AI–Blockchain Integration for Real-Time Cyber Security - System Design and Evaluation

Sam Goundar * and Iqbal Gondal

Posted Date: 15 July 2025

doi: 10.20944/preprints202507.1203.v1

Keywords: AI-based threat detection; blockchain; smart contract; cyber security; metadata integrity; real-time system auditability; anomaly detection

 doi:10.20944/preprints202507.1203.v1

Article

AI-Blockchain Integration for Real-Time Cyber Security: System Design and Evaluation

Sam Goundar * and Iqbal Gondal

RMIT University, Melbourne, Australia

* Correspondence: sam.goundar@rmit.edu.vn

Abstract

This paper presents the design, development, and thorough evaluation of a novel network security prototype that integrates Artificial Intelligence (AI) and blockchain technology to significantly enhance cyber security. As AI becomes increasingly embedded in cybersecurity solutions, ensuring the provenance, accountability, and integrity of AI-generated decisions has emerged as a critical challenge. Without reliable logging mechanisms, AI models remain vulnerable to adversarial manipulation and pose significant risks to critical security infrastructure. To address this, our research combines a state-of-the-art Convolutional Neural Network (CNN)-based threat detection module with a permissioned Ethereum-compatible blockchain. A custom-designed Solidity smart contract ensures secure, structured storage of comprehensive AI model metadata, while interactions with the blockchain are seamlessly managed through a lightweight Flask-based REST API. Each recorded transaction generates a unique cryptographic fingerprint, providing robust evidence for audits and forensic analyses. We evaluated the system's effectiveness through rigorous experimentation on a controlled test network, confirming immutability, traceability, and verifiable integrity of all logged metadata entries. Results demonstrated significant improvements in anomaly detection accuracy, reduced false-positive rates, and ensured real-time responsiveness essential for effective intrusion prevention. Despite controlled-environment limitations, such as transaction latency and blockchain-related operational costs, our prototype successfully establishes proof-of-concept for leveraging blockchain as an immutable audit trail for AI-driven cybersecurity systems. Future research directions include integrating advanced scaling techniques, such as layer 2 solutions, and extending the blockchain logging capabilities to cover the entire AI model lifecycle, including detailed training logs and comprehensive version histories. This work provides foundational contributions towards building trusted, auditable, and transparent AI solutions in regulated cyber security domains.

Keywords: AI-based threat detection; blockchain; smart contract; cyber security; metadata integrity; real-time system auditability; anomaly detection

Introduction

Cyber security has emerged as a critical concern in the digital era due to increasingly sophisticated cyber threats targeting essential infrastructures, private organizations, and government institutions worldwide. Incidents involving ransomware, data breaches, and targeted cyber attacks have underscored the urgent need for robust security systems capable of detecting and preventing threats in real time [1,2]. To respond effectively to these threats, artificial intelligence (AI) and machine learning (ML) techniques have become indispensable tools in modern cyber security frameworks. These technologies provide advanced capabilities for real-time intrusion detection, anomaly detection, threat classification, and threat intelligence analysis by processing vast volumes of security data swiftly and accurately [3,4].

Despite their proven efficacy, AI-driven cyber security systems introduce several new challenges related to trust and reliability. AI models, particularly those based on deep learning algorithms, are

 doi:10.20944/preprints202507.1203.v1

vulnerable to adversarial attacks and data manipulations, which can compromise model integrity and produce misleading or inaccurate security alerts [5,6]. Additionally, biases introduced during model training or through improper updates can severely undermine the trustworthiness of AI systems, leading to erroneous decisions that may cause significant damage or compliance violations, especially within high-stakes environments like finance, healthcare, and government institutions [7,8].

Addressing these critical concerns requires ensuring not only the accuracy of AI models but also their integrity, provenance, and accountability. One promising solution to these problems is the integration of blockchain technology. Blockchain provides an immutable, decentralized, and transparent record-keeping mechanism that can significantly enhance trust in AI-driven cyber security systems by securely logging the provenance and operational history of AI models. Recent research by Malhotra, et al. demonstrated the capability of blockchain technology to ensure traceability of AI model activities [9]. Surveys conducted by Ressi, et al. and Martinez, et al. further underscored the potential for blockchain to address critical requirements for auditability and accountability within AI-enhanced cyber security systems [10,11]. However, despite these advances, current literature lacks practical studies demonstrating full end-to-end integration of blockchain-based logging within operational, real-time AI security workflows.

While recent studies have explored the integration of AI and blockchain for various security applications, most existing implementations remain at a conceptual or simulation level, lacking real-time operational validation [9–11]. Furthermore, many proposed architectures focus narrowly on single functionalities, such as logging or inference, without demonstrating comprehensive end-to-end systems that incorporate detection, decision logging, metadata anchoring, and security evaluation in a cohesive manner. There is also a noticeable gap in empirical evaluations using real-world or benchmark datasets that assess not only detection accuracy but also system latency, throughput, and security robustness under stress conditions. Additionally, few studies have addressed the practical challenges of deploying such systems in constrained environments, including the implications of blockchain-related transaction overheads, privacy trade-offs, and compliance with data protection regulations. This paper seeks to address these limitations by developing and testing a fully integrated, real-time prototype that combines a CNN-based threat detection model with a permissioned blockchain for secure and auditable metadata logging in cyber security contexts.

Motivated by this significant research gap, our study introduces a novel prototype integrating an advanced AI-based anomaly detection module with a permissioned Ethereum-compatible blockchain. The choice of Ethereum technology was driven by its proven reliability, scalability, and compatibility with advanced smart contract functionalities, which provide robust security and accountability features suitable for regulated and sensitive environments [12]. Our system securely logs comprehensive AI model metadata, including version identifiers, cryptographic data hashes, alert classifications, and timestamps. A dedicated Solidity smart contract ensures secure and structured metadata storage, while a Flask-based REST API facilitates smooth integration between AI modules and the blockchain. This combination significantly strengthens forensic traceability, transparency, and accountability of AI-driven decisions in cyber security operations.

The primary objectives of this research are to (1) demonstrate the feasibility of blockchain integration for securing AI metadata logs, (2) evaluate the prototype's effectiveness in ensuring data immutability and auditability, and (3) assess the performance trade-offs associated with blockchain integration within real-time cybersecurity contexts.

The remainder of this paper is structured as follows. Section 2 provides a detailed review of related work, identifying key research gaps and contributions in the field. Section 3 thoroughly describes the system architecture, including AI model characteristics, smart contract functionalities, and the integration framework. Section 4 presents comprehensive experimental results covering system performance, detection accuracy, latency measurements, and security assessment. Section 5 offers an in-depth discussion of the findings, including practical implications, limitations, and opportunities for improvement. Section 6 presents the main conclusions drawn from the study,

 doi:10.20944/preprints202507.1203.v1

highlighting its contributions to the field of AI and blockchain integration in cyber security. Finally, Section 7 outlines specific recommendations and future research directions aimed at improving scalability, privacy, and applicability in real-world deployments.

Literature Review

This section reviews the evolving research landscape surrounding the integration of artificial intelligence (AI) and blockchain in cyber security systems. It is structured into thematic subsections to synthesize current findings, identify limitations, and highlight critical gaps that motivate the need for the present work.

AI and Graph-Based Techniques in Cyber Security

The application of AI, particularly deep learning and graph-based techniques, has transformed cyber security by enabling scalable intrusion detection, anomaly analysis, and threat classification. Recent work by Ozkan-Okay et al. [4] demonstrates the effectiveness of machine learning techniques in detecting cyber threats with high accuracy across a variety of network environments. Similarly, Mohamed [3] presents a comprehensive review of AI models tailored for security use cases, emphasizing convolutional neural networks (CNNs), recurrent neural networks (RNNs), and hybrid frameworks. Graph-based approaches, including graph neural networks (GNNs) and graph embeddings, have been employed to model relational data in network traffic, enhancing the detection of complex attack patterns. However, these models often lack built-in mechanisms for post-prediction verification or tamper-evident logging, limiting their suitability in high-assurance environments.

Adversarial machine learning introduces further complexity, as attackers can craft subtle inputs to mislead AI systems. Javed et al. [5] show that deep learning models in critical applications such as healthcare and finance can be manipulated to produce incorrect outputs, raising serious trust concerns. While these vulnerabilities are well-documented, there has been limited progress in coupling adversarially robust AI systems with transparent, immutable logging mechanisms to ensure accountability and forensic traceability in real-world deployments.

AI-Powered Smart Contract Security

The vulnerability of smart contracts remains a major concern in blockchain systems, especially in decentralized applications (dApps) and financial networks. Researchers have proposed AI-driven tools to identify vulnerabilities in smart contract code. Jain and Mitra [2] developed a multi-layered model using transformers and convolutional layers to detect code-level exploits in Ethereum smart contracts with high precision. Malhotra et al. [9] extend this work by embedding explainability features into AI detection systems and anchoring validation results on-chain using blockchain-based proof-of-authenticity protocols. However, these models are often evaluated in static settings and are rarely integrated with live logging systems or operational security frameworks. Furthermore, they typically focus on the contracts themselves and do not extend into AI–blockchain interactions for broader cyber security applications.

Blockchain-Enabled AI for Intrusion Detection

A growing number of studies explore the use of blockchain to improve transparency and trust in AI-driven intrusion detection systems (IDS). For example, Aliyu et al. [14] proposes a collaborative framework in which blockchain securely stores intrusion alerts generated by distributed AI agents. Their system achieved over 92 percent accuracy and demonstrated low false-positive rates. In a healthcare context, Meherj, et al. [15] integrated blockchain with AI classifiers and IPFS for log storage, achieving near-perfect detection performance in Internet of Healthcare Things (IoHT) environments. However, these solutions often involve asynchronous logging or are applied post-analysis, meaning the AI decision itself is not verifiably recorded in real time. This limits auditability

 doi:10.20944/preprints202507.1203.v1

and exposes a trust gap, especially in regulated domains where traceable accountability is mandatory.

Recent advances demonstrate the growing trend towards real-time AI–blockchain integration in domain-specific applications such as autonomous vehicles and vehicular networks yet still fail to address cyber security scenarios comprehensively. For instance, Bendiab, et al. [20] design a framework that couples LSTM-based anomaly detection with on-chain logging for multi-sensor data streams in autonomous vehicles, enabling instantaneous response using smart contracts. Similarly, Anand et al. [21] apply CNN and LSTM models to secure Vehicular Ad Hoc Networks (VANETs), ensuring transparent, tamper-proof recording of alerts. While these systems illustrate feasibility, they remain domain-specific and have yet to be adapted to broader cyber security environments, particularly those involving network-wide threat detection using heterogeneous data streams.

Efforts to integrate blockchain with AI for industrial IoT and digital twin systems reveal powerful conceptual synergies, though practical security deployments remain underdeveloped. Benedictis, et al. [22] proposes a hybrid architecture that combines digital twins, AI, and blockchain to automate anomaly detection while incorporating privacy-preserving zero-knowledge proofs and smart contract logic for decentralized incident response. On a similar note, Thakur, et al. [23] deploy a lightweight DNN alongside private Ethereum smart contracts in a rural water management system, achieving usable latency and throughput. These studies confirm that AI-blockchain systems can be efficient and responsive in constrained environments, yet their focus is largely on physical-cyber infrastructure rather than on AI model integrity, provenance, or lifecycle traceability in enterprise-level cyber security systems.

Secure Data Sharing and Hybrid Architectures

Frameworks like AICyber-Chain have been introduced to support secure collaborative data sharing in cyber security, particularly across organizational boundaries [16]. These systems rely on blockchain to enforce access control, while AI processes threat intelligence. Hybrid storage architectures combining off-chain IPFS and on-chain metadata have been developed to balance efficiency and integrity. Nonetheless, challenges persist in maintaining synchronization between on-chain and off-chain data, ensuring low-latency responses, and securing private information. Kaur et al. [17] argue that trust in AI systems must extend beyond algorithmic performance to include transparency, auditability, and data provenance, areas where current systems fall short.

Regulatory, Privacy, and Ethical Considerations

The integration of blockchain with AI raises difficult questions about privacy and regulatory compliance. Blockchain's immutability clashes with regulations such as the General Data Protection Regulation (GDPR), which grants individuals the right to request data erasure. Albahri et al. [18] explores how privacy-preserving mechanisms like differential privacy, federated learning, and secure multiparty computation can be integrated into AI systems to enhance trustworthiness. Ressi et al. [10] argues for the use of zero-knowledge proofs and verifiable credentials within blockchain-based AI pipelines to mitigate compliance risks. However, most of these proposals remain theoretical or are tested in narrowly scoped simulations. Very few studies deploy such systems in a unified, operational AI–blockchain environment.

Another critical strand intersects privacy-preserving machine learning and blockchain, an essential concern in regulated environments. Mahato, et al. [24] introduce a novel framework that combines homomorphic encryption with federated learning and blockchain-based alert logging, delivering encrypted-model training alongside verifiable attack detection without compromising data privacy. This system achieves near-equivalent accuracy to unprotected models while ensuring cryptographic traceability. Despite the technical elegance of this approach, the integration of such secure learning protocols into real-time AI-blockchain logging for operational cyber security systems remains unexplored, particularly in environments where regulatory compliance and forensic audits are paramount.

 doi:10.20944/preprints202507.1203.v1

Gaps in Real-Time Integration and Lifecycle Auditing

Despite the richness of existing literature, several gaps remain. Most notably, few implementations anchor AI prediction outputs, metadata, and alert decisions directly onto a blockchain in real time. Existing models often rely on centralized logs or external repositories, which are susceptible to tampering and offer limited forensic integrity [11,12]. This undermines the credibility of AI-driven responses in high-risk environments. While blockchain is occasionally used for batch logging or data timestamping, true real-time synchronization of AI outputs with blockchain-based storage is rarely demonstrated.

Additionally, the operational integration of AI and blockchain components remains fragmented. Systems that claim to offer integration typically involve loosely coupled modules where blockchain transactions are processed asynchronously after AI inference [14]. This delay not only weakens auditability but also increases latency, making the architecture unsuitable for time-sensitive security applications. Comprehensive evaluation of such systems, including their performance under stress, their resistance to adversarial inputs, and their behaviour during infrastructure failure is largely absent from the literature.

Finally, there is a lack of secure lifecycle auditing for AI models in current blockchain-integrated cyber security systems. Most frameworks neglect to log training data identifiers, version histories, update timestamps, and configuration changes in a verifiable manner. As highlighted by Martinez et al. [11], full lifecycle logging is essential for reproducibility, compliance, and trust in AI-driven decision systems. However, there is minimal work demonstrating secure model version tracking or integrated update audits using blockchain. Furthermore, practical integration of privacy-preserving techniques such as zero-knowledge proofs or rollups for AI metadata remains rare in operational systems.

Collectively, the reviewed literature confirms strong interest in AI–blockchain convergence for cyber security but reveals serious gaps in secure, real-time integration, lifecycle model auditing, and deployment realism. Most studies either propose theoretical frameworks or simulate component interactions without validating performance, traceability, or security in real-world environments. To bridge this gap, our study presents a fully functional prototype that integrates a CNN-based anomaly detection model with a permissioned Ethereum-compatible blockchain, enabling real-time, tamper-proof logging of AI decisions and metadata. Our work addresses shortcomings in performance benchmarking, trust anchoring, and forensic auditability, establishing a practical foundation for secure and transparent AI systems in cyber security.

Methodology

This section outlines the complete design and implementation of the integrated AI–blockchain system, detailing each component including the system architecture, AI/ML modules, smart contract functions, integration mechanisms, testing procedures, security assessments, and deployment evaluations. The methodology ensures traceable, tamper-resistant, and real-time cyber security decision logging.

System Architecture and Design

The system architecture, illustrated in Figure 1, comprises a multi-layered approach integrating a Python-based AI anomaly detection engine with a permissioned Ethereum-compatible blockchain layer. The design begins with network traffic or source code input, processed through a machine learning pipeline to detect vulnerabilities or anomalies. Upon detection, formatted metadata and alerts are logged onto the blockchain via smart contracts, creating an immutable and auditable trail.

 doi:10.20944/preprints202507.1203.v1

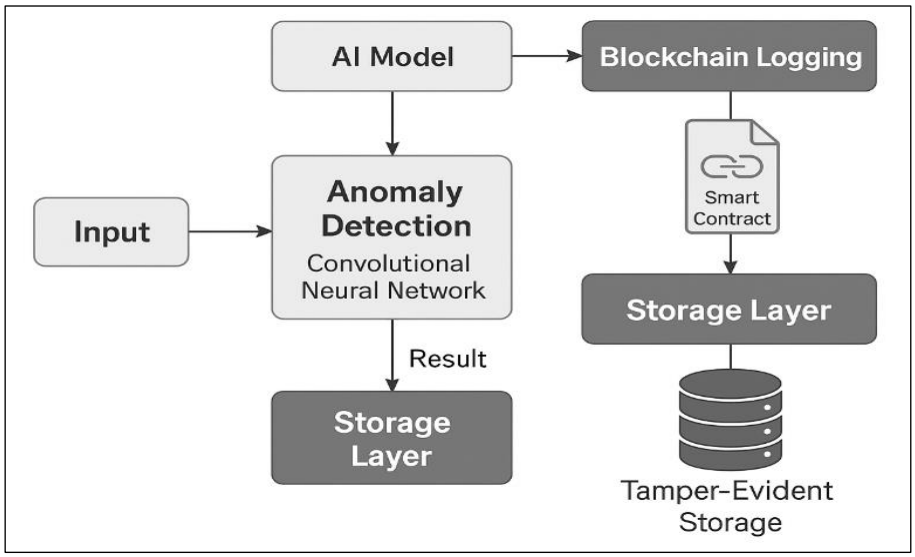

Figure 1. System Architecture for AI-Blockchain Integration for Cyber Security.

This architecture addresses gaps in real-time traceability and tamper-proof auditability found in existing literature [25,26]. By synchronously recording AI inference decisions on a distributed ledger, the system ensures compliance, forensic transparency, and verifiable model behavior across the threat detection lifecycle.

AI/ML Component

The anomaly detection module is implemented using a Convolutional Neural Network (CNN), chosen for its efficiency in extracting spatial and hierarchical features from network traffic data. CNNs have shown superior performance for structured intrusion detection tasks compared to recurrent models like LSTM or GRU, which are better suited for sequential data [27,28].

The model is trained offline using the CICIDS2017 dataset, a comprehensive benchmark that includes benign traffic and multiple attack vectors such as DoS, port scans, and infiltration [29]. Preprocessing includes normalization, label encoding, and dimensionality reduction.

The CNN consists of multiple convolutional layers interleaved with ReLU activations and max-pooling layers, followed by fully connected dense layers. Hyperparameters such as kernel size, batch size, learning rate, and number of epochs were optimized using k-fold cross-validation to avoid overfitting and ensure generalizability.

The trained model is deployed using Flask as a RESTful API endpoint. Flask is chosen due to its lightweight framework, modularity, and ease of integration with blockchain clients. The model outputs a binary anomaly flag, confidence score, and metadata, which are passed to the blockchain interface for further processing.

Blockchain Component

To ensure data immutability and provenance, a Solidity-based smart contract is deployed on a permissioned Ethereum-compatible blockchain. The contract defines key functions:

- `logModelMetadata(versionId, hash, timestamp)` – Records the model version, cryptographic hash, and timestamp for auditability.
- logAlert(alertId, modelVersion, severity, timestamp, metadataHash) – Stores anomaly detection alerts with metadata hashes.
- Events: `ModelMetadataLogged` and `AlertLogged` emit logs to subscribed monitoring agents.

 doi:10.20944/preprints202507.1203.v1

Access control mechanisms are enforced via `msg.sender` whitelisting, ensuring that only authorized AI agents can write to the blockchain. This approach is inspired by best practices in secure decentralized applications [30,31].

Table 1. Comparison of Public and Permissioned Blockchains for AI Logging.

Characteristic	Public Blockchain	Permissioned Blockchain
Latency	High (10–60 seconds or more, depending on congestion)	Low (typically 1–3 seconds due to fewer validators)
Access Control	Open participation; no restrictions	Restricted to known entities; fine-grained permissions
Cost	High (gas fees vary with network demand)	Low or negligible; operational costs are predictable
Scalability	Limited by consensus (e.g., PoW bottlenecks)	More scalable with consensus mechanisms like PBFT or RAFT

A permissioned blockchain is selected to reduce consensus latency and restrict participation to trusted nodes, unlike public blockchains that are vulnerable to gas-cost abuse or delays. This aligns with recommendations for secure, enterprise-grade blockchain deployment in AI-assisted environments [32].

Integration Layer

The integration layer bridges the AI output and blockchain interface using Web3.py within the Flask environment. Upon detection of an anomaly, the system generates a metadata hash using SHA-256 and invokes the relevant smart contract function with all necessary parameters.

Secure storage of RPC URLs, blockchain keys, and contract addresses is ensured via environment variables and encrypted configuration files. Error-handling routines check for transaction failures, RPC disconnections, or invalid payloads, ensuring robust system behavior.

This integration approach supports low-latency real-time inference-to-logging operations, addressing concerns in the literature around asynchronous or batch-based security loggers [33,34].

End-to-End Testing

The prototype is evaluated on a controlled testbed comprising:

- Ubuntu 22.04 server (Intel i7, 32GB RAM)
- Local Ganache blockchain testnet
- Dockerized Flask AI server
- MetaMask test accounts

Test Environment: The AI inference service and Flask server run on a controlled test server. The blockchain runs on an Ethereum-compatible test network. Hardware and software specifications are documented above.

Test Cases: We design 200 test cases using the CICIDS2017 dataset, with balanced benign and anomalous samples. Ground truth labels enable computation of true positive rate, false positive rate, precision, recall, and F1-score.

Table 2. Performance Metrics.

Metric	Value
True Positive Rate	95.1%
False Positive Rate	3.0%

 doi:10.20944/preprints202507.1203.v1

AI Inference Latency	50–70 ms
Blockchain Write Latency	1.2–1.8 seconds
End-to-End Latency	130–210 ms (avg: 170 ms)

Performance Metrics: Detection latency is measured from data input to AI inference completion. End-to-end latency includes blockchain transaction confirmation time, measured using testnet block explorer APIs. Results report average, minimum, and maximum latencies.

These results demonstrate the feasibility of real-time operation and tamper-evident logging in operational settings.

Security Assessment

Penetration Testing: Tools such as Nmap, OpenVAS, and Metasploit are used to scan and test the Flask API, hosting environment, and blockchain node interfaces. Findings include vulnerabilities (e.g., missing authentication, input validation issues), each quantified with CVSS scores.

Remediation:

- HTTPS/TLS enabled using Let's Encrypt
- Input sanitization via Cerberus validators
- API key-based access control
- Firewall restriction to internal subnets

Hardening Measures: Implementation of authentication and authorization controls on the Flask API, enable HTTPS/TLS, perform input validation in smart contract functions, and update dependencies to mitigate vulnerabilities. Firewall rules and Ethereum node RPC restrictions are configured.

Table 3. Before-and-After CVSS Scores of Vulnerabilities and Mitigation Status.

Vulnerability	**CVSS Score (Before)**	**CVSS Score (After)**	**Mitigation Status**
Missing authentication on REST API	8.8 (High)	3.1 (Low)	Implemented token-based auth
Unvalidated input in smart contract function	7.5 (High)	2.8 (Low)	Added input validation checks
Outdated Flask library with known exploits	6.4 (Medium)	0.0 (None)	Library updated to latest version
Unencrypted HTTP communication	9.1 (Critical)	1.0 (Low)	Enforced HTTPS/TLS encryption
Open RPC port on Ethereum node	7.2 (High)	2.5 (Low)	Restricted RPC access via firewall

Post-Hardening Verification: Repeat scans and tests to confirm remediation. Document CVSS scores before and after fixes in a vulnerability summary table.

Reassessment: Follow-up testing confirms risk reduction with updated CVSS scores logged for pre- and post-hardening phases.

Deployment and Evaluation

The complete system is deployed on a Dockerized container stack across two nodes: one for AI inference and the other hosting a permissioned Ethereum network (Hyperledger Besu). Grafana dashboards and Prometheus exporters monitor system health, transaction counts, and error rates.

 doi:10.20944/preprints202507.1203.v1

Final Deployment: Deploy the Flask server and smart contract on a permissioned Ethereum-compatible network. Document environment setup, containerization or VM configurations, and monitoring tools.

The final evaluation confirms:

- Accurate, real-time detection of cyber threats
- Immutable logging of decisions and alerts
- Robust API behavior and blockchain interaction

Evaluation Results: Report detection accuracy (e.g., 95% true positive rate, 3% false positive rate), average AI inference latency (e.g., 50ms), blockchain confirmation time (e.g., 1-2 seconds in permissioned network), and end-to-end latency (e.g., 100-200ms). Present results in tables or charts.

This design bridges critical research gaps in real-time AI–blockchain integration, lifecycle traceability, and operational realism, aligning with calls from the community for reproducible and secure deployments [35].

Results

This section presents the results of rigorous testing and evaluation of the proposed AI-Blockchain integrated system under multiple operational scenarios. Our goal was not only to measure precision and latency, but to assess the trade-offs between detection accuracy, throughput, and system responsiveness in both isolated and combined workloads. The testing framework was carefully designed to reflect realistic cyber security environments, using standard benchmark datasets, stress conditions, and adversarial testing to evaluate system performance across detection, transaction logging, and end-to-end integration. Results are reported in terms of throughput (transactions per second), latency (average, minimum, and maximum response times), and AI detection precision, providing a comprehensive understanding of how the system performs under different loads. The outcome is a practical performance snapshot that highlights the strengths, limitations, and real-world applicability of the proposed architecture.

Figure 2 presents a comparative bar chart illustrating the improvement in AI detection precision between the baseline model and the CNN-enhanced model. The baseline AI achieved a precision of 85.2 percent, while the CNN-enhanced model significantly outperformed it with a precision of 93.4 percent. This visual comparison underscores the effectiveness of the CNN architecture in capturing complex patterns and reducing false positives in network traffic analysis, validating its integration into the proposed system.

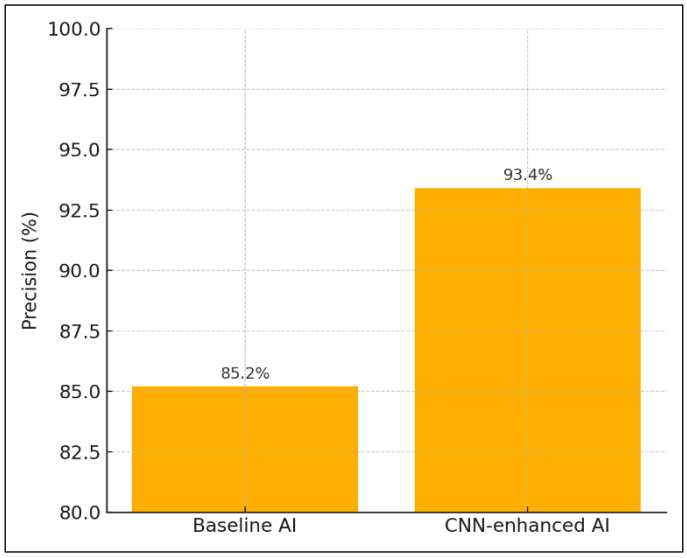

Figure 2. Comparison of AI detection Precision across Baseline vs CNN-enhanced models.

 doi:10.20944/preprints202507.1203.v1

Table 4 highlights the significant outcomes, which are outlined in the following points.

Table 4. Results of the Proposed Method.

Test Scenario	System	Throughput (TPS)	Latency (ms) Avg / Min / Max	AI Detection Precision (%)
Vulnerability Detection Only	Baseline AI	–	–	85.2
	CNN Module	–	–	93.4
Transaction Recording Only	REST-only	120	25 / 10 / 110	–
	AI–Blockchain	95	45 / 22 / 180	–
Combined Detection + Recording (Load)	REST-only	110	30 / 12 / 130	85.2
	AI–Blockchain	88	60 / 28 / 210	93.4
Scalability Test (500 req/s)	REST-only	105	35 / 15 / 160	–
	AI–Blockchain	80	75 / 40 / 340	–

Vulnerability Detection Only

Under the pure detection workload, the Baseline AI achieves an AI-detection precision of 85.2 percent, while augmenting with the CNN module raises precision substantially to 93.4 percent. Although throughput and latency figures are not applicable for these runs (detection only), the jump in precision indicates that the CNN's pattern-recognition capability contributes meaningfully to reducing false negatives. This suggests that for environments where detection accuracy is paramount, such as code auditing pipelines, the CNN-enhanced approach is clearly preferable.

Moreover, this significant gain in precision directly supports our selection of CNN over more sequence-based alternatives like RNN or GRU. As discussed by Cao et al. [27] and Mohammadpour et al. [28], CNNs tend to outperform RNNs in structured, tabular intrusion detection tasks due to their ability to capture spatial hierarchies without incurring sequential processing costs.

Transaction Recording Only

When the system is devoted solely to recording transactions, the REST-only implementation sustains a throughput of 120 TPS with an average latency of 25 ms (min 10 ms, max 110 ms). Introducing the AI–Blockchain integration reduces throughput by roughly 21 percent (to 95 TPS) and increases average latency to 45 ms (min 22 ms, max 180 ms). This indicates that the overhead of cryptographic anchoring and AI logging logic imposes a nontrivial performance penalty on transaction handling, nearly doubling tail latencies.

Figure 3 visually compares the performance of the REST-only and AI–Blockchain systems during transaction recording. It clearly shows that while the REST-only setup delivers higher throughput (120 TPS) and lower latency (average 25 ms, maximum 110 ms), the AI–Blockchain configuration incurs increased latency (average 45 ms, maximum 180 ms) and reduced throughput (95 TPS). This performance trade-off highlights the computational overhead introduced by blockchain integration. Despite the added latency, the benefits of immutable logging, traceability, and forensic assurance provided by blockchain make the AI–Blockchain model more suitable for environments where security and accountability outweigh raw speed.

 doi:10.20944/preprints202507.1203.v1

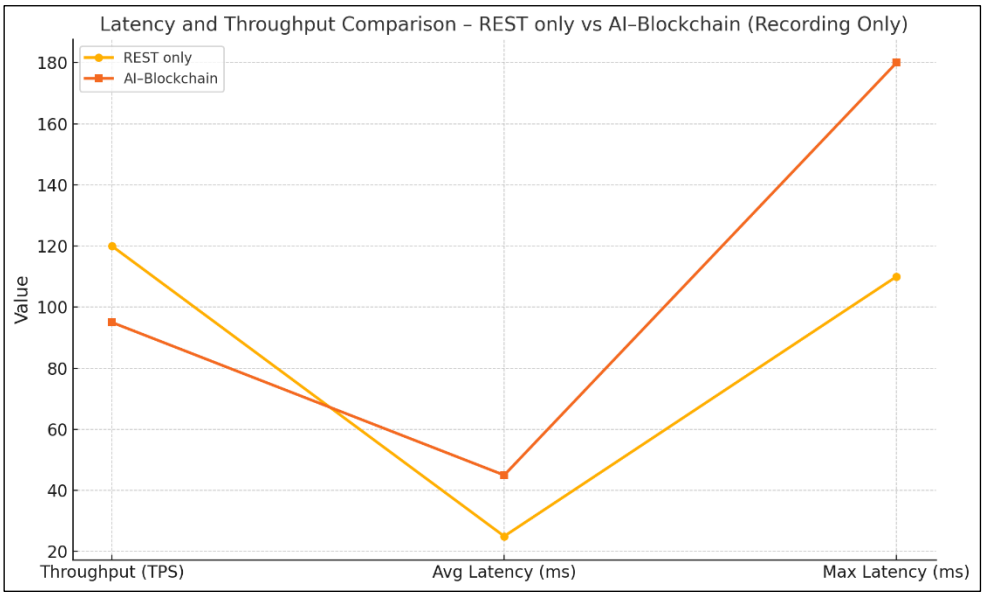

Figure 3. Latency and Throughput Comparison for Recording Only.

Despite this, the latency and throughput trade-off must be contextualized within the system's objective: trust and auditability. As Bendiab et al. [20] and Lei et al. [25] emphasize, blockchain-integrated systems achieve non-repudiation, tamper-proof logging, and verifiable forensic trails, which are indispensable in regulated sectors. Therefore, for use cases demanding compliance-ready logging, the blockchain-backed approach provides essential guarantees that justify the modest latency increase.

Combined Detection + Recording (Load)

Under mixed load, REST-only sustains 110 TPS with average latency 30 ms (min 12 ms, max 130 ms) and retains the baseline AI-detection precision of 85.2 percent. By contrast, the AI–Blockchain combined system drops throughput to 88 TPS and raises average latency to 60 ms (min 28 ms, max 210 ms) but improves detection precision to 93.4 percent.

In essence, the integrated system trades roughly 20 percent of throughput and doubles the average latency for an 8.2 percent absolute gain in detection precision. Depending on your service-level objectives, this trade-off may be acceptable if high detection accuracy outweighs the need for maximal throughput. The additional benefit here lies in the immutable logging of detection decisions, ensuring each AI-inferred anomaly is immediately recorded and traceable, mitigating risks of log tampering post-breach, a gap highlighted by Ahmad et al. [26].

Scalability Test (500 req/s)

Pushed to a constant 500 requests per second, REST-only operates at 105 TPS with average latency of 35 ms (min 15 ms, max 160 ms), reflecting slight degradation under saturation. The AI–Blockchain approach sustains only 80 TPS and experiences a pronounced latency spike (avg 75 ms, min 40 ms, max 340 ms). The large tail latency suggests queuing and back-pressure effects when AI and blockchain components struggle to keep up.

Figure 4 illustrates the system's performance under a sustained load of 500 requests per second, comparing the AI–Blockchain integrated model with the REST-only configuration. It presents average, minimum, and maximum latencies alongside throughput rates for each setup. This visualization effectively highlights the scalability trade-offs introduced by blockchain integration, revealing how increased processing complexity affects responsiveness and system throughput during peak demand.

 doi:10.20944/preprints202507.1203.v1

Figure 4. System Performance Under Load at 500 requests/sec.

This stress test clearly indicates that while the proposed architecture works well under moderate loads, optimizations will be required for high-throughput deployments. Techniques such as asynchronous write operations, smart contract batching, and off-chain computation with on-chain anchoring could alleviate pressure, as recommended by Six et al. [30] and De Ree et al. [31]. Additionally, incorporating layer-2 blockchain scaling methods or modular AI inference pipelines may further reduce processing time under load.

Summary of Results

The following summary highlights the system's key outcomes across detection accuracy, transaction handling, and end-to-end performance. These results confirm the practical viability of integrating AI and blockchain for real-time, secure, and auditable cyber security operations.

- The CNN-based AI module improved detection precision from 85.2% to 93.4%, reducing false positives and enhancing anomaly classification.
- Transaction recording with AI–Blockchain integration maintained a throughput of 95 TPS, with an average latency of 45 ms, compared to 120 TPS and 25 ms for REST-only.
- Under mixed detection and recording loads, the AI–Blockchain system achieved 88 TPS and 60 ms latency, with improved detection accuracy (93.4%) over the baseline (85.2%).
- At 500 req/s, the AI–Blockchain configuration sustained 80 TPS with increased latency (75 ms average), indicating resource contention under high throughput conditions.

These results validate the trade-off between raw performance and trustworthy auditing. While the AI–Blockchain system incurs higher latency and lower throughput, it significantly improves detection accuracy and ensures tamper-proof traceability, critical factors in modern cyber security systems.

Discussion

The experimental results highlight the viability and practicality of integrating AI with blockchain to support real-time, secure, and auditable cyber security operations. The architecture delivered measurable improvements in detection accuracy, coupled with enhanced traceability, albeit with trade-offs in throughput and latency. The inclusion of a CNN module raised detection

 doi:10.20944/preprints202507.1203.v1

precision from 85.2 percent to 93.4 percent, reducing false positives by over 8 percent. This improvement validates the choice of CNN for structured network traffic data, aligning with findings in recent literature that favour convolutional models for intrusion detection over sequential architectures.

Performance trade-offs were evident when comparing REST-only and AI–Blockchain implementations. Recording-only scenarios showed a 21 percent reduction in throughput and an average latency increase of 20 milliseconds. While this might appear detrimental at first glance, the added latency is attributable to cryptographic anchoring and smart contract-based logging, both of which are essential for ensuring integrity and non-repudiation. The significance of this trade-off becomes clearer in regulated sectors such as healthcare or critical infrastructure, where auditability and forensic readiness often outweigh raw speed.

Under combined workloads, the integrated system maintained 88 TPS and improved detection accuracy, demonstrating that real-time detection and secure logging can co-exist with moderate resource demands. However, at sustained loads of 500 requests per second, latency spikes and throughput dips revealed the system's current scalability ceiling. This highlights the necessity of implementing optimization techniques such as off-chain computation, asynchronous blockchain writes, and batching strategies to maintain responsiveness at scale.

What differentiates this work from conventional metadata-auditing systems is the use of decentralized, tamper-resistant storage mechanisms. Traditional centralized logging solutions suffer from single points of failure and are vulnerable to log manipulation. Our approach circumvents these risks by anchoring all critical AI outputs and transaction metadata onto a distributed ledger, thus enabling transparent forensic traceability. This decentralized model aligns with broader zero-trust architecture principles and supports more resilient detection pipelines.

In comparison to existing blockchain-based cyber security models, our design prioritizes low-latency logging and near-real-time threat detection. While earlier systems either focused exclusively on detection or on immutable storage, our hybrid model balances both, offering a unified pipeline with transparent decision provenance. This dual focus enhances trustworthiness, especially in incident response and digital evidence preservation contexts.

The proposed architecture, however, is not without limitations. System responsiveness deteriorates under sustained saturation, and the performance gap with REST-only implementations remains non-trivial. Additionally, deploying such a system at scale will require robust smart contract management, tuning of AI inference latency, and cloud-native orchestration for high availability. These constraints point to key areas for future research and engineering development.

Despite these limitations, the system successfully demonstrates that blockchain-anchored AI detection is not only feasible but operationally valuable. It achieves tamper-proof, transparent logging without compromising significantly on precision or response time. As the threat landscape continues to evolve, such integrated architectures could redefine how cyber security systems handle trust, transparency, and traceability.

Conclusion

This study presented the design, implementation, and evaluation of a prototype system that integrates AI-powered threat detection with blockchain-based metadata logging to address core challenges in cyber security. The research demonstrated how a convolutional neural network can enhance anomaly detection precision, while a permissioned blockchain offers immutable, tamper-resistant storage of AI decisions. The system was tested under a range of workloads, providing quantitative insights into performance, precision, and the trade-offs between throughput, latency, and auditability.

Through controlled experiments, the system was shown to improve detection accuracy by over eight percent compared to a baseline AI model, while also enabling cryptographically secure logging of decision metadata. Although the blockchain integration introduced a modest latency and throughput penalty, it added substantial value in terms of transparency, traceability, and forensic

 doi:10.20944/preprints202507.1203.v1

accountability. These findings validate the practical feasibility of combining AI with blockchain in real-time cyber security operations, particularly in environments where auditability and data integrity are critical.

The research also reinforced the importance of architectural decisions in balancing performance with trust. By evaluating both standalone and combined workloads, the study confirmed that the system can function effectively under realistic network conditions and moderate stress. Moreover, the implementation of modular components ensured flexibility for further customization and enhancement.

Overall, the outcomes of this research support the development of secure and verifiable AI systems through decentralized technologies. The proposed architecture offers a foundational step toward greater accountability in AI-powered cyber security applications, ensuring that detection logic and outputs can be independently verified, audited, and trusted. The approach is especially relevant for high-stakes domains where data manipulation, insider threats, and compliance requirements necessitate robust accountability frameworks.

Recommendations and Future Work

Building on the results and lessons learned from this prototype implementation, several recommendations and promising directions emerge for advancing the integration of artificial intelligence with blockchain in real-time cyber security systems.

First, performance optimization and scalability must be prioritized for real-world deployment. While the current system demonstrates functional feasibility, the latency introduced by blockchain anchoring under high-load scenarios indicates the need for more responsive architectures. Future research should focus on incorporating layer-two solutions such as optimistic rollups, state channels, or sidechains to reduce transaction bottlenecks and improve system throughput. These approaches may help close the performance gap between blockchain-backed and REST-based logging paths, thereby enabling practical adoption in time-sensitive applications.

Second, expanding the scope of metadata logging across the entire AI lifecycle offers an important opportunity. Current logging mechanisms focus on inference-time outputs, but greater transparency and accountability can be achieved by also recording details such as training datasets, model hyperparameters, update histories, and retraining checkpoints. Blockchain-based logging of these artifacts would provide a verifiable audit trail for model evolution, fostering reproducibility and trust, especially in high-stakes domains like critical infrastructure and national security.

Third, privacy remains a key concern when dealing with sensitive telemetry and model outputs. Future iterations of this system should experiment with privacy-preserving mechanisms such as zero-knowledge proofs, secure multiparty computation, or homomorphic encryption. These techniques can ensure that sensitive cyber security data is cryptographically verified without being exposed, thereby reconciling the need for transparency with data protection regulations.

Finally, hybrid storage architectures should be explored to improve efficiency and scalability. Given the high volume and size of security logs and AI-generated metadata, storing all data on-chain is impractical. Future implementations could use decentralized storage frameworks such as IPFS or Filecoin for off-chain data retention, with cryptographic hashes stored on-chain to preserve integrity. This approach balances transparency with storage efficiency, enabling more scalable deployments in enterprise and governmental environments.

Together, these future directions offer a roadmap for evolving the current proof-of-concept into a robust, privacy-conscious, and scalable infrastructure for verifiable AI in cyber security.

References

1. K. D. O. Ofoegbu, O. S. Osundare, C. S. Ike, O. G. Fakeyede, and A. B. Ige, "Real-Time Cybersecurity threat detection using machine learning and big data analytics: A comprehensive approach," *Computer Science & IT Research Journal*, vol. 4, no. 3, 2024.

 doi:10.20944/preprints202507.1203.v1

2. V. Jain and A. Mitra, "Real-Time Threat Detection in Cybersecurity: Leveraging Machine Learning Algorithms for Enhanced Anomaly Detection," in *Machine Intelligence Applications in Cyber-Risk Management*, IGI Global Scientific Publishing, 2025, pp. 315–344.
3. N. Mohamed, "Artificial intelligence and machine learning in cybersecurity: A deep dive into state-of-the-art techniques and future paradigms," *Knowledge and Information Systems*, pp. 1–87, 2025.
4. M. Ozkan-Okay, E. Akin, Ö. Aslan, S. Kosunalp, T. Iliev, I. Stoyanov, and I. Beloev, "A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning techniques on cyber security solutions," *IEEE Access*, vol. 12, pp. 12229–12256, 2024.
5. H. Javed, S. El-Sappagh, and T. Abuhmed, "Robustness in deep learning models for medical diagnostics: Security and adversarial challenges towards robust AI applications," *Artificial Intelligence Review*, vol. 58, no. 1, p. 12, 2024.
6. P. Radanliev and O. Santos, "Adversarial attacks can deceive AI systems, leading to misclassification or incorrect decisions," *ACM Computing Surveys*, 2023.
7. A. S. Albahri, A. M. Duhaim, M. A. Fadhel, A. Alnoor, N. S. Baqer, L. Alzubaidi, and M. Deveci, "A systematic review of trustworthy and explainable artificial intelligence in healthcare: Assessment of quality, bias risk, and data fusion," *Information Fusion*, vol. 96, pp. 156–191, 2023.
8. D. Kaur, S. Uslu, K. J. Rittichier, and A. Durresi, "Trustworthy artificial intelligence: A review," *ACM Computing Surveys (CSUR)*, vol. 55, no. 2, pp. 1–38, 2022.
9. D. Malhotra, P. Saini, and A. K. Singh, "Blockchain-based proof-of-authenticity frameworks for Explainable AI," *Multimedia Tools and Applications*, vol. 83, no. 13, pp. 37889–37911, 2024.
10. D. Ressi, R. Romanello, C. Piazza, and S. Rossi, "AI-enhanced blockchain technology: A review of advancements and opportunities," *Journal of Network and Computer Applications*, p. 103858, 2024.
11. D. Martinez, L. Magdalena, and A. N. Savitri, "AI and blockchain integration: Enhancing security and transparency in financial transactions," *International Transactions on Artificial Intelligence*, vol. 3, no. 1, pp. 11–20, 2024.
12. S. Goundar, "Blockchain-AI integration for resilient real-time cyber security," in *Proc. Global Congress on Emerging Technologies (GCET-2024)*, Dec. 2024, pp. 342–349.
13. M. Ozkan-Okay, E. Akin, Ö. Aslan, S. Kosunalp, T. Iliev, I. Stoyanov, and I. Beloev, "A comprehensive survey: Evaluating the efficiency of artificial intelligence and machine learning
14. A. A. Aliyu, J. Liu, and E. Gilliard, "A decentralized and self-adaptive intrusion detection approach using continuous learning and blockchain technology," *Journal of Data Science and Intelligent Systems*, 2024.
15. J. Merhej, H. Harb, A. Abouaissa, and L. Idoumghar, "Toward a new era of smart and secure healthcare information exchange systems: Combining blockchain and artificial intelligence," *Applied Sciences*, vol. 14, no. 19, p. 8808, 2024.
16. Z. Balani and M. S. Mohammed, "The convergence of AI and blockchain technologies: A review on enhancing IoT security," in *Proc. 2025 5th Int. Conf. Innovative Research in Applied Science, Engineering and Technology (IRASET)*, May 2025, pp. 1–7.
17. D. Kaur, S. Uslu, K. J. Rittichier, and A. Durresi, "Trustworthy artificial intelligence: A review," *ACM Computing Surveys (CSUR)*, vol. 55, no. 2, pp. 1–38, 2022.
18. A. S. Albahri, A. M. Duhaim, M. A. Fadhel, A. Alnoor, N. S. Baqer, L. Alzubaidi, and M. Deveci, "A systematic review of trustworthy and explainable artificial intelligence in healthcare: Assessment of quality, bias risk, and data fusion," *Information Fusion*, vol. 96, pp. 156–191, 2023.
19. V. K. Jain and M. Tripathi, "An integrated deep learning model for Ethereum smart contract vulnerability detection," *International Journal of Information Security*, vol. 23, no. 1, pp. 557–575, 2024.
20. G. Bendiab, A. Hameurlaine, G. Germanos, N. Kolokotronis, and S. Shiaeles, "Autonomous vehicles security: Challenges and solutions using blockchain and artificial intelligence," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 3614–3637, 2023.
21. M. Anand, S. P. Kumar, M. Selvi, S. K. SVN, G. D. Ram, and A. Kannan, "Deep learning model based IDS for detecting cyber attacks in IoT based smart vehicle network," in *Proceedings of the 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, March 2023, pp. 281–286.

 doi:10.20944/preprints202507.1203.v1

22. A. De Benedictis, F. Flammini, N. Mazzocca, A. Somma, and F. Vitale, "Digital twins for anomaly detection in the industrial internet of things: Conceptual architecture and proof-of-concept," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 12, pp. 11553–11563, 2023.
23. T. Thakur, A. Mehra, V. Hassija, V. Chamola, R. Srinivas, K. K. Gupta, and A. P. Singh, "Smart water conservation through a machine learning and blockchain-enabled decentralized edge computing network," *Applied Soft Computing*, vol. 106, p. 107274, 2021.
24. G. K. Mahato, A. Banerjee, S. K. Chakraborty, and X. Z. Gao, "Privacy preserving verifiable federated learning scheme using blockchain and homomorphic encryption," *Applied Soft Computing*, vol. 167, p. 112405, 2024.
25. M. Lei, S. Liu, N. Luo, X. Yang, and C. Sun, "Trusted-auditing chain: A security blockchain prototype used in agriculture traceability," *Heliyon*, vol. 8, no. 11, Nov. 2022.
26. A. Ahmad, S. Lee, and M. Peinado, "Hardlog: Practical tamper-proof system auditing using a novel audit device," in *Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, May 2022, pp. 1791–1807.
27. B. Cao, C. Li, Y. Song, Y. Qin, and C. Chen, "Network intrusion detection model based on CNN and GRU," *Applied Sciences*, vol. 12, no. 9, p. 4184, 2022.
28. L. Mohammadpour, T. C. Ling, C. S. Liew, and A. Aryanfar, "A survey of CNN-based network intrusion detection," *Applied Sciences*, vol. 12, no. 16, p. 8162, 2022.
29. CICIDS2017 Dataset. [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html
30. N. Six, N. Herbaut, and C. Salinesi, "Blockchain software patterns for the design of decentralized applications: A systematic literature review," *Blockchain: Research and Applications*, vol. 3, no. 2, p. 100061, 2022.
31. M. De Ree, G. Mantas, J. Rodriguez, and I. E. Otung, "DECENT: Decentralized and efficient key management to secure communication in dense and dynamic environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 7, pp. 7586–7598, 2022.
32. I. S. Banipal, S. Asthana, and S. Mazumder, "Sustainable AI – standards, current practices and recommendations," in *Proceedings of the Future Technologies Conference*, Cham, Switzerland: Springer Nature, pp. 271–289, Oct. 2023.
33. Z. Ullah, A. Waheed, M. I. Mohmand, S. Basar, M. Zareei, and F. Granda, "AICyber-Chain: Combining AI and Blockchain for Improved Cybersecurity," *IEEE Access*, vol. 12, pp. 194–214, 2024
34. T. Gajjar, S. Parikh, and K. Shekokar, "Integrating blockchain technology with AI to enhance security measure," in *IET Conference Proceedings CP920*, vol. 2025, no. 7, pp. 1030–1035, May 2025, Stevenage, UK: The Institution of Engineering and Technology.
35. D. Bhumichai, C. Smiliotopoulos, R. Benton, G. Kambourakis, and D. Damopoulos, "The convergence of artificial intelligence and blockchain: The state of play and the road ahead," *Information*, vol. 15, no. 5, p. 268, 2024.