles, and elements of blockchain technology, cryptography, and cybersecurity;
- Establishing the connections between the discussed areas;
- Developing a conceptual framework that involves elements of blockchain technology, cryptography, and cybersecurity;
- Considering the threats and security requirements;
- Demonstrating the application of the developed conceptual framework in various application areas;
- Identifying the challenges and prospects of the areas under consideration.

1.3 Paper Organization

The remainder of this paper is organized as follows. Section 2 presents the background and preliminaries of blockchain, cryptography, and cybersecurity. Section 3 reviews the existing literature and identifies the research gap. Section 4 presents the proposed converged framework, including its conceptual model, layer interactions, design principles, and assumptions. Section 5 describes the threat model and security analysis. Section 6 presents the framework's use cases and applications. Section 7 describes the implementation, experimental setup, algorithmic design, and results. Section 8 evaluates the framework's performance, comparison with non-converged approaches, strengths, and limitations. Section 9 discusses future research directions and open challenges. Finally, Section 10 concludes the paper with the key findings and future scope.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

2. Background and Preliminaries

In this section, we briefly describe the background concepts related to blockchain, cryptography, and cybersecurity, which form the basis of the proposed converged security framework.

2.1 Blockchain Basics

Blockchain is a decentralized, immutable distributed ledger of blocks with validated transactions stored in a chain using cryptographic hashing [6]. It utilizes various consensus mechanisms, such as PoW, PoS, PBFT, and DPoS, for transaction validation depending on the architecture type, which can either be public, private, or consortium, as shown in Figure 3. The main trade-offs among these consensus mechanisms are summarized in Table 2.

2.2 Crypto Basics

Modern cryptography provides confidentiality, integrity, authentication, privacy, and non-repudiation using technologies such as encryption (AES), public-key cryptography (RSA/ECC), hashing (SHA-256), digital signatures (ECDSA), Merkle trees, and Zero-Knowledge Proofs (ZKPs) [7-8]. The blockchain-related applications of these cryptographic primitives are given in Table 3. The blockchain immutability relies on hash chaining:

$$H_i = Hash(B_i \,||\, H_{i-1}) \tag{1}$$

2.3 Cybersecurity Basics

The cybersecurity principles related to blockchain networks are best described within the CIA (Confidentiality, Integrity, and Availability) triad as shown in Figure 4. The most common cyber-attack types in blockchain systems include Sybil, 51%, MITM, replay, and smart-contract attacks [9].

2.4 Why do we need to converge these three?

Blockchain relies on cryptography to ensure trust and enable authentication in decentralized systems. At the same time, cybersecurity plays a crucial role in securing blockchain networks and cryptographic protocols, as evident from the interrelation between the CIA triad and blockchain security. This interconnectivity is well demonstrated in Figure 5 below, which shows how cybersecurity and cryptography converge to secure blockchains.

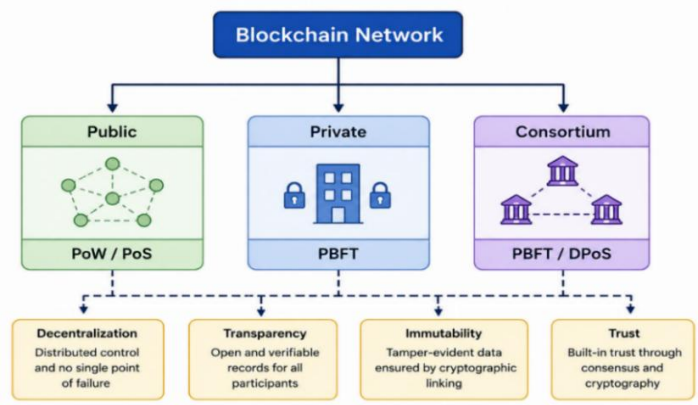


Figure 3. Blockchain network classifications and their commonly used consensus mechanisms

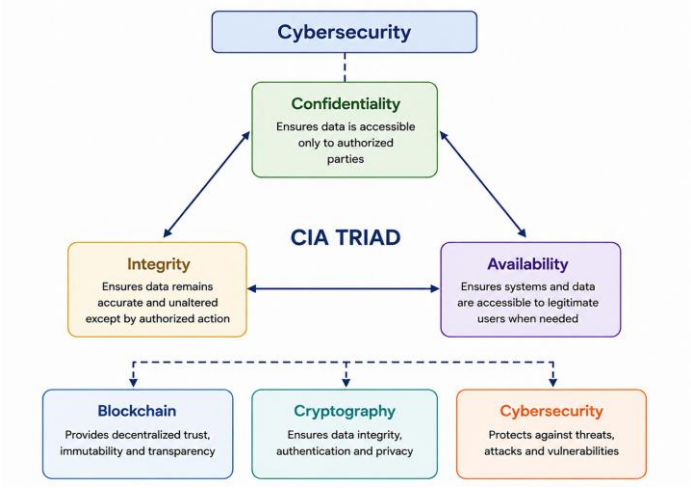


Figure 4. CIA triad and its relationship with blockchain-enabled systems

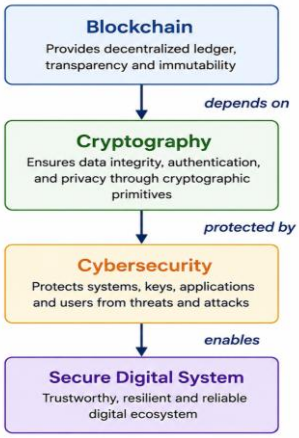


Figure 5. Interdependence of blockchain, cryptography, and cybersecurity for secure digital systems

Table 2. Comparison of Blockchain Consensus Mechanisms

Consensus	Energy Consumption	Throughput	Decentralization	Fault Tolerance	Example
PoW	High	Low	High	51% honest mining power	Bitcoin
PoS	Low	Medium–High	High	Honest majority of stake	Ethereum
PBFT	Low	High	Low	≤1/3 Byzantine nodes	Hyperledger Fabric
DPoS	Low	High	Medium	Honest elected delegates	EOS, TRON

Table 3. Major Cryptographic Primitives in Blockchain

Primitive	Primary Purpose	Blockchain Application
AES	Confidentiality	Secure data storage
RSA/ECC	Authentication & key exchange	Wallets, digital identities
SHA-256	Data integrity	Block hashing, transaction verification
ECDSA	Authentication & non-repudiation	Transaction validation
Merkle Tree	Efficient verification	Block/transaction verification
ZKP	Privacy preservation	Anonymous transactions, identity verification

3. Literature Review

The existing research on blockchain, cryptography, and cybersecurity includes studies related to consensus security, smart contracts, privacy, post-quantum cryptography, IoT security, and decentralized identity. Table 4 below contains a summary of existing works in terms of key technologies, their contributions, and limitations.

Table 4. Comparative Analysis of Existing Literature

Parameter	Study	Brief finding
Blockchain Security Studies		
Consensus & system-wide vulnerabilities	Yakubu et al. (2024) [2]	PRISMA review of 3,749 papers; 290 selected; analyzed consensus security and vulnerabilities.
	Bordeianu & Popescu (2025) [10]	Linked blockchain design factors to 51%, Sybil, and double-spending attacks; proposed BVQRF.
	Siam et al. (2025) [11]	Classified network, consensus, smart-contract, and user risks and mitigation strategies.
	Jiang et al. (2026) [12]	Cross-layer DeFi survey; highlighted interactions among code, economics, and infrastructure.
Smart contract vulnerability detection	De Baets et al. (2024) [13]	Reviewed 88 studies on ML/DL/hybrid vulnerability detection.
	Wei et al. (2024) [14]	Reviewed lifecycle-wide smart-contract security, including attacks, tools, datasets, and evaluation.

Parameter	Study	Brief finding
	Wu et al. (2025) [15]	Examined reentrancy, overflow, access control, and ordering attacks and mitigation.
	Hejazi & Lashkari (2025) [16]	Surveyed 256 detection tools; identified coverage limitations.
	Salzano et al. (2025) [17]	Evaluated 20 tools/2,182 vulnerabilities; combined approaches improved coverage.
	Yuan et al. (2025) [18]	Proposed MOS, an LLM-based MoE framework for Solidity vulnerability detection.
Sybil & identity threats	Liu et al. (2025) [7]	Subgraph-LightGBM detected Sybil addresses with >0.90 precision, recall, F1, and AUC.
	Platt et al. (2024) [19]	Proposed Sybil Attack Vulnerability Trilemma, highlighting unavoidable design trade-offs.
	Erfan et al. (2026) [20]	FL-based Sybil defense for IIoT achieved >91% accuracy and recall.
	Tej, Y. V. et al. (2024) [21]	Used trust, anomaly detection, signatures, and Merkle trees for Sybil detection.
Cryptographic Advances Applied to Blockchain		
Post-quantum cryptography & blockchain migration	Chhetri et al. (2025) [3]	Surveyed PQC schemes and assessed security, performance, and blockchain/IoT suitability.
	Ahmed et al. (2025) [4]	Evaluated 9 cryptographic libraries for ML-KEM, ML-DSA, FALCON, and SLH-DSA; identified implementation challenges.
	Arfaoui & El-Gayar (2025) [8]	Reviewed 48 lattice-based cryptography studies; highlighted overhead, compatibility, and standardization issues.
	Sezer & Akleylek (2026) [22]	Proposed PQC blockchain for IoT combining Kyber, Dilithium, STARK-ZKPs, and smart contracts.
	Faridi et al. (2022) [23]	Reviewed quantum threats to blockchain and quantum-resistant solutions; identified lattice signatures as promising.
3.2.2 Zero-knowledge proofs for blockchain privacy	Shashidhara et al. (2025) [24]	Surveyed zk-SNARKs, zk-STARKs, Bulletproofs, and PLONK for blockchain privacy and Ethereum applications.
	Gupta (2025) [25]	Reviewed ZKP applications in blockchain, identity, healthcare, and finance; highlighted scalability/interoperability challenges.
	Oude Roelink et al. (2024) [26]	Systematically compared 41 studies on zk-SNARKs, zk-STARKs, and Bulletproofs for privacy-preserving authentication.
	Zhou et al. (2024) [27]	Reviewed ZKP-based blockchain identity sharing, privacy mechanisms, and future directions.
	Malatrait & Sirac (2025) [28]	FibRace benchmark: 6,047 participants generated 2.19M+ proofs, demonstrating feasible mobile ZKP generation.
3.2.3 Homomorphic encryption & secure computation	Bao et al. (2024) [29]	Proposed DHSMPC, reducing ciphertext size/computation costs while enabling secure blockchain multiparty computation.
	Raj et al. (2024) [30]	Developed a blockchain-based FHE framework for encrypted statistical computation via smart contracts.
	Wang et al. (2024) [31]	Combined Paillier encryption, ZKPs, and secondary encryption for secure, low-leakage transactions.
	Wu et al. (2024) [32]	Proposed FHE-based ciphertext computation integrated with Ethereum for secure, scalable computation.
	Tawfik et al. (2025) [33]	Proposed PriCollabAnalysis using Hyperledger Fabric, HE, and SMPC for privacy-preserving healthcare analytics.
Cybersecurity Frameworks Incorporating Blockchain/Cryptography		
Blockchain-integrated IDS for IoT/IIoT	Nandanwar & Katarya (2025) [34]	Hybrid blockchain-IoT IDS using ECC, DSA, SHA-512, SADE, and PBFT for privacy, authentication, and trust.
	Alharbi et al. (2024) [35]	BC-IDSs used an Ethereum-based shared IoMT malicious-IP blacklist; improved scalability by 50%.
	Nishad et al. (2026) [5]	Lightweight blockchain-FL intrusion prevention for IIoT using PoT, BFT, and communication optimization; 97.8% accuracy.
	Kumar et al. (2025) [36]	Blockchain + federated-style learning IDS using lightweight cryptography and SHAP-XAI for DDoS, ransomware, and injection attacks.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

Parameter	Study	Brief finding
	Aljumah (2025) [37]	Combined SDN, blockchain, and edge computing; achieved 98.7% accuracy, 1.2% FPR, and 101.1 ms response.
Zero trust & decentralized identity	Nie et al. (2025) [38]	Blockchain-based Zero Trust IoT/6G using smart contracts, reputation, and IPE for secure access.
	Yousseef et al. (2025) [39]	Hyperledger Iroha-based Zero Trust decentralized identity for autonomous vehicles; supports continuous verification and V2X security.
	Ahmadi (2025) [40]; Daah et al. (2024) [41]	Examined DID/VC-based ZTA, enabling privacy-preserving authentication and least-privilege access with reduced centralization.
	Xevgenis et al. (2026) [42]	Blockchain-based decentralized identity for Zero-Touch 6G networks using DIDs, VCs, ZKPs, and oracles.
	Singh et al. (2025) [9]; Alevizos et al. (2022) [43]	Blockchain-enabled Zero Trust FinTech using Ethereum, MFA, RBAC, and just-in-time authorization for continuous verification and auditing.
Identity & access management surveys	Ghaffari et al. (2022) [44]	Surveyed blockchain-based IAM and SSI, emphasizing privacy, decentralization, and elimination of single points of failure.
	Babel & Sedlmeir (2023) [45]	Evaluated ZKP languages for digital identity wallets; demonstrated zk-SNARKs for anonymous credentials and selective disclosure.

3.1 Summary of Research Gap

The reviewed literature on Blockchain, Cryptography, and Cybersecurity demonstrates that each area has been widely explored in isolation and in select combinations. Blockchain-related studies primarily focused on addressing consensus, smart-contract, Sybil, and identity attacks [2,7,12,13]. In turn, cryptographic research explored post-quantum cryptography, zero-knowledge proofs, and homomorphic encryption to ensure privacy and support secure computations [3,24,29]. Finally, cybersecurity studies utilized blockchain- or cryptographic-based solutions to detect and respond to intrusions, implement Zero Trust architecture, and utilize decentralized identity management [5,34,38,42]. However, the literature lacks research that integrates Blockchain, Cryptography, and Cybersecurity to form a system that utilizes the combined potential of decentralized trust, cryptographic security, and dynamic cybersecurity intrusion detection and response. Therefore, the key research gap is a blockchain-based solution that employs cryptographic methods and cybersecurity measures to create an end-to-end security system.

4. The Converged Framework

4.1 Conceptual Model

The converged framework is comprised of three layers, namely the Cryptographic Layer, Blockchain Layer, and Cybersecurity Layer, as shown in Figure 6 below. This model differs from the previous ones as it allows for interactions and interdependencies between the layers. For instance, the cybersecurity layer provides feedback to the cryptographic layer in case of any detected attack, and the blockchain layer is also provided with feedback from the cybersecurity layer [46].

However, for independent layers, security can be expressed as:

$$S(system) \approx S(cryptographic) + S(blockchain) + S(cybersecurity) \quad (2)$$

For interdependent layers, the security is given by:

$$S(system) = f(S(cryptographic), S(blockchain), S(cybersecurity), I_{12}, I_{23}, I_{31}) \quad (3)$$

where I_{12}, I_{23}, and I_{31} represent interactions among the layers.

The cryptographic layer provides hash functions, Public Key Infrastructure (PKI), digital signatures, Zero Knowledge Proof (ZKP), and post-quantum cryptography. The blockchain layer, on the other hand, implements hashing, signatures, ZKP, PKI, smart contracts, and consensus mechanisms. Lastly, the cybersecurity layer offers threat detection, key rotation, access control, monitoring, and response mechanisms such as blacklisting nodes and suspending smart contracts.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

4.2 Interaction Between Layers

The interactions and data flows between the three layers of the converged framework are depicted below.

- **Cryptographic layer to blockchain layer:** This involves hashing and signature functions.
- **Blockchain layer to cybersecurity layer:** This is achieved through audit trails and smart contracts.
- **Cybersecurity layer to cryptographic layer:** This interaction is manifested through key rotation or migration.
- **Cybersecurity layer to blockchain layer:** This occurs through blacklisting nodes, validator responses, and pausing smart contracts.

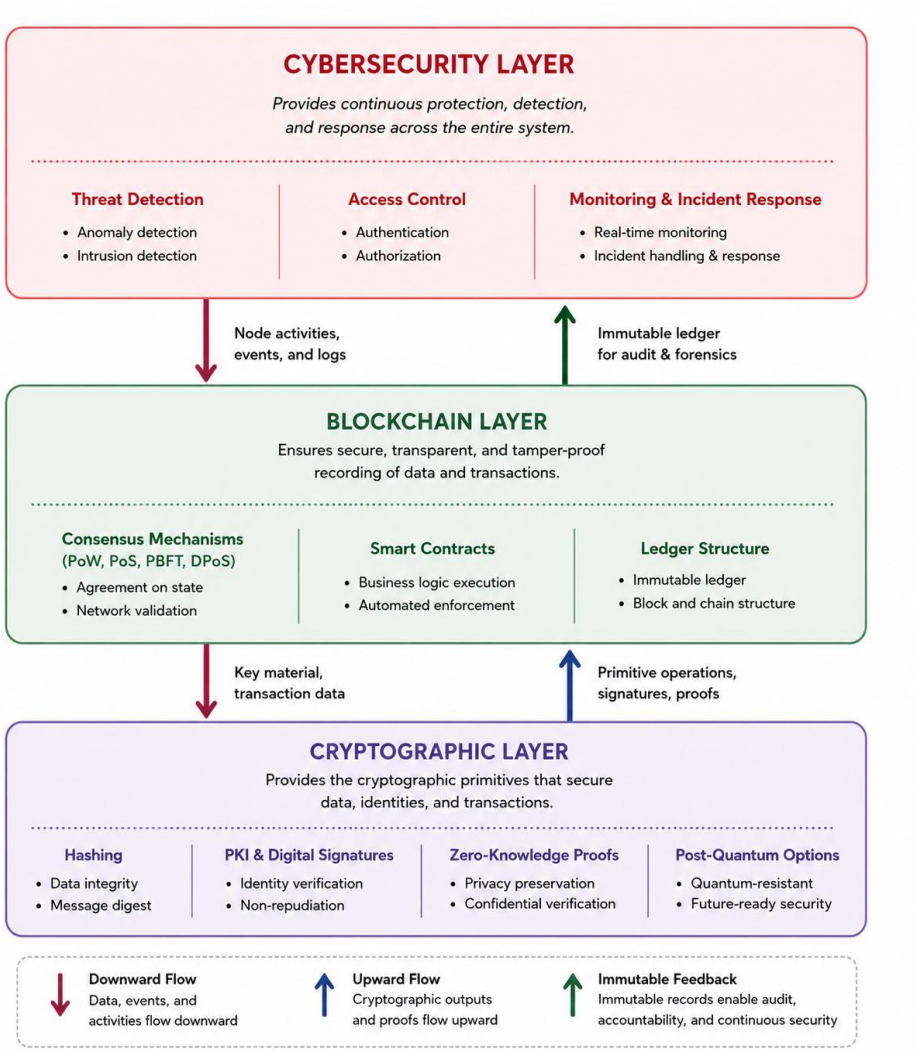


Figure 6. Three-Layer Architecture of the Converged Framework

5. Threat Model and Security Analysis

This section describes the adversary model, defines the attack surface, and provides a semi-formal security justification.

5.1 Adversary Model and Assumptions

The framework is designed under the threat model that incorporates a number of assumptions about blockchain and distributed-system security. An adversary can observe, inject, alter, delay, or replay messages in transit, take over some number of Byzantine nodes in the network, exploit smart contracts, and launch endpoint attacks to capture private keys. However, an adversary is limited in his computational power and cannot break the cryptographic primitives employed by the framework. The framework assumes that the majority of network participants act honestly, cryptography is secure, and cybersecurity-related network anomalies can be

identified within a bounded period of time. It focuses on securing endpoints and organizational networks by relying on access control and monitoring techniques.

5.2 Attack Surface

Table 5. Attacks Addressed by Different Layers of Framework

Attack	Mitigating Layer(s)	Key Mitigation
Sybil	Blockchain + Cybersecurity	Consensus cost + anomaly detection
51%	Blockchain + Cybersecurity	Consensus + hash/stake monitoring
Double-spending	Blockchain + Cryptographic	Finality + signatures
MITM	Cryptographic + Cybersecurity	PKI + intrusion detection
Replay	Cryptographic	Nonces + transaction-bound signatures
Smart contracts	Blockchain + Cybersecurity	Auditing + monitoring/pausing
Key theft	Cryptographic + Cybersecurity	Access control + key rotation
Data tampering	Cryptographic + Blockchain	Hashing + immutable ledger
Insider threats	Cybersecurity	Access control + monitoring

Most attacks therefore require multi-layered defense, spanning more than one category.

5.3 Security Properties

The framework offers the following security capabilities:

- **Confidentiality**: Cryptographic masks such as encryption and ZKP combined with cybersecurity-centric access control mechanisms.
- **Integrity**: Hashing algorithms, Merkle trees, and the blockchain's inherent nature prevent data tampering.
- **Availability**: Decentralized consensus mechanisms with replication and cybersecurity auditing and monitoring reduce the possibility of denial of service.
- **Non-repudiation**: Digital signatures combined with the tamper-proof audit trail built into the blockchain provide non-repudiation of transactions.
- **Resiliency**: The feedback loop on the cross-layer level allows for dynamic adaptation, such as key rotation and node isolation, to respond to potential attacks and new vulnerabilities

5.4 Semi-Formal Justification

This subsection aims to shortly argue why the properties stated in Section 5.3 follow from the adversary model of Section 5.1, ensuring that the paper claims are valid.

- **Integrity**: Each block's hash is computed from its contents and the previous block's hash (cf. Section 2.2, Eq. 1). Therefore, changing any transaction's data will modify the corresponding block's hash, which renders the subsequent blocks' stored link value invalid, thus creating an inconsistency that can be detected at any later point in time.
- **Non-repudiation:** By the unforgeability property of Section 5.1 and integrity of this subsection, any transaction's signature can be checked against the signer's public key at any point in time after the signing, producing a verifiable proof of signature, thus preventing repudiation by the signer.
- **Resilience:** If any one of the three layers is compromised, the other two layers will detect and isolate the damage, thus ensuring resilience. In particular, compromised nodes are isolated by the consensus mechanism, by the honest Majority assumption, and the cybersecurity layer flags malicious activity, while the cryptographic layer's integrity and non-repudiation properties are guaranteed by the underlying cryptographic assumptions. If the cybersecurity layer fails to timely detect the attack, the other layers still provide integrity and non-repudiation guarantees.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

6. Use Cases and Applications

The converged framework can be applied to different use cases depending on the level of confidentiality, accountability, throughput, and trust needed for a given application.

6.1 IoT Security

IoT networks that operate with low resources benefit from light cryptography, consortium/private blockchain, and PBFT consensus mechanisms. Cybersecurity offers intrusion detection and access control, while feedback from other layers can help in isolating compromised devices. The main threats are Sybil attacks, data tampering, and insider attacks.

6.2 Financial Systems and CBDCs

Financial systems and CBDCs require confidentiality, accountability, high throughput, and non-repudiation. ZKPs can be utilized to achieve confidentiality, while consortium blockchain and PBFT or permissioned PoS deliver accountability and throughput. Cybersecurity monitors fraudulent activities, double-spending, and other irregular transactions, while signatures offer transaction-level proof of authorization.

6.3 Healthcare Data Sharing

Healthcare data sharing applications require confidentiality and compliance with regulatory requirements. Encrypted data with controlled decryption can offer confidentiality, while blockchain can store hashes, permissions, and consents. Cybersecurity can manage access control and monitor the network for unauthorized activities.

6.4 Digital Identity Management

Digital identity management applications utilize ZKPs and blockchain to offer selective disclosure and tamper-proof identity information. Cybersecurity helps protect the endpoints and mitigates risks such as phishing, key compromise, and impersonation,

6.5 Supply Chain Integrity

Supply chains can utilize blockchain and cryptography to ensure non-repudiation and custody recording of shipments. Consortium blockchain can be utilized to allow only approved entities to participate in the network. Cybersecurity monitors the network for irregular activities and ensures organizational access control to shipment records.

7. Implementation

To show the practical feasibility of the converged framework proposed in Section 5, this section describes its implementation for the IoT security use case, discussed in Section 6.1, the algorithms used in each layer, and the experimental results.

7.1 Tools and Platform

The proof-of-concept was implemented in Python 3 using the following components for the framework's three layers that were discussed in Section 4.2:

- **Cryptographic Layer:** The cryptography library (v46), using Ed25519 for digital signatures and SHA-256 for hashing has been used.
- **Blockchain Layer:** A custom-implemented hash-chained ledger with a simulated PBFT-style quorum mechanism, where 2f+1 of N nodes, with $f = \lfloor (N-1)/3 \rfloor$, must verify and approve a block before it is committed, has been considered.
- **Cybersecurity Layer:** A custom monitoring module for replay-attack detection (nonce tracking) and Sybil-pattern detection (registration-burst rate limiting), with a cross-layer response that isolates flagged devices, has been used.
- **Environment:** Single-machine simulation has been used.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

7.2 Experimental Setup

The experiments used the following setup:

- **Network Configuration:** 7 simulated blockchain nodes with a PBFT-style quorum requirement of 5 nodes.
- **Workload:** 20 simulated IoT devices each submitting 10 signed transactions, giving 200 transactions, grouped into blocks of 5 transactions.
- **Attacks:**
  - **Replay attack:** A previously committed, validly signed transaction was resubmitted to the cybersecurity layer.
  - **Sybil-style registration burst:** 12 device registrations were submitted rapidly against a threshold of 5 registrations within 1 second.
  - F**alse-positive check:** 4 legitimate device registrations were submitted within the same window to check for false positives.
- **Metrics collected:** Signature generation time, block commit latency, end-to-end throughput, replay-detection time, Sybil-burst detection time, and false-positive count.

7.3 Algorithmic Design

Each of the layer's core operations was implemented as the following algorithms, which jointly describe cryptographic, blockchain, and cybersecurity mechanisms and their interactions.

- ***Algorithm 1*** implements the cryptographic layer's key generation, signing, and verification schemes.
- ***Algorithm 2*** implements the blockchain layer's transaction signing and block construction mechanisms.
- ***Algorithm 3*** implements the blockchain layer's block proposal and PBFT-inspired consensus mechanism, which utilizes the primitives defined in Algorithm 1.
- ***Algorithm 4*** implements the cybersecurity layer's replay-attack prevention mechanism, which acts as a transaction validator before Algorithm 3.
- ***Algorithm 5*** implements the cybersecurity layer's Sybil-pattern detection mechanism that analyzes device registrations for suspicious spikes.

7.4 Results

Table 6. Implementation results

Metric	Result	Layer(s) Involved
Devices / total transactions	20 devices / 200 transactions	—
Nodes / PBFT-style quorum	7 nodes/quorum of 5	Blockchain
Avg. signature generation time (Algorithm 1)	0.054 ms/transaction	Cryptographic
Avg. block commit latency (Algorithm 3, 7-node verification)	4.25 ms/block	Blockchain
End-to-end throughput (single-machine simulation)	1,173 tx/s	Blockchain + Cryptographic
Replay attack detection time (Algorithm 4)	0.005 ms	Cybersecurity
Replayed transaction reached the ledger?	No, blocked before consensus	Cybersecurity
Sybil-style burst (12 registrations) flagged at (Algorithm 5)	Registration #5 of 12	Cybersecurity
False positives on legitimate sub-threshold burst (4 registrations)	0	Cybersecurity

These results demonstrate the cross-layer interaction described in Section 4.2: Algorithm 4's replay detection prevents the attacked transaction from being committed while legitimate transactions are processed with sub-millisecond cryptographic overhead to achieve consensus.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

Algorithm 1: CryptographicLayer Operations

```
Input:  message M, private key sk, public key pk, signature σ
Output: key pair (sk, pk) | signature σ | boolean valid

Procedure GenerateKeyPair():
    sk ← Ed25519.GenerateKey()
    pk ← DerivePublicKey(sk)
    return (sk, pk)

Procedure Sign(sk, M):
    σ ← Ed25519.Sign(sk, M)
    return σ

Procedure Verify(pk, σ, M):
    try:
        Ed25519.VerifySignature(pk, σ, M)
        return TRUE
    catch InvalidSignature:
        return FALSE

Procedure Hash(data):
    return SHA256(data)
```

Algorithm 2: BuildTransactionAndBlock

```
Input:  device_id d, payload p, nonce n, private key sk,
        block index i, previous block hash h_prev,
        transaction list T
Output: signed transaction tx | block B with computed hash

Procedure CreateTransaction(d, p, n, sk):
    tx.device_id ← d
    tx.payload ← p
    tx.nonce ← n
    tx.timestamp ← CurrentTime()
    M ← Concatenate(d, p, n)          // message bound by the signature
    tx.signature ← Sign(sk, M)
    return tx

Procedure ComputeBlockHash(i, h_prev, T):
    tx_repr ← SerializeSorted(T)       // canonical representation
    return Hash(Concatenate(i, h_prev, tx_repr))

Procedure CreateBlock(i, h_prev, T):
    B.index ← i
    B.prev_hash ← h_prev
    B.transactions ← T
    B.hash ← ComputeBlockHash(i, h_prev, T)
    return B
```

Algorithm 3: ProposeBlock (PBFT-style quorum)

```
Input:  transaction list T, chain C, node count N
Output: committed block B, or NULL if quorum not reached

1:  f ← ⌊(N − 1) / 3⌋
2:  quorum ← 2f + 1
3:  prev ← Last(C)
4:  B ← CreateBlock(Length(C), prev.hash, T)
5:  votes ← 0
6:  for node = 1 to N do
7:       ok ← TRUE
8:       for each tx in T do
9:            if Verify(tx.pubkey, tx.signature, MessageBytes(tx)) = FALSE:
10:               ok ← FALSE
11:      if ok = TRUE:
12:           votes ← votes + 1
13: if votes ≥ quorum:
14:      Append(C, B)
15:      return B
16: else:
17:      return NULL
```

Algorithm 4: CheckReplay

```
Input:  transaction tx, seen-nonce set S
Output: boolean is_replay (S updated as a side effect)

1:  key ← (tx.device_id, tx.nonce)
2:  if key ∈ S:
3:       return TRUE                  // flagged as replay
4:  else:
5:       S ← S ∪ {key}
6:       return FALSE                 // accepted, now recorded
```

Algorithm 5: RecordRegistration

```
Input:  device_id d, event log E, time window w, threshold θ
Output: boolean is_flagged (E updated as a side effect)

1:  t_now ← CurrentTime()
2:  E ← E ∪ {(d, t_now)}
3:  recent ← { (d', t') ∈ E : t_now − t' ≤ w }
4:  if |recent| ≥ θ:
5:       return TRUE                  // burst flagged
6:  else:
7:       return FALSE
```

Figure 7. Implementation Algorithms

7.5 Sample Workflow

The following steps were executed in the workflow captured in the results above:

1. **Device registration (Algorithm 1):** 20 simulated devices created an Ed25519 keypair using GenerateKeyPair() (avg pDevice: 0.32 ms)
2. **Transaction signing (Algorithms 1-2):** Each device signed its transactions with CreateTransaction(), which internally invokes Sign() (avg Tx: 0.054 ms)
3. **Cybersecurity pre-check (Algorithm 4):** Before consensus, each transaction was checked against the set of known (device, nonce) pairs using CheckReplay()
4. **Consensus (Algorithm 3):** Groups of signed transactions were proposed to all 7 nodes for voting via ProposeBlock(); proposals achieving majority vote (5/7) were committed (avg Block: 4.25 ms)
5. **Attack:** One committed transaction was replayed
6. **Detection and response (Algorithm 4 -> Algorithm 3 feedback, Section 5.3):** CheckReplay() detected the duplicate nonce in 0.005 ms; the replayed transaction was excluded from the next call to ProposeBlock() and the attacker isolated.

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

8. Evaluation and Discussion

8.1 Performance Metrics

Table 6 of Section 7.4 summarizes the performance metrics obtained from the implementation, including the layer responsible for each result. Cryptographic and cybersecurity overhead is under a millisecond per transaction, and thus presents negligible overhead compared to the block commit latency. In other words, the security layer was not observed to impact throughput at this scale.

8.2 Comparison with Non-Converged Approaches

A blockchain-only framework with no cybersecurity layer was tested against the same replay attack. It successfully processed the double-spending transaction a second time. Meanwhile, the converged framework rejected it before consensus. This is an example of a value provided by the converged framework that neither layer could provide alone.

8.3 Strengths

- **Proven cross-layer value**: The results demonstrate that the cybersecurity layer can prevent attacks that would otherwise be unsuccessful against the blockchain layer.
- **Low overhead**: The cryptographic and cybersecurity operations add less than a millisecond to each transaction, ensuring that the benefits of enhanced security do not come at the expense of efficiency, at least at the scale of our experiments.
- **Consensus and platform agnostic:** The two layers interact through well-defined interfaces but are otherwise independent and can be implemented on top of different consensus mechanisms and blockchain platforms.
- **Inherent auditability:** All cross-layer operations are encoded in transactions on the blockchain and hence are immutable and publicly available for inspection.
- **Low false-positive rate:** The Sybil detection mechanism was able to distinguish between legitimate and adversarial registration attempts and did not trigger at rates below the threshold set by the adversary's traffic. Nevertheless, it remains important to evaluate how the cybersecurity layer will perform against more realistic and diverse traffic patterns.

8.4 Limitations and Trade-offs

- **Scalability:** While 7 nodes are sufficient for testing purposes, the PBFT consensus algorithm is known to be scaling-challenged due to the increased communication overhead with every additional node.
- **Quantum resistance:** The Ed25519 signing algorithm is not protected against quantum computing attacks.
- **Regulatory compliance:** Blockchain immutability may violate GDPR requiring data deletion upon request.

8.5 Future Research Directions

- **Scalability vs. security trade-offs:** The proof-of-concept was tested at a small scale (7 nodes, single machine). PBFT-style consensus is known to scale poorly in communication overhead as node count grows, and zero-knowledge proof generation and cybersecurity monitoring both introduce costs that increase with network size. Jointly optimizing consensus, cryptographic, and monitoring overhead; rather than tuning each independently; remains open.
- **Post-quantum readiness:** The Ed25519 signatures used in the implementation are not resistant to quantum adversaries. While the framework's layered design allows post-quantum primitives to be substituted into the cryptographic layer without redesigning the other two, this paper did not measure the resulting performance impact, nor did it address migration strategy for networks already in production or the long-term risk of "harvest now, decrypt later" attacks against non-repudiation.
- **Regulatory and interoperability issues:** Blockchain immutability conflicts with regulations granting a right to erasure (e.g., GDPR), particularly relevant to the healthcare use case in this paper. Cross-

chain interoperability across independently operated consortium deployments, such as in the supply chain use case, also remains unaddressed, as does regulatory fragmentation across jurisdictions for cross-border deployments such as CBDCs.

- **Standardization gaps:** No standardized interface currently governs how the cryptographic, blockchain, and cybersecurity layers interact. Detection thresholds, incident response procedures, and cross-layer feedback protocols are implemented ad hoc by individual systems, including the proof-of-concept in this paper, making frameworks difficult to compare or adopt without bespoke integration work. Establishing standardized interaction protocols is a necessary step toward interoperable, production-ready converged security architectures.

9. Conclusion

This paper set out to address the fragmentation of blockchain, cryptography, and cybersecurity into separate research tracks, despite their growing interdependence in practice. To address this, it proposed a converged, three-layer framework connected through explicit cross-layer feedback rather than sequential dependency, and evaluated it through a defined threat model, a semi-formal security justification, and a working proof-of-concept. The empirical results substantiate the paper's central claim directly: under an identical replay attack, a blockchain-only baseline accepted a duplicate transaction, while the converged framework blocked it, concrete evidence that security value emerges from the interaction between these three domains, not from any one in isolation. The framework's relevance was further demonstrated across IoT security, financial systems, healthcare data sharing, digital identity, and supply chain integrity, each drawing on the three layers in different proportions. Scalability at production scale, post-quantum migration, regulatory tensions, and the absence of standardized cross-layer interfaces remain open, as discussed in Section 9. As digital systems increasingly depend on decentralized trust, cryptographic assurance, and active threat response simultaneously, treating blockchain, cryptography, and cybersecurity as a single, mutually reinforcing discipline, rather than three separate ones, offers a stronger foundation than current siloed approaches, and this framework is offered as a step toward that goal.

Author Contributions

Tanishka Pahwa conceptualized and framed the manuscript and prepared the original draft. Geet Bawa carried out the experimentation.

References

[1] Bawa, G., & Singh, H. (2026). Revolutionizing organ transplantation logistics through blockchain, smart contracts, and Internet of Medical Things-driven system. *Indian Journal of Transplantation, 20*(2), 183–195. https://doi.org/10.4103/ijot.ijot_118_25

[2] Yakubu, M. M., Hassan, M. F. B., Danyaro, K. U., Junejo, A. Z., Siraj, M., Yahaya, S., Adamu, S., & Abdulsalam, K. (2024). A systematic literature review on blockchain consensus mechanisms' security: Applications and open challenges. *Computer Systems Science and Engineering, 48*(6), 1437–1481. https://doi.org/10.32604/csse.2024.054556

[3] Chhetri, S. R., Somvanshi, D., Hebli, P., Brotee, S., & Das, A. K. (2025). Post-quantum cryptography: Challenges, opportunities, and emerging applications. *IEEE Access, 13*, 120345–120389. https://doi.org/10.1109/ACCESS.2025.3521847

[4] Ahmed, N., Zhang, L., & Gangopadhyay, A. (2025). A survey of post-quantum cryptography support in cryptographic libraries. In *Proceedings of the 2025 IEEE International Conference on Quantum Computing and Engineering (QCE 2025)* (pp. 906–917). IEEE. https://doi.org/10.1109/QCE65121.2025.00102

[5] Nishad, D. K., Singh, R., & Khalid, S. (2026). A lightweight blockchain-enabled federated intrusion prevention framework for resource-constrained industrial IoT devices to detect and mitigate emerging cyberattacks. *Journal of Cloud Computing, 15*, Article 24. https://doi.org/10.1186/s13677-026-00843-3

[6] Bawa, G., & Singh, H. (2025). A framework for IoT-enabled transportation of transplantable organs using smart contracts. In *Proceedings of the 3rd International Conference on Signal and Data Processing* (pp. 517–533). Springer Nature Singapore. https://doi.org/10.1007/978-981-97-9578-9_45

[7] Liu, Q., Huang, Q., Fan, F., Wu, H., & Tang, X. (2025). Detecting Sybil addresses in blockchain airdrops: A subgraph-based feature propagation and fusion approach. In *Proceedings of the 2025 IEEE International*

Conference on Blockchain and Cryptocurrency (ICBC) (pp. 1–7). IEEE. https://doi.org/10.1109/ICBC64466.2025.11185061

[8] Arfaoui, S., & El-Gayar, O. F. (2025). Systematic review of lattice-based cryptography algorithms for securing blockchain networks in the post-quantum era. *Issues in Information Systems, 26*(4), 443–461. https://doi.org/10.48009/4_iis_2025_136

[9] Singh, A., Pareek, V., & Sharma, A. (2025). Blockchain-enabled zero trust framework for securing FinTech ecosystems against insider threats and cyber attacks. *arXiv*. https://doi.org/10.48550/arXiv.2507.19976

[10] Bordeianu, A. A., & Popescu, D. E. (2025). Blockchain variables and possible attacks: A technical survey. *Computers, 14*(12), Article 567. https://doi.org/10.3390/computers14120567

[11] Siam, M. K., Saha, B., Hasan, M. M., Faruk, M. J. H., Anjum, N., Tahora, S., Siddika, A., & Shahriar, H. (2025). Securing decentralized ecosystems: A comprehensive systematic review of blockchain vulnerabilities, attacks, and countermeasures and mitigation strategies. *Future Internet, 17*(4), Article 183. https://doi.org/10.3390/fi17040183

[12] Jiang, S., You, W., Xuan, S., & Shen, J. (2026). Decentralized finance security: A survey of attacks, defenses, and open challenges. *High-Confidence Computing, 6*(2), Article 100383. https://doi.org/10.1016/j.hcc.2026.100383

[13] De Baets, C., Suleiman, B., Chitizadeh, A., & Razzak, I. (2024). Vulnerability detection in smart contracts: A comprehensive survey. *arXiv*. https://doi.org/10.48550/arXiv.2407.07922

[14] Wei, Z., Sun, J., Zhang, Z., Zhang, X., Yang, X., & Zhu, L. (2024). Survey on quality assurance of smart contracts. *ACM Computing Surveys, 57*(2), Article 32, 1–36. https://doi.org/10.1145/3695864

[15] Wu, J., Xie, L., & Li, X. (2025). Security vulnerabilities in Ethereum smart contracts: A systematic analysis. *arXiv*. https://doi.org/10.48550/arXiv.2504.05968

[16] Hejazi, N., & Lashkari, A. H. (2025). A comprehensive survey of smart contracts vulnerability detection tools: Techniques and methodologies. *Journal of Network and Computer Applications, 237*, Article 104142. https://doi.org/10.1016/j.jnca.2025.104142

[17] Salzano, F., Antenucci, C. K., Scalabrino, S., Rosa, G., Oliveto, R., & Pareschi, R. (2025). An empirical analysis of vulnerability detection tools for Solidity smart contracts using line level manually annotated vulnerabilities. *arXiv*. https://doi.org/10.48550/arXiv.2505.15756

[18] Yuan, H., Yu, L., Huang, Z., Zhang, J., Lu, J., Cheng, S., Yang, L., Zhang, F., Ma, J., & Zuo, C. (2025). MOS: Towards effective smart contract vulnerability detection through mixture-of-experts tuning of large language models. *arXiv*. https://doi.org/10.48550/arXiv.2504.12234

[19] Platt, M., Platt, D., & McBurney, P. (2024). Sybil attack vulnerability trilemma. *International Journal of Parallel, Emergent and Distributed Systems, 39*(4), 446–460. https://doi.org/10.1080/17445760.2024.2352740

[20] Erfan, F., Bellaïche, M., & Halabi, T. (2026). Sybil attack defense in blockchain-based industrial IoT systems using decentralized federated learning. *Internet of Things, 37*, Article 101927. https://doi.org/10.1016/j.iot.2026.101927

[21] Tej, Y. V., & Naidu, K. S. E. K. (2024). Preventing Sybil attacks in blockchain using distributed behavior monitoring and integrity validation of miners. *International Journal of Computer Science Review, 1*(1), 12–20. https://doi.org/10.5281/zenodo.14440824

[22] Sezer, B. B., & Akleylek, S. (2026). Lattice-based blockchain platform for IoT: Privacy-enhanced application with lattice-based blind signatures. *Computer Standards & Interfaces, 96*, Article 104077. https://doi.org/10.1016/j.csi.2025.104077

[23] Faridi, A. R., Masood, F., Shamsan, A. H. T., Luqman, M., & Salmony, M. Y. (2022). Blockchain in the quantum world. *International Journal of Advanced Computer Science and Applications, 13*(1), 637–652. https://doi.org/10.14569/IJACSA.2022.0130167

[24] Shashidhara, R., Chirakarotu Nair, R., & Kumar Panakalapati, P. (2025). Promise of zero-knowledge proofs (ZKPs) for blockchain privacy and security: Opportunities, challenges, and future directions. *Security and Privacy, 8*(1), Article e461. https://doi.org/10.1002/spy2.461

[25] Gupta, S. (2025). Zero-knowledge proofs for privacy-preserving systems: A survey across blockchain, identity, and beyond. *Engineering and Technology Journal, 10*(7), 5755–5761. https://doi.org/10.47191/etj/v10i07.23

[26] Oude Roelink, B., El-Hajj, M., & Sarmah, D. (2024). Systematic review: Comparing zk-SNARK, zk-STARK, and bulletproof protocols for privacy-preserving authentication. *Security and Privacy, 7*(5), Article e401. https://doi.org/10.1002/spy2.401

[27] Zhou, L., Diro, A. A., Saini, A., Kaisar, S., & Pham, C. H. (2024). Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities. *Journal of Information Security and Applications, 80*, Article 103678. https://doi.org/10.1016/j.jisa.2023.103678

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access

[28] Malatrait, S., & Sirac, A. (2025). FibRace: A large-scale benchmark of client-side proving on mobile devices. *arXiv*. https://doi.org/10.48550/arXiv.2510.14693

[29] Bao, H., Yuan, M., Deng, H., Xu, J., & Zhao, Y. (2024). Secure multiparty computation protocol based on homomorphic encryption and its application in blockchain. *Heliyon, 10*(14), Article e34458. https://doi.org/10.1016/j.heliyon.2024.e34458

[30] Raj, R., Peker, Y. K., & Mutlu, Z. D. (2024). Blockchain and homomorphic encryption for data security and statistical privacy. *Electronics, 13*(15), Article 3050. https://doi.org/10.3390/electronics13153050

[31] Wang, G., Li, C., Dai, B., & Zhang, S. (2024). Privacy-protection method for blockchain transactions based on lightweight homomorphic encryption. *Information, 15*(8), Article 438. https://doi.org/10.3390/info15080438

[32] Wu, X., Wang, J., & Zhang, T. (2024). Integrating fully homomorphic encryption to enhance the security of blockchain applications. *Future Generation Computer Systems, 161*, 467–477. https://doi.org/10.1016/j.future.2024.07.015

[33] Tawfik, A. M., Al-Ahwal, A., Tag Eldien, A. S., & Zayed, H. H. (2025). PriCollabAnalysis: Privacy-preserving healthcare collaborative analysis on blockchain using homomorphic encryption and secure multiparty computation. *Cluster Computing, 28*, Article 191. https://doi.org/10.1007/s10586-024-04928-z

[34] Nandanwar, H., & Katarya, R. (2025). A hybrid blockchain-based framework for securing intrusion detection systems in Internet of Things. *Cluster Computing, 28*(7), Article 471. https://doi.org/10.1007/s10586-025-05135-0

[35] Alharbi, S., Alghazzawi, D., Hakeem, A., Mohaisen, L., Cheng, L., & Attiah, A. (2024). A blockchain-based collaborative intrusion detection systems framework. *IEEE Internet of Things Journal, 11*(15), 26765–26778. https://doi.org/10.1109/JIOT.2023.3347492

[36] Kumar, A., Sharma, B., & Noonia, A. (2025). Secure blockchain based intrusion detection for IoT networks. *Discover Computing, 28*, Article 226. https://doi.org/10.1007/s10791-025-09754-4

[37] Aljumah, A. (2025). Blockchain-inspired distributed security framework for Internet of Things. *Scientific Reports, 15*, Article 10066. https://doi.org/10.1038/s41598-025-93690-2

[38] Nie, S., Ren, J., Wu, R., Han, P., Han, Z., & Wan, W. (2025). Zero-trust access control mechanism based on blockchain and inner-product encryption in the Internet of Things in a 6G environment. *Sensors, 25*(2), Article 550. https://doi.org/10.3390/s25020550

[39] Yousseef, A., Satam, S., Saber Latibari, B., Abdel-Malek, M., Salehi, S., & Satam, P. (2025). Zero Trust-based decentralized identity management system for autonomous vehicles. *arXiv*. https://doi.org/10.48550/arXiv.2509.25566

[40] Ahmadi, S. (2025). Distributed identity for zero trust and segmented access control: A novel approach to securing network infrastructure. *International Journal of Advanced Computer Science and Applications, 16*(3), 11–21. https://doi.org/10.14569/IJACSA.2025.0160302

[41] Daah, C., Qureshi, A., Awan, I., & Konur, S. (2024). Enhancing zero trust models in the financial industry through blockchain integration: A proposed framework. *Electronics, 13*(5), Article 865. https://doi.org/10.3390/electronics13050865

[42] Xevgenis, M. G., Polychronaki, M., Kogias, D. G., Leligkou, H. C., & Liotou, E. (2026). Securing zero-touch networks with blockchain: Decentralized identity management and oracle-assisted monitoring. *Electronics, 15*(2), Article 266. https://doi.org/10.3390/electronics15020266

[43] Alevizos, L., Eiza, M. H., Ta, V. T., Shi, Q., & Read, J. (2022). Blockchain-enabled intrusion detection and prevention system of APTs within Zero Trust Architecture. *IEEE Access, 10*, 89270–89288. https://doi.org/10.1109/ACCESS.2022.3200165

[44] Ghaffari, F., Gilani, K., Bertin, E., & Crespi, N. (2022). Identity and access management using distributed ledger technology: A survey. *International Journal of Network Management, 32*(2), Article e2180. https://doi.org/10.1002/nem.2180

[45] Babel, M., & Sedlmeir, J. (2023). Bringing data minimization to digital wallets at scale with general-purpose zero-knowledge proofs. *arXiv*. https://doi.org/10.48550/arXiv.2301.00823

[46] Bawa, G., Singh, H., Rani, S., Kataria, A., & Min, H. (2025). Smart traceable framework for transportation of transplantable organs using IPFS, IoT, and smart contracts. Scientific Reports, 15(1), 23364. https://doi.org/10.1038/s41598-025-06471-2

© 2026 IJCRETM | Published by A2Z EduLearningHub, India | Open Access