Provenance Blockchain with Predictive Auditing Framework for Mitigating Cloud Manufacturing Risks in Industry 4.0

By

Mifta Ahmed Umer

A dissertation submitted to the School of Information Technology and Engineering-SiTE

College of Technology and Built Environment-CTBE

In Paritial Fulfillment of the Requirements for the Degree of Doctor of Philosophy (Ph.D)

Under the Supervision of Prof. Dr. Luis Borges Gouveia and Dr. Elefelious Getachew Belay

June, 2025

Dissertation Report Approval

Prof. Luis Borges Gouveia	Assinado por: **LUÍS MANUEL BORGES GOUVEIA** Num. de Identificação: 07336056 Data: 2025.06.20 12:49:38+01'00'
Name	Signature
Dr. Elefelious Getachew	
Name	Signature

Examiners

Dr. Surafel	
Name	Signature
Dr. Birhanu	
Name	Signature

Abstract

Cloud manufacturing is an evolving concept that enables various manufacturers to connect and address shared demand streams regardless of their geographical location. Although this transformation facilitates operational flexibility and resource optimization, it concurrently introduces critical challenges related to continuous visibility, traceability, and proactive security management within Industrial Internet of Things (IIoT)-enabled cloud manufacturing environments. Notably, the absence of real-time insights into device states and operational behaviors increases susceptibility to unauthorized access, latent security breaches, and operational disruptions, whereas existing blockchMLn-based solutions predominantly emphasize initial authentication and transactional integrity but lack mechanisms for ongoing device verification and continuous provenance tracking. Simultaneously, artificial intelligence (ML)-driven predictive auditing techniques have evolved in isolation, without harnessing the immutability, accountability, and policy enforcement capabilities afforded by blockchMLn technology. This fragmentation results in limited traceability and weakened system integrity, particularly in dynamic IIoT ecosystems, where timely data-driven decision making is imperative. This study MLms to address these gaps through three primary objectives: (i) optimize blockchMLn architectures to support continuous monitoring, traceability, and visibility in IIoT environments; (ii) develop and integrate ML-based predictive auditing mechanisms with blockchMLn to proactively detect and mitigate security risks in IIoT-based cloud manufacturing; and (iii) evaluate the effectiveness of the integrated blockchMLn and predictive auditing framework in addressing security, traceability, and real-time visibility challenges while mMLntMLning operational continuity. Adopting a Design Science Research Methodology (DSRM), this study develops and rigorously evaluates an integrated framework that combines dynamic blockchMLn-based provenance logging with ML-driven anomaly detection. The experi-

mental evaluation was conducted using a scenario-based experimental setup in a cloud simulated multizone warehouse environment involving IIoT-enabled forklifts that operated under three behavioral scenarios: fully compliant, partially compliant, and rogue. Key evaluation metrics included validation accuracy 94%, prediction precision (up to 99.7%, F1 score 90%, traceability rate (ranging from 82% to 85%, average system latency (3.95 seconds), transaction rejection rate (100% for rogue inputs), and operational uptime (100% resilience with no downtime). The results substantiate the ability of the framework to provide real-time responsiveness, robust security, and continuous traceability while mMLntMLning operational continuity, even under adversarial or non-compliant conditions. This study contributes to the body of knowledge by bridging the gap between blockchMLn technology and ML in IIoT-enabled cloud-manufacturing security. These findings have practical implications for the secure deployment of IIoT technologies across smart manufacturing ecosystems.

Keywords: Cloud Manufacturing, IIoT, Industry 4.0, Risks, Provenance, Predictive Auditing. Integrated Visualization.

Acknowledgments

This dissertation would not have been possible without the support, guidance, and encouragement of many individuals, to whom I am sincerely grateful.

First and foremost, I would like to express my heartfelt gratitude to my supervisors, Prof. Luis Borges Gouveia and Dr. Elefelious Getachew for their invaluable mentorship, insightful guidance, and continuous support throughout every stage of this research. Their academic rigour, constructive criticism, and unwavering commitment were instrumental in shaping this work.

I would also like to thank the School Graduating Committee, for their valuable input and critical evaluations that helped refine and strengthen this work. Their perspective has complemented and strengthened the research process in meaningful ways.

Finally, I extend my heartfelt thanks to my family for their patience, understanding, and unwavering support throughout this academic journey.

Contents

List of Figures

List of Tables

Chapter I

Introduction

1.1 Background and Context

The Fourth Industrial Revolution, referred to as Industry 4.0, is defined as the integration of cyber-physical systems (CPS) with the Industrial Internet of Things (IIoT), fundamentally reshaping the manufacturing industry. This paradigm emphasizes automation, real-time data analytics, and interconnected systems for optimizing the manufacturing processes. Within this transformative landscape, Cloud Manufacturing (C-MFG) has emerged as a revolutionary concept that enables geographically dispersed manufacturers to collaborate through networks that address common demand streams[1]–[3]. Currently, the infrastructure for information technology, systems, and communication (ITSC) is seamlessly integrated into the operational framework of manufacturing companies under a new paradigm known as Industry 4.0 [2]. Manufacturing organizations have adopted modern technologies such as the Industrial Internet of Things (IIoT), autonomous and adaptive robotics and machines, big data analytics, vertical and horizontal integration, augmented reality, and additive manufacturing to respond to the dynamic nature of markets, consumer demands, and business environments [3]. Production, logistics, and supply chain engineering systems are interconnected and Internet-enabled through the Industrial Internet of Things (IIoT) [3]. Traditionally, networks of these engineering systems have been established using programmable logic controllers and supervisory/distributed control systems [4]. These networks have created isolated connectivity; however, the advent

of IIoT has facilitated comprehensive connectivity among all engineering systems utilized by manufacturing organizations. Consequently, industrial systems capable of sensing process parameters can collaborate cognitively and transmit their data and reports to big data servers in cloud-computing environments [5]. A simplified representation of the C-MFG layer is shown in Fig. 1. Software-based manufacturing controllers are deployed in cloud computing over big data servers to make decisions based on the collective analysis of sensory data and send back commands for actuation, tuning, or enabling/disabling process-governing parameters to field-level engineering systems [6]. Cloud manufacturing is a framework for software-based manufacturing systems and controllers deployed in cloud computing [5]–[8]. In the Industry 4.0 framework, engineering systems that utilize IIoT create a digitalized perception layer, which is a key advancement of Industry 3.0. Above this layer, systems involving big data and artificial intelligence are implemented to interpret the data gathered by the perception layer and to facilitate decisions related to actuation, activation, and governance. Cloud manufacturing can be used to create digitalized manufacturing consortiums of shared manufacturing and logistics resources participating in shared demand fulfilment [5]–[8].

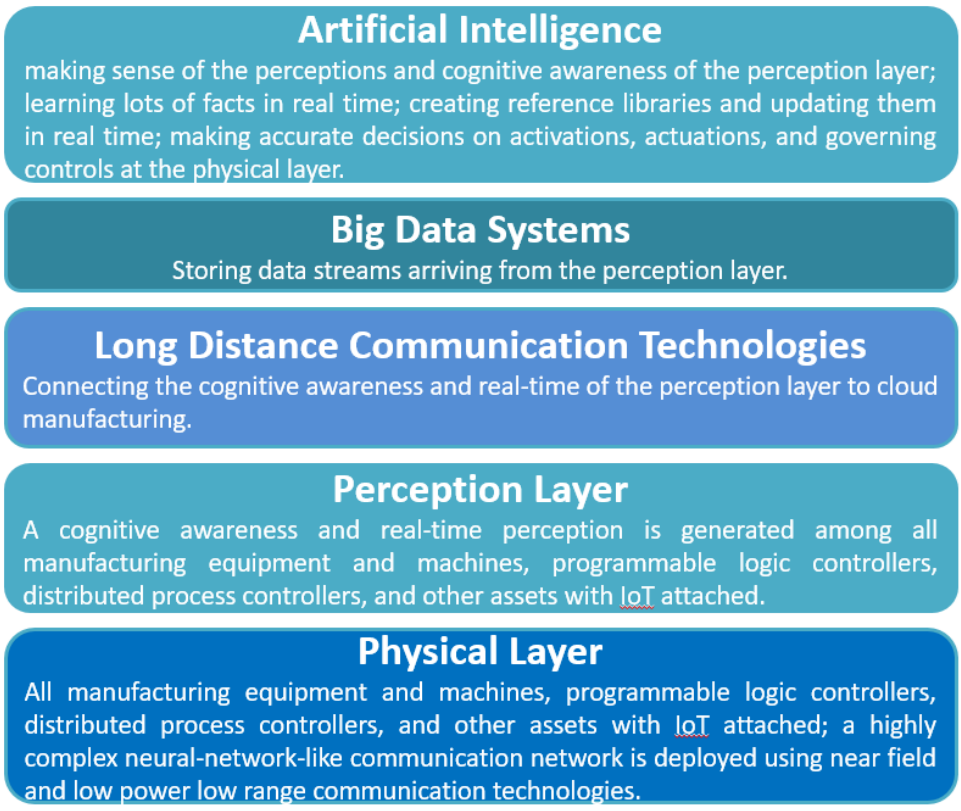


Figure 1: Cloud manufacturing layers (formulated from the analyses in [6],[7], [8])

As illustrated in Figure 1, The AI layer collects data streams from field sensors embedded in industrial and logistics systems through the IPv6 communication protocol, forming what is known as a CPS or IIoT. At the core, it is essential that the industrial sensors powered by the IIoT are precisely configured and calibrated. Nonetheless, at a more advanced level, issues arise concerning the accountability, traceability, fairness, and transparency of the data gathered by AI systems. This necessitates the detection, tracking, and capturing of all data manipulation processes within IIoT devices to achieve a thorough understanding of provenance data [9]–[11]. Conventional data visualization frameworks, such as database queries, are unreliable because of their reliance on single-point capture. Reliable data visualization is possible when multiple sources are utilized. Blockchain technology has been developed in this context. Data collected from field events can be encrypted and duplicated across multiple peer computers to ensure immutability, tamper resistance, and data security. The data remain reliable and maintain strong privacy within a closed semi-private network of peer organizations that share secure and encrypted network segments. The next subsection discusses the blockchain.

1.1.1 Blockchain Technology for Mitigating Data Protection Risks in Cloud Manufacturing (C-MFGs)

Blockchain technology for cloud manufacturing and logistics/supply chain applications is an extended conceptualization of the Ethereum technology used in the world of crypto currencies [10]–[13]. Blockchain refers to an encrypted network established among participating organizations that authorizes their personnel (called peers) to sign agreements and conduct transactions as per them. The technology for digital contracting and capturing transactions related to it is called the "smart contract ledger". The participating organizations make their industrial engineering assets (production machines, production and support robots, equipment, trolleys, cranes, vehicles, etc.) visible to the smart ledger through the IIoT integration. This integration makes them "IIoT devices." These assets can be tied to smart contracts by defining their respective roles. The execution of the respective roles by assets is tracked in the form of transactions, such that payments can be released using them as evidence. Each asset is allowed to transmit relevant data to the smart ledger to track the execution of the roles of these industrial engineering assets. For example, if a transport has transported goods from Point A to Point B, the smart ledger

can track them as execution evidence, based on the data transmitted from the truck and consignments. Data transmission can be related to every process event executed by industrial engineering assets relevant to the execution of smart contracts.

Blockchain technology can mitigate the data protection risks of Industry 4.0, manufacturing settings [11]–[13]. This is because all smart contracts are encrypted, digitally signed, and constitute cryptographic hash functions that ensure the integrity of the transactional blocks. The blocks are replicated to ledgers held in all peer machines authorized by collaborating organizations. A blockchain anchor peer may be a customer hiring multiple cloud manufacturers through cloud manufacturing applications. Several studies have discussed blockchain and its smart-ledger applications [10]–[14]. However, very little emphasis has been placed on ensuring the reliability and trustworthiness of sources that supply information to the blockchain. These data sources are logistics engineering assets operating outside the blockchain. The data sources were declared by the participating manufacturing organizations. Once they are allowed to send transactional data and are protected in the blockchain using key pairs (encryption and decryption keys, symmetric or asymmetric, depending on the blockchain design), they are deemed trusted and their transactions are protected by the blockchain. Blockchain transactions are immutable. Blockchains are designed to protect transactional information from trusted sources; they will end up doing the same for malicious sources if trusted mistakenly, as well as being planted by malicious actors, such as illegal traders, proliferators, and masquerading asset owners. For example, if the blockchain has protected the information supplied by a "counterfeit production agency," it will be very difficult for the "genuine production agency" to seek protection on the same blockchain.

One of the solutions to this challenge that has evolved recently is provenance data capturing and its use as a mechanism for authenticating information sources (the IIoTs in the context of this research). Blockchain provenance data capture and approval can be used to enhance the IIoT security in cloud manufacturing. One may imagine a blockchain network in which only authorized devices are allowed to run processes through registration in smart contracts and monitoring and control using smart ledger updating. Such systems have been the focus of many recent studies conducted to find solutions to IIoT threats and vulnerabilities identified in several other studies [14]–[19]. These designs

should incorporate a trust validation system that employs the exchange of asymmetric cryptographic keys. By setting up the necessary data structures from IIoT devices, a smart contract for provenance can be developed to confirm their genuine (uncompromised) status [10], [17], [20]. A smart contract was established to oversee provenance and create essential data structures for IIoT devices to verify their authenticity and ensure that they remain uncompromised. The blockchain framework assigns crucial roles to data owners, data users, provenance auditors, and provenance validators, to uphold operational transparency and accountability. Figure 2 presents a simplified diagram that integrates the architectures suggested in [68] and [69]. The design of block structures,

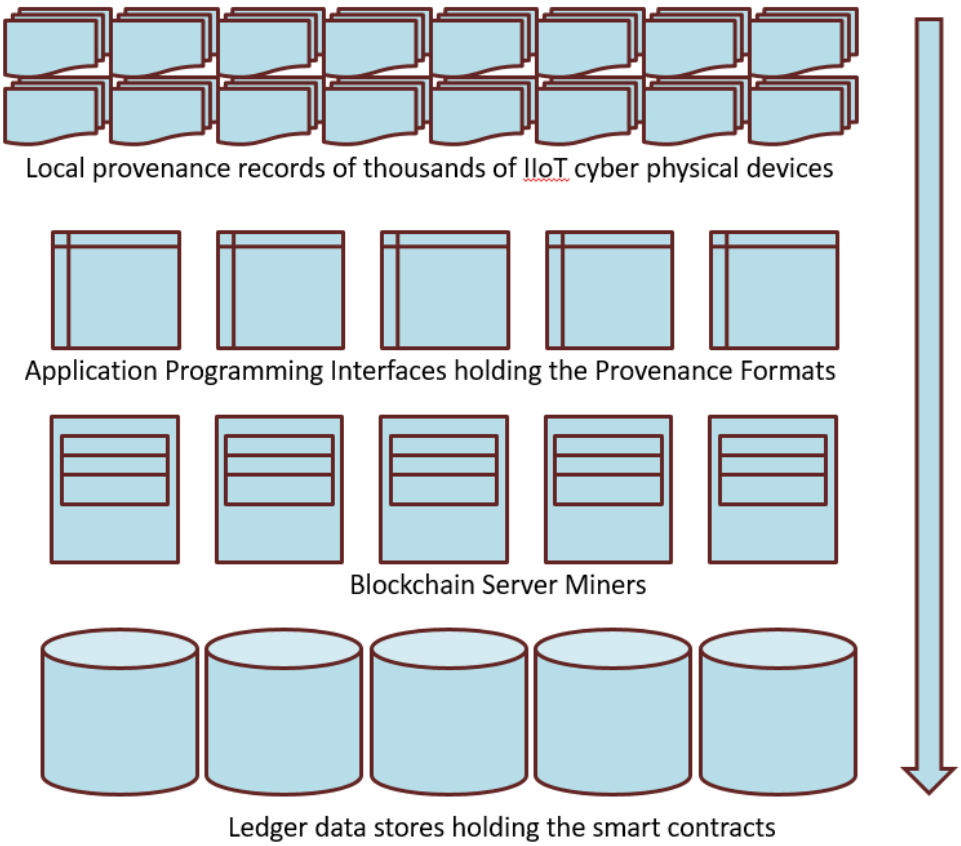


Figure 2: A simplified infrastructure design of provenance blockchains (based on [68],[69])

data modeling, and integrated strategic consensus modeling are essential for these services [19]. A block structure may consist of a block hash, a transaction digest, and a state digest. To confirm the outcomes against the values recorded in a block's digests using a block hash, it is necessary to execute both the transaction and state digests within that block. When the results match, the block is considered unaltered and can be incorporated into the ledger linked to smart contracts that require provenance data.

Ideally, block records are transmitted over encrypted networks using JavaScript Object Notation format. The records can then be processed, encrypted, and stored as read-only files. Each blockchain utilizes Application Programming Interfaces (APIs) with specific file formats to ensure the security of transactions involving data retrieval, submission, and encryption. These file structures are accessible only to authenticated APIs. Blocks obtained through the API must adhere to the file formats specified by blockchain APIs. This structure is validated before the blockchain can confirm and record the blocks in its ledgers. Consequently, each blockchain retains its localized storage within the host cloud. In a multi-cloud environment, blockchain requires the implementation of application programming interfaces (APIs) in each cloud to operate as network peers. All data traffic was secured using the TLS and SSL protocols. At the heart of the system, a smart contract builder coordinates the activities of purchasers, suppliers, and asset owners, according to the provenance rules defined for IIoT devices within the contractual terms of the blockchain. Several studies have investigated the use of provenance capture in blockchains to enhance security, privacy, and trust validation in IIoT environments [16], [17], and [18]. Blockchain technology offers significant advantages in terms of decentralization and immutability, making it a promising solution to IIoT data management and security challenges.

Despite these advantages, the latency issues inherent in blockchain implementation can limit its effectiveness for real-time continuous validation in IIoT environments. Additionally, blockchain's inherent immutability is an advantage for securing past transactions, but it does not possess built-in capabilities for real-time anomaly detection, which are crucial for maintaining security and operational continuity in dynamic environments. Most existing solutions focus on secure data storage, access control, and transaction logging challenges [21],[22] but do not specifically address the need for real-time verification of provenance data. This gap is particularly evident when considering the dynamic nature of IIoT environments and the need for timely decision making based on trustworthy data. If IIoT devices linked to an active smart contract are compromised by cybercriminals or internal threats, those responsible for managing the smart contract's implementation will be unable to detect intrusions. Such vulnerabilities can result in unethical actions such as the production of counterfeit goods, substandard work quality, unsafe procedures, and even accidents and dangers during the implementation of smart contracts.

In cloud manufacturing architecture, thousands of IIoT devices may be incorporated into the multiparty collaboration of manufacturers [6]–[8]. The continuous tracking of IIoT devices in action is a significant challenge. The modern conglomerates of several manufacturers contributing to large focal organizations in a cloud manufacturing setting may face several security threats caused by rogue IIoT devices [23]–[25]. To further understand the problem, cybersecurity threats caused by unsolicited IIoT devices are presented in the next subsection.

1.1.2 Cyber Security Threats from Unsolicited IIoT Devices

The manufacturing sector is facing significant cybersecurity challenges [2]. Approximately seventy-five percent of companies in the oil and gas sector have experienced at least one successful cyberattack, resulting in noticeable business consequences [2]. In 2017, power grids accounted for approximately 15 percent of all cyberattacks [2]. A recent report by Varonis and Forbes [26], [27] provided statistics on this issue. According to their reports, there was a staggering 1000 percent rise in malicious PowerShell scripts aimed at cyber-physical devices, with an average of approximately 5,200 attacks per month in 2021 and 2022 [26], [27]. Typically, defenses against unauthorized IIoT devices installed by insiders can create significant vulnerabilities, particularly when employing harmful and opaque algorithms [20], [25]–[29].This alarming pattern involves insiders intentionally creating loopholes [20], [27], [29]. Insider actions are responsible for 30 percent of all attacks [26], [27]. The degree to which unauthorized IIoT devices can infiltrate manufacturing networks remains unquantified. Nonetheless, insider activity accounts for 30 percent of 5,200 cyberattacks on IIoT devices in 2021 and 2022, indicating a notable trend that is anticipated to increase [26], [27]. As IIoT systems rely on computing boards, memory, and storage with very limited capacities [2], [14], [20], [27] – [29]), the inherent risks related to data authenticity and integrity cannot be addressed using the level of control possible on the cloud computing side [53], [54]. To address these issues, it is essential to view cyber security threats from a perspective distinct from that of traditional manufacturing [2], [20], [22]–[28]. In the context of Industry 4.0, industrial sensors and actuator systems have been converted into cyber-physical systems (CPS) by integrating Industrial Internet of Things (IIoT) technologies [9], [23], [30]. This shift has led to the migration of Supervisory Control and Data Acquisition (SCADA) distributed control systems (DCS),

and various new industrial control applications to cloud computing, thereby creating a framework known as cloud manufacturing. As a result of this shift, decentralized network manufacturing has developed, allowing manufacturers to share resources globally to meet a unified demand chain [33], [34], [35]. This model, which is the dominant approach for businesses utilizing cloud manufacturing, provides numerous advantages, including adaptability, efficient resource use, enhanced quality, quicker processing and delivery, and cost savings. In decentralized network manufacturing, sensor data and actuation commands are transmitted over Internet-enabled connections and interfaces that operate on a shared TCP/IP framework by using the IPv6 protocol. This shift has made industrial systems and their controllers vulnerable to cyber-attacks [36]. Such attacks can be more perilous than typical cyber-attacks targeting corporate business applications. When industrial systems are attacked, they can suffer physical harm, leading to industrial accidents or espionage. There are also considerable operational challenges, including supply chain transparency concerns regarding collected data and quality control. Primary issues were identified in [33]-[41].

(a) Validating the identity of CPS facilitated by IIoT communication.

(b) Swift tracking deployment and Internet integration of millions of CPS devices.

(c) Tracking the addition, modification, and removal of CPS devices, especially when these devices are mounted on mobile assets.

(d) Validating the reliability of sensor data that affects process events, which are interpreted from sensory inputs and the decision-making algorithms responsible for carrying out actuation commands.

(e) Defining the responsibility and accountability of individuals who possess CPS devices.

(f) Algorithmic transparency refers to accountability regarding the behavior of algorithms used to manage the operations of CPS devices.

(g) CPS devices engage in incorrect or harmful processing, which adversely affects the performance of smart contracts.

1.2 Motivation and Goals of the Study

The expansion of cloud manufacturing (CM) and widespread adoption of the Industrial Internet of Things (IIoT) in the context of Industry 4.0 [9] have introduced significant transformative possibilities to manufacturing systems. These technologies promise significant improvements in efficiency, flexibility, and cost reduction, particularly in developing economies. However, this growth introduced a range of security, traceability, and data integrity issues, resulting in IIoT hacking by malicious actors and insider trading. Such events can compromise the execution of smart contracts in favor of attackers, such as producing and transporting compromised counterfeit and unsafe products and running unsafe and unethical practices. Smart contract execution can cause quality control issues resulting in defective products, customer dissatisfaction, intellectual property theft, and legal liabilities [28],[42]–[45]. In extreme scenarios, compromised IIoT-enabled CPS devices can cause industrial accidents, resulting in loss of life and property.

Blockchain technology, which is known for its decentralized, tamper-proof, and transparent nature, has been proposed as a promising solution to these challenges by creating immutable records of device transactions [47]. Moreover, AI methods have demonstrated significant promise in analyzing behavior from data gathered by IIoT to identify and forecast misuse, irregularities, trends, intrusions, fraud, data expansion, or a combination of these elements. AI-driven predictive auditing techniques often operate in isolation [33]–[45], [52]–[54] without leveraging the transparent and immutable features of the blockchain to track and validate device behavior and interactions. However, the integration of blockchain technology for data provenance with AI-driven anomaly detection for proactive risk assessment has not been explored sufficiently.

Existing research has primarily focused on leveraging provenance blockchains to capture, record, and monitor traces of IIoT-enabled CPS devices, starting with processes such as authentication, authorization, and accounting. [14]–[19], [46], [47]. When IIoT-enabled CPS devices are allocated to smart contract tasks, their provenance details are recorded to oversee and manage the operation of the smart contracts. The designs of Prov-Trust, ProvChain, and Smart Provenance, as discussed in [16], [17], and [18], respectively, are empirical studies that demonstrate this functionality. Although capturing and recording

provenance information in blockchains initially secures, protects, and establishes trust in IIoT-enabled devices, it does not ensure ongoing security, privacy, or trust in identifying potential security threats before they occur. When IIoT-enabled assets linked to an active smart contract are targeted by malicious actors or insider traders, those overseeing the execution of the smart contract are unable to identify the breach. This is because the blockchain architecture lacks inherent features for real-time anomaly detection or predictive auditing. Additionally, current machine learning models used for anomaly detection and predictive auditing in IIoT environments often operate in isolation [33]–[45], [52]–[54] and have not integrated blockchains to analyze data streams for the detection of malicious or noncompliant activities of IIoT-enabled CPS assets linked to smart contracts.

Despite advancements in blockchain and AI technologies, a significant gap remains in the literature regarding the integration of a provenance blockchain with predictive auditing to mitigate security and transparency challenges and ensure the integrity and security of the cloud manufacturing process. This gap presents a clear need for research exploring the synergies between blockchains and AI to provide a proactive security framework.

This study aims to fill this gap by developing an integrated blockchain provenance and AI-driven predictive auditing framework. This hybrid framework aims to enhance the security, traceability, and operational transparency of IIoT systems in cloud-manufacturing environments. This study sought to enhance the security, transparency, traceability, and operational continuity of the manufacturing process by creating a unified framework that leverages the advantages of blockchain and ML. Additionally, it aims to improve quality control and strengthen the protection of intellectual property. Integrating predictive features into the provenance blockchain can verify the legitimacy of the data collected from IIoT devices, enabling swift identification of any breach in rules or controls. This functionality can be achieved solely by incorporating artificial intelligence into the provenance blockchain framework. Predictive analytics can facilitate oversight of IIoT devices that have already been verified and integrated into cloud manufacturing systems. Predictive analytics techniques can identify vulnerabilities introduced into IIoT devices, such as sensor tampering, transmission of incorrect sensory data, and improper use of assets linked to IIoT data transmitters. The outcomes of this study will shed light on the effective strategies for managing the traceability of IIoT devices that have been added, modified,

or removed. Additionally, it clarifies the accountability and liability of the individuals who own these devices. This will not only benefit manufacturers but also contribute to the broader field of industrial cybersecurity and smart manufacturing technologies, the specific goals of which are as follows:

1. Optimize blockchain-based provenance for continuous monitoring, traceability, and visibility in IIoT systems.

2. Develop a hybrid framework that integrates blockchain with ML-driven predictive auditing for proactive security.

3. Evaluate the effectiveness of the integrated framework in detecting, mitigating, and auditing security risks; ensuring operational continuity; and maintaining trust in the IIoT ecosystem.

1.3 Problem Statement

Although blockchain technology has been widely adopted in industrial settings to ensure the immutability of provenance data [14], [16], [18], and artificial intelligence (AI) has shown promise in predictive anomaly detection and risk classification [26], [27], current implementations tend to treat these technologies as isolated solutions. In the context of IIoT-enabled cloud manufacturing (C-MFG) systems, this fragmented approach results in critical gaps in ensuring real-time security, behavioral traceability, and operational resilience. Despite efforts to develop blockchain-based provenance frameworks [25], [28], [50], [51], most of these are log-centric and reactive, lacking the real-time predictive capabilities needed to detect or prevent security violations before compromising the system. Similarly, many AI-based security mechanisms have been deployed independently of on-chain provenance logs, limiting their trustworthiness, traceability, and integration with system-wide governance policies. As such, none of the existing frameworks fully leverage the synergistic potential of AI and blockchain to support real-time predictive auditing, dynamic risk mitigation, and autonomous policy enforcement within IIoT-enabled cloud manufacturing infrastructure. This lack of an integrated ML–Blockchain-based framework for predictive auditing and provenance tracking exposes IIoT-enabled systems to the following vulnerabilities:

- **Missed Anomalies:** Events occurring between audit cycles or log timestamps may go undetected, creating behavioral blind spots that rogue devices or malicious actors can exploit.

- **Delayed Threat Response:** Without real-time predictive insight, critical incidents may only be identified after substantial operational damage, increasing recovery costs, and downtimes.

- **Incomplete Traceability:** Log-based traceability alone offers limited contextual depth, impairing the ability to reconstruct behavioral violations or enforce accountability.

Consequently, the absence of a holistic, predictive, and traceable governance framework not only undermines system transparency and integrity, but also threatens continuity, safety, and stakeholder trust across the manufacturing supply chain. As manufacturing systems become more autonomous and decentralized, addressing this integrated gap becomes urgent and foundational to future-ready industrial security architectures.

1.4 Research Questions

1. How can a blockchain-based provenance tracking be optimized to support continuous visibility and traceability of IIoT-enabled Cloud Manufacturing (C-MFG)? Addresses the gaps in [21], [22], [65], [66].

2. How can predictive auditing be integrated with blockchain technology to enhance real-time anomaly detection and proactive threat mitigation? Addresses the gaps in [73], [76]-[80].

3. How can integrating blockchain and predictive auditing ensure data integrity, traceability, and security, while maintaining operational continuity in IIoT-enabled C-MFGs? Addresses the gaps in [55]-[57], [71].

1.5 Objectives of the Study

1.5.1 General Objective

The general objective of this study is to develop and evaluate an integrated blockchain-based provenance tracking and ML-driven predictive auditing framework to enhance security, traceability, and operational continuity in IIoT-enabled C-MFGs.

1.5.2 Specific Objectives

The specific objectives include:

- To design a provenance-enabled blockchain framework for real-time tracking and monitoring of IIoT devices in dynamic operational environments.
- To develop a mechanism for tracking state changes using off-chain storage for real-time data collection and on-chain logging for batch recording.
- To design and implement predictive auditing mechanisms using machine learning algorithms to identify potential threats or anomalies in real-time.
- To develop a framework that integrates provenance blockchain and predictive auditing to ensure data integrity, traceability, and adaptive security enforcement in IIoT-enabled cloud manufacturing.
- To develop a consensus mechanisms that allow blockchain systems to adapt to real-time data, ensuring efficiency and security in dynamic IIoT environments.
- To evaluate the framework through simulation-based experimentation, assess its effectiveness in detecting security threats, maintaining operational continuity, and optimizing decision-making processes in real-time IIoT monitoring.

1.6 Major Contributions of the Study

The following are the four major contributions of this study that advance the security and operational continuity of IIoT-enabled CPS within the context of the C-MFG:

(a) Continuous monitoring and validation of IIoT devices

A significant contribution of this study is the development of continuous monitoring and validation of IIoT devices that are linked to smart contracts within a blockchain.The framework employs a predictive auditing algorithm to continuously monitor real-time data streams from IIoT devices, ensuring their operations align with the provenance constraints established in the blockchains. This enables the system to detect potential anomalies based on continuous learning from the behavioral analysis of IIoT devices before escalating into significant issues. The predictive auditing mechanism extends beyond post-event detection to offer proactive risk mitigation, thus enabling faster responses to potential threats in IIoT environments. In real industrial systems, this design is expected to detect anomalies relevant to many aspects such as counterfeit products, unsafe products, unethical practices, quality control issues (that may result in defective products, customer dissatisfaction, and legal liabilities), and intellectual property protection challenges.

(b) Optimized blockchain architecture for Real-Time IIoT Data Provenance

This study developed an optimized blockchain architecture to address the high computational overhead and latency issues inherent in the existing blockchain implementations [55]–[57]. Integrating the on-chain and off-chain hierarchical structures and optimizing the algorithms to meet the high-throughput and low-latency requirements of IIoT makes it more suitable for time-sensitive IIoT applications without compromising the integrity or security of the data. This design balances computational efficiency with data transparency by temporarily storing real-time IIoT data off-chain for rapid access and analysis, while committing hash values of the data to the blockchain at predefined intervals. This ensures the integrity of the data without overburdening the blockchain, thereby making the solution practical for resource-constrained environments.

(c) Development of an Integrated Security Framework for IIoT in Cloud Manufacturing

Another significant contribution of this study is the development of a security framework that integrates ML-driven predictive auditing for real-time anomaly detection with blockchain technology, to ensure immutable audit trial records. This integration allows for the early identification of noncompliant or potentially compromised IIoT devices before they can inflict any harm. By harmonizing these two technologies, the proposed framework broadens the concept of provenance by verifying and authorizing the continuous reliability of IIoT devices to ensure full operational compliance in accordance with the constraints established by smart contractors within a blockchain specifically designed for provenance.

(d) Ensuring End-to-End Traceability

This study extends the existing blockchain-based provenance body of knowledge to full spectrum traceability, which is limited to authentication. By leveraging blockchain's ledger to track not only data provenance but also device interactions, system status, and operational anomalies in real time. This enables end-to-end traceability of the data lifecycle, ensuring that every piece of data generated by IIoT devices is recorded securely, tamper-proof, and transparent. Such a system offers substantial improvements in accountability, data integrity, and security in cloud-manufacturing environments.

1.7 Publications

The findings and contributions of this study have been disseminated in the following publications:

1. **Journal Publication:** "Umer, M.; Belay, E. and Gouveia, L. (2024). Leveraging Artificial Intelligence and a provenance blockchain framework to mitigate risks in cloud manufacturing in Industry 4.0. Electronics, 13(3), 660. ISSN: 2079-9292. DOI: 10.3390/electronics13030660"

2. **Journal Publication:** "Umer, M.; Gouveia, L. and Belay, E. (2023). Provenance blockchain for ensuring IT security in cloud manufacturing. Frontiers in Blockchain, 6:1273314. DOI: 10.3389/fbloc.2023.1273314".

3. **Journal Publication:** "Umer, Mifta Ahmed, Elefelious Getachew Belay, and Luis Borges Gouveia. 2024. "Fortifying Industry 4.0: Internet of Things Security in Cloud Manufacturing through Artificial Intelligence and Provenance Blockchain A Thematic Literature Review" Sci 6, no. 3: 51. https://doi.org/10.3390/sci6030051"

4. **Book Chapter:** "Mifta Ahmed. (2023). BlockchMLn for integrated logistics engineering. In Blockchain for Integrated Logistics Engineering (pp. 60–68). MKSES Publication Lucknow. https://doi.org/10.5281/zenodo.10279885"

1.8 Significance and Impact of the Study

This study's academic importance stems from its contribution to the existing body of knowledge on blockchain technology and auditing practices. Specifically, it advances the understanding of how blockchain can be integrated with predictive auditing to create a continuous verification of provenance information through machine learning using the state transition rules engine of a smart contract loaded on a blockchain to continuously monitor the compliance of operating assets within the boundaries defined by the smart contract. This study also expands the theoretical understanding of continuous IIoT provenance validation during operations. This thought domain may also provide new insights into how emerging technologies can be leveraged to enhance the auditing processes.

From a practical perspective, this study offers a solution to the pressing need for the real-time monitoring of IIoT data in cloud manufacturing systems. By combining the transparency and immutability of blockchain with a predictive proactive auditing approach, this study provides a novel approach for enhancing the efficiency, security, and accuracy of audits in manufacturing environments. With provenance and predictive auditing in place through blockchain, manufacturers, auditors, and regulators may obtain the following benefits.

(a) Every IIoT device is identified at the time of installation or removal using its ID, ownership details, manufacturer details, running algorithm identification, and role.

(b) Every IIoT device is registered with full metadata in an intrinsic blockchain registry.

(c) Every IIoT device will have a recorded history of ownership changes and deployments quickly accessible by predictive auditing, even if the IIoT device memory does not have it recorded.

(d) IIoT device deployments by hackers will fail because the unregistered IIoT device will be rejected, and an attempt to install it will be highlighted.

1.9 Structure of the Dissertation

The first chapter presents the background, context, specific details of the research, objectives, motivation, and significance of the study. The rest of this dissertation is structured as follows.

Chapter 2: Literature Review and Theoretical Framework provides the most relevant literature on blockchain, AI, and IIoT security and discusses the gaps in the literature that this research seeks to fill. Furthermore, it provides the theoretical framework that underpins this study.

Chapter 3: This chapter presents a comprehensive analysis of the research methodology along with the Proposed Framework. The design details of the provenance blockchain and AI are reviewed, including the architecture and technologies applied in this study.

Chapter 4: Discuss the experimental results including framework's performance in detecting and mitigating security threats in real-time

Chapter 5 Interprets the findings in line with the research questions, compares them with existingliterature, and discusses the theoretical and practical implications.

Chapter 6: The conclusion summarizes key findings, limitations, and future research perspectives.

Chapter II

Literature Review and Theoretical Framework

2.1 Introduction

The background was established in Chapter 1 to gain a relevant understanding of background knowledge to establish the context and help highlight the research problem, motivation, and contribution. This chapter presents a comprehensive review of existing research on blockchain, AI, and IIoT security, identifies the limitations of current approaches, and highlights how this research fills these gaps. The chapter also outlines the key theories and concepts that form the basis of the proposed framework and explains how they inform the design and analysis methods used in this study.

2.2 Cybersecurity Risks to IIoT-Enabled C-MFG

Cloud manufacturing marks a transformative shift from conventional industrial systems [6]. Programmable Logic Controllers (PLCs) have been converted into Internet of Things (IoT) devices featuring advanced firmware and electronic upgrades that enable communication through TCP/IP protocols. This evolution allows them to connect via open-standard wireless networks to the Internet, thereby becoming cyber-physical systems (CPS) [6] [58]. In this contemporary landscape, the integration of information technol-

ogy, systems, and communication (ITSC) infrastructure with the operational framework of manufacturing organizations is embodied in the Industry 4.0 paradigm [2]. To adapt to rapidly changing market conditions, consumer preferences, and business environments, manufacturing organizations have adopted state-of-the-art technologies such as the Industrial Internet of Things (IIoT), autonomous and adaptive robotics, and machinery [3].

As artificial intelligence (AI) rapidly evolves, driven by advances in machine learning algorithms and the subsequent automation of decision-making and field actions, the reliability of the data sources utilized by the AI system and the effectiveness of machine learning algorithms have become significant issues [9],[20],[29],[30]. AI systems gather data streams from field sensors that are part of the industrial and logistics infrastructure by utilizing the IPv6 communication protocol (commonly referred to as cyber-physical systems or the Industrial Internet of Things). However, at a higher level, there are issues regarding accountability, traceability, fairness, and transparency of the data gathered by AI systems. This necessitates the detection, tracking, and capturing of all data manipulation mechanisms in IIoT devices to achieve a thorough understanding of provenance data [14], [23]–[25].

The streams of data gathered by machine-learning algorithms are utilized to make decisions regarding operational sequences and send actuation commands for the functioning of industrial machinery within advanced automation systems [23]–[25]. In automation frameworks, the functions performed by machines, equipment, and robots must be validated by their end customers to confirm their reliability and adherence to agreements that have been signed and executed. Conventional data visualization systems (such as database queries) are considered unreliable owing to their reliance on single-point capture. Reliable data visualization can be achieved when various sources are integrated. Blockchain technology has significantly progressed in recent years. One can perceive the blockchain as a network comprising distinct data-mining participants that validate and confirm transactions, posting their information as “decentralized ledgers” or “blocks” blocks’ that are duplicated for all members of the network [54]. Once these blocks are shared, they cannot be compromised because the data mining participants are concealed from conventional attack vectors on the Internet or cloud computing.

In contrast to conventional transactional records, these blocks are not kept on centralized ERP servers, making unauthorized alterations possible [54], [59]. Instead, storage systems are distributed across the blockchain and secured using sophisticated cryptography. Only members of the blockchain have access to these storage systems. The use of advanced cryptography and digital accountability facilitated by digital signatures guarantees that transactions are permanent and safeguarded against unauthorized changes. Additionally, new transactional members who wish to join the system must be verified by current members rather than relying on a single authentication method. Members are introduced through smart contracts that outline rules for their interaction and validation. Smart contracts function as transactional agreements rather than as master agreements linked to traditional paper contracts [59]. This means that transactions are executed under transaction-level smart contracts comprising all details related to the transactions needed by blockchain members who have executed and digitally signed the contract. This essentially implies that the transaction will be terminated if all transaction-specific details required by the network members in the blockchain are not populated in the smart contract. These are called state rules, defined at low levels for every real-world device assigned to a smart contract to execute its terms. For example, if a crane is assigned to a smart contract, several state rules regarding the operational constraints allowed for the crane should be built within the smart contract. These rules capture ownership and accountability. These rules validate transactions, and are reliable for blockchain members who invest in the execution of smart contracts. In Industry 4.0, smart contracts are required when multiple manufacturers collaborate to execute joint manufacturing processes by using cloud computing (cloud manufacturing).

Blockchain possesses architectural features and components that are ideal for establishing cybersecurity measures to safeguard IoT in industrial settings. Cryptographic methods can be implemented within blockchains to secure IoT devices during communication via individual radio frequency identification [60],[61]. IIoT sensors collect data from industrial processes from the points of events and transmit them to cloud computing applications used to visualize these events during the running processes [62]. Such visualization forms a perception layer regarding the processes and events occurring in the physical layer, thus making the entire supply chain transparent [62]. In a large-scale collaborative system, the events occurring across all echelons of a supply chain are visible in real time to man-

agers by virtue of the data collected from IIoT networks deployed in manufacturing and logistics engineering systems at the echelons [54],[59],[62]. If multiple organizations collaborate in a manufacturing and supply chain system, supply chain events are accessible to wider groups of individuals through their respective computers or mobile terminals. They can not only see the sequence of events through real-time data visualization but also utilize real-time data analytics with the help of big data technologies. Furthermore, digitized and automated supply chains comprising artificial intelligence have enabled decision making for engineering-level field actuations based on the data collected on events. The decisions made by artificial intelligence are based on enormous industrial engineering transactional knowledge entered in big data systems, which are used as baselines for comparing the data received from IIoT systems. Decisions are made in real time to generate automation commands for manipulating the operations carried out by industrial machines, robots, and equipment. These commands are issued through actuators attached to devices configured to receive data from the cloud (unlike sensors, which are configured to receive data from process events).

The threats in Industry 4.0, software systems, and applications differ from those in the information systems used in Industry 3.0. Threats arise from diversity, a lack of standardization and controls, and limited onboard resources on IIoT devices deployed in industries [28], [50], [51]. Numerous risks are associated with insecure IIoT devices. Some attack scenarios using a rogue IIoT device inserted into a running manufacturing network are as follows [23]–[25], [28], [30], [50], [51]:

(a) Eavesdropping attack: Attacks caused by unauthorized entry of penetration systems into the running process sequences.

(b) Masquerading attack: attacks caused by penetration systems falsely appear to be genuine and are authorized to run security checks on the processes.

(c) Distributed Denial of Service (DDoS) attacks occur when systems capable of producing vast quantities of fraudulent data streams overwhelm network connections, computing power, memory, and storage.

(d) Side-channel attacks: penetration attacks caused by less-monitored and scru-

tinized side channels connected to the main entry channels into the running processes.

(e) Cross-site scripting attacks: involve scripts that can be mixed with existing scripts controlling industrial processes, leading to penetration attacks.

(f) Automated code-based attacks: caused by automated codes called auto bots that can generate behaviors similar to those of human access to industrial information systems.

(g) Exploit-based attacks: caused by exploits, with payloads appearing as genuine traffic capable of exploiting known vulnerabilities in software systems.

(h) Identity thefts (of authorized IIoT devices): penetration attacks occur when the authentication and authorization information of IIoT devices is stolen, allowing malicious devices to be substituted in order to inflict specific damage on industrial systems.

(i) Insider trading and proliferation: Individuals with insider access exchange confidential security details with their partners outside the organization to compromise industrial systems.

(j) Deceptive sensor data input and actuation attacks in control systems: Devices are manipulated by unauthorized personnel to stream fake sensor data into the monitoring and control software running on cloud computing, such that the software makes wrong control decisions leading to actuation signal flows, causing damage to industrial systems, which can be a very targeted attack. For instance, an attacker may manipulate the system by reducing the valve pressure in a crucial pipeline, prompting the control system to incrementally increase the pressure, which could result in an explosion. This scenario might occur because insider traders have access to equipment; however, the possibility of external attacks causing such incidents cannot be dismissed.

To protect cloud manufacturing systems from potential threats, it is crucial to implement a comprehensive control-system framework that addresses multiple aspects of industrial computing systems. Central to this framework is the security of the Industrial Internet of Things (IIoT). Enhancing IIoT security requires continuous and accurate monitoring of its introduction, deployment, and redistribution in industrial systems and processes. Recent academic research has identified a provenance blockchain as a potential solution, which is explored in the subsequent section.

2.3 Provenance Blockchain for Securing IIoT in C-MFG

At its core, provenance can be described as dynamic metadata, which is essentially data about data linked to information to pinpoint the individuals, events, and contexts involved in their creation and alterations over time [24]. These dynamic metadata are generated sequentially in conjunction with the operations performed on the data, marked by dates and timestamps [25]. Consequently, this method is useful for reverse tracing in forensics. In the era of Industry 4.0, provenance has significantly extended its applications, particularly in algorithmic logging and the traceability of events produced in cloud computing [31],[32]. Carata [46] laid the foundation for understanding provenance metadata by emphasizing their unique role in ensuring backward traceability and capturing the essential elements of a system's operation. Unlike general metadata, provenance metadata must be carefully structured to document interrelationships, thus enabling fine-grained insights into software processes. Carata identified the key components of automated provenance capture, including files, databases, operating system processes, and memory maps, while also addressing manual methods such as parsing engines to capture human-related data. Importantly, this study underscores the need for metadata integrity and security, advocating robust access-control mechanisms and encryption. Although this research provided crucial insights for conceptualizing algorithmic designs for provenance systems, its focus was primarily on static data, leaving gaps in its applicability to dynamic, real-time environments.

Building on Carata's work, Suriarachchi [63] explored provenance metadata capture in

big-data streams. With the growing importance of provenance in data-intensive computation environments (e.g., Industry 4.0 systems enabling data streams from process engineering sensors), a solution for the automated capture of provenance metadata regarding data sources or content was investigated in this research. Suriarachchi [63] found that the complexities of provenance metadata structures may at times result in metadata records becoming larger than the actual data that they describe. Suriarachchi [63] investigated a mechanism of provenance streams that captures reduced provenance data using a system called "global reducer." This system uses multi-processing with parallel executions of stream captures and conducts on-the-fly stream pre-processing and processing using a one-pass algorithm to reduce and capture provenance data. However, a significant limitation of this approach is that the provenance data cannot be constrained based on capture rules, which increases the risk of data manipulation and false identifiers. Hence, there is a finite chance that eavesdroppers play with the system, resulting in false identifiers and misdirected traceability. If such vulnerabilities are realized, the purpose of having a provenance system as a governance and control mechanism can be lost.

Blockchain technology has been widely explored for enhancing security and trust in Industrial Internet of Things (IIoT) environments, particularly focusing on data provenance and authentication mechanisms. Blockchains can make the industrial system and networking closed and accessible only to authorized members, such that an attacker may not be able to insert any form of a rogue IIoT device into a running network. One may imagine a blockchain network in which only authorized devices are allowed to run processes through registration in smart contracts and monitoring and control using smart ledger updating. Such systems have been the focus of many recent studies to find solutions to IIoT threats and vulnerabilities identified in several other studies [23]–[25], [50], [62]. Ali [67] modeled a traceability algorithm using a provenance data capture and policy-based system in a blockchain design. Ali [64] examined policy enforcement and traceability by using provenance data traceability in cloud computing. With the massive deployment of services in cloud computing in the application, platform, and infrastructure layers, the performance, reliability, and ownership of these two critical factors are key problems. This study investigates how provenance data traceability can be used to solve these problems. This research contributed to the development of an enhanced provenance traceability model based on an earlier model called "PROV-DM (Prov Data

Model)" developed by W3C (2013). However, this study overlooked the comprehensive traceability of device interactions within IIoT ecosystems. Building on these insights, Nwafor et al. [20] presented a data model for provenance captured by a blockchain network from each IIoT device before permitting it. The data model captures the details of all agents and activities related to the IIoT device to form a provenance-sensing model. In the blockchain-capturing mechanism, Nwafor et al. [20] created a mapping algorithm between the provenance data and IIoT sensor being audited. The "tracer" in the blockchain should be a microcontroller having knowledge about the common tracing format used by the blockchain network, which is translated into the provenance data model by breaking down the individual data units. Although both studies underscored the critical importance of understanding how to track and audit data in the IIoT plane, they did not sufficiently integrate these processes into the cloud manufacturing plane for a comprehensive traceability. This oversight reveals a research gap that future studies should address by integrating IIoT and cloud planes in the model to form comprehensive traceability and an integrated risk visualization framework. Several studies have explored provenance data with bindings between device metadata and ownership data, as proposed in recent studies [17], [20], [23], [29], [61]. However, there are multiple issues related to provenance data capturing and management, such as breach of confidentiality and integrity of data bindings, privacy and access control of data, fresh (latest) data availability, cryptography, reliable forensics, issuance and management of cryptographic keys, and the overall privacy of cloud computing virtual boundaries.

The concept of system interaction of a provenance blockchain and its algorithm was presented in [17]. The design includes a trust validation mechanism that uses user registration, data encryption, data sharing, and data validation by verifying the keys and ownership of the users. The keys are issued by a decentralized PKI network. The system comprises a provenance auditor interfacing a blockchain network hosting IIoTs. The cloud service provider is protected by the blockchain network because only data from validated IIoTs reach the cloud-hosted sensing and control applications. These solutions typically capture data from IIoT devices during registration or at key operational checkpoints, and store information in the blockchain for future reference. A simplified representation of the interaction architecture is shown in Figure 3.

In ProvChain, ledger transactional data are stored locally on blockchain nodes, whereas provenance auditors store validation records on servers. This design also presents the block structure of integrated provenance and ledger transaction data. However, the significantly long hash keys are owing to the serial appending of the previous block hash in the next block. At some stages, message sizes may comprise infinitely long hashes with limited data per block. These frameworks successfully generate secure auditable records. However, their high computational overhead and inability to process continuous data streams render them impractical for IIoT systems, which require rapid and responsive security actions.

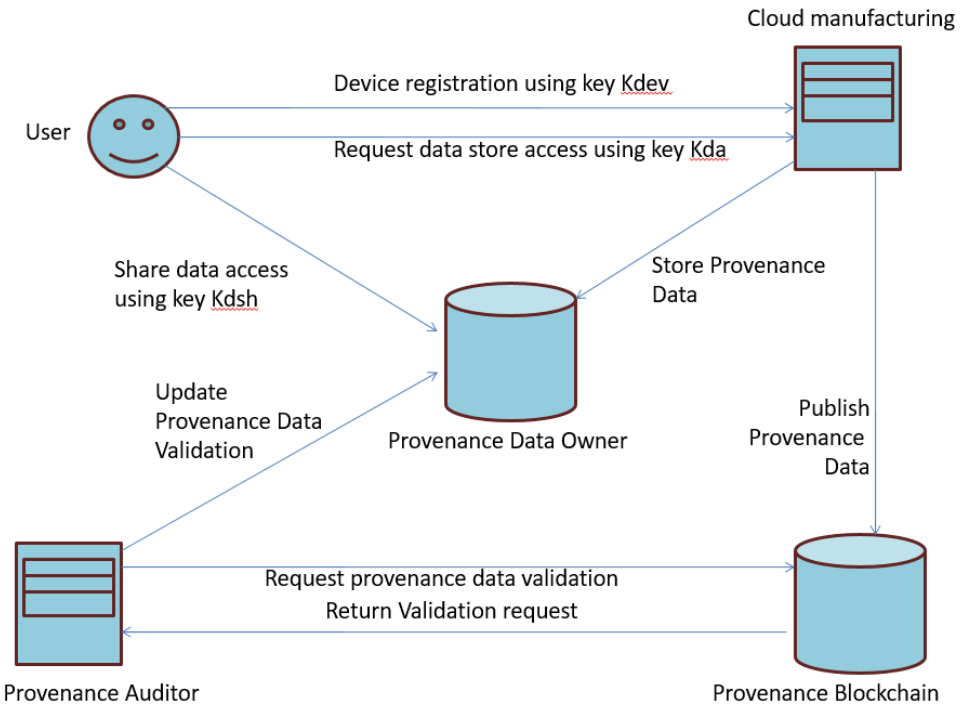


Figure 3: Simplified representation of ProvChain provenance blockchain based on [17].

Furthermore, Wang et al. [65] proposed a blockchain-based lightweight message-authentication scheme for IIoTs in cross-domain scenarios. In [65], the focus was on addressing the efficiency and security challenges. Similarly, Cui et al. [66] proposed a blockchain-based cross-domain authentication scheme to facilitate secure communication for cross-domain IIoT devices. Although these studies achieved notable results in utilizing blockchain for data provenance and authentication in IIoTs, a limitation remains in addressing the complete traceability of device interactions. Several studies have explored the use of blockchains for provenance tracking in IIoT-enabled CPS environments to recognize, reg-

ister, record, and control IIoT cyber-physical devices [16], [67]. These studies recommend the storage of provenance data at two locations: the data store of the provenance data owner, and the provenance blockchain. Whenever a new device requests registration and shares its provenance data for this purpose, it requests a cloud-manufacturing application. However, a validation cycle is triggered to validate the data from the blockchain. The user sharing the provenance data also needs to send a data store access request to the cloud manufacturing application to publish the provenance data in the blockchain. Cloud manufacturing applications allow temporary storage of provenance data. When a data storage event occurs, the provenance auditor auditor of the provenance blockchain confirms the data before the provenance data shared by the device is permanently stored in the data repository of the provenance data owner. If the validation failed, the temporary provenance data stored in the data store of the provenance data owner were deleted. Although this approach provides a high level of data integrity, continuous monitoring is often lacking, leaving systems vulnerable to threats that occur between logging intervals. In addition, the computational overhead of blockchain can introduce latency, making it difficult to implement in real-time environments.

In a recent study [68], these interaction topologies and algorithms were enhanced by introducing a secure provenance-based smart contracting framework. Likewise, Malik et al. [69] developed a framework for the PrivChain architecture. Figure 4 illustrates a simplified version of the secure provenance-based smart contracting architecture, and Figure 5 depicts the privacy chain architecture.

Figure 4 in [68] illustrates a cycle that can be employed to authenticate the origin of any product acquired from a supply chain by scanning the QR code label for its data. This concept was similarly explored in [69] using a secure provenance-based smart contracting system, as depicted in Figure 5. The primary differences between the workflows in Figures 4 and 5 include the fact that, in the latter, consumers interact directly with the blockchain for provenance verification, which works in conjunction with the blockchain to ensure that payments are processed only after the provenance records have been verified and confirmed, and there is no auditor involved in the provenance validation process. In this configuration, the provenance data are stored solely in the blockchain, with no temporary storage used for validation.

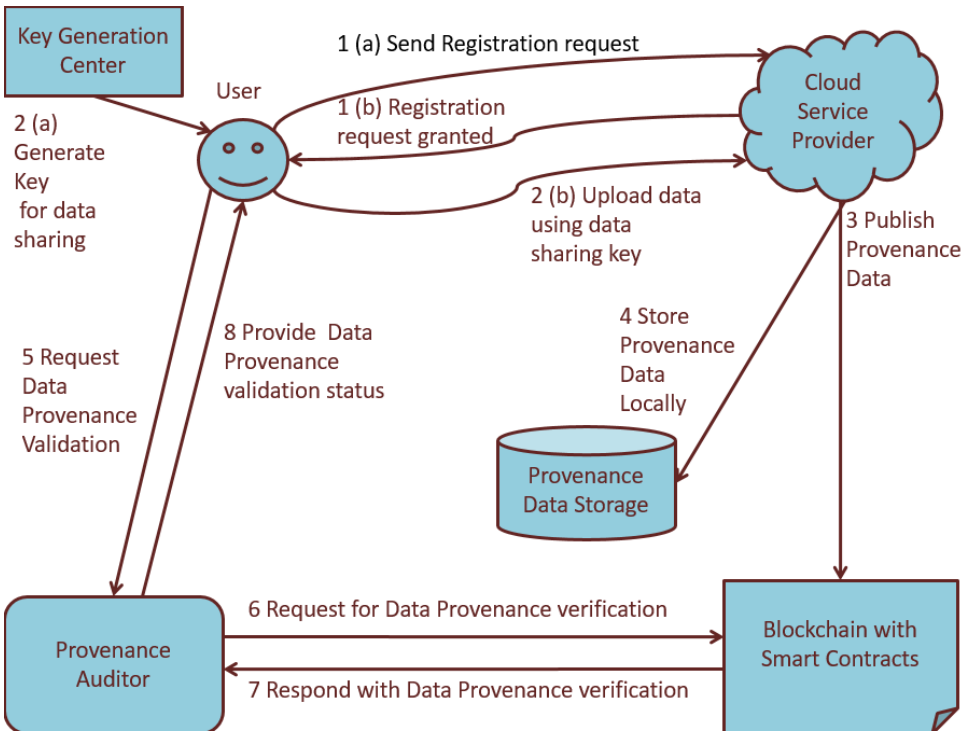


Figure 4: Redrawn representation of secure provenance-based smart contract architecture [68].

These two studies proposed an architectural framework in which a provenance blockchain serves as a trusted intermediary. This blockchain is tasked with verifying "secret information" through cryptographic techniques without revealing the actual content. Within this framework, studies advise storing provenance data in the provenance blockchain and maintaining cryptographic keys during storage. Provenance data are stored in two locations: with the cloud service provider in the designated space of the data owner and on the blockchain, which includes smart contracts linked to provenance information. When a new user wishes to register data (using a temporary key provided by the key generation center), they upload the data to a cloud-hosted application for verification. This action triggers a validation process to authenticate data via a blockchain. The cloud-hosted application temporarily holds provenance data in the provenance data store before publishing it in the blockchain. Following this, the user requests that the provenance auditor confirms the provenance of the data. The auditor then consults the provenance blockchain to verify the data and the verification result is relayed to the user.

Similarly, several studies [15]–[19], [20], [30] explored the use of provenance in a blockchain with algorithmic validation (built-in smart contract rules) to enhance security in indus-

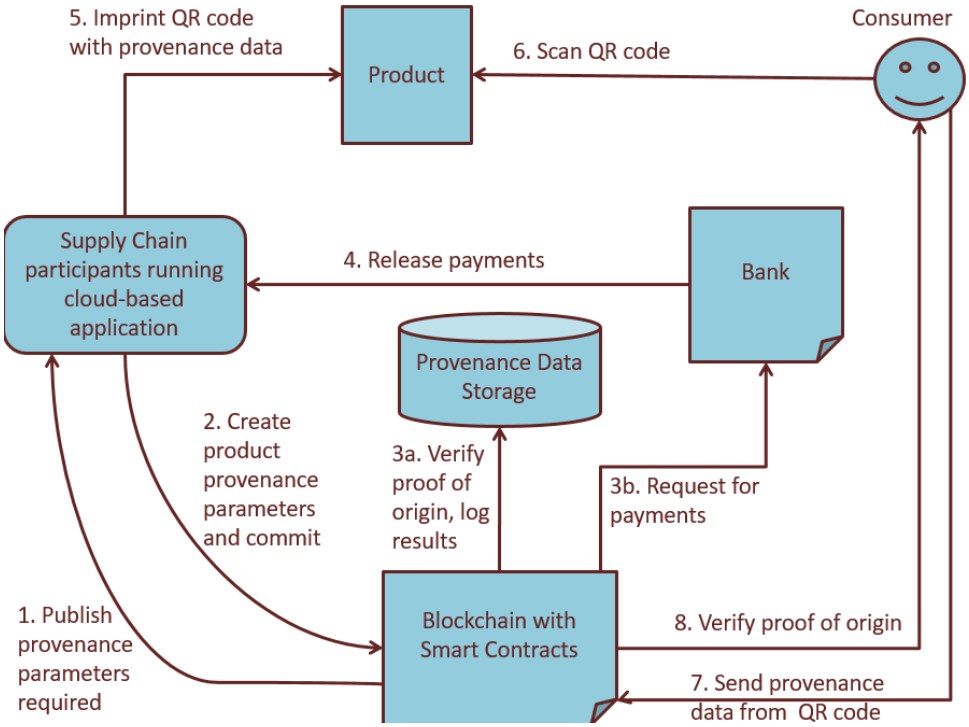


Figure 5: Redrawn representation of PrivChain [69].

trial Internet of Things (IIoT) environments. Although this approach ensures the initial security, privacy, and trustworthiness of the devices allocated to smart contracts, continuous monitoring is often lacking, leaving systems vulnerable to threats that occur between logging intervals. At the fundamental level, blockchain security has a limited influence on IIoT devices because the devices operate outside the blockchain network, and blockchain smart contracts are dependent upon the sensory data collected from the IIoT devices to validate the completion of contractual terms. The concept of smart contracts can be used for reliable execution of transactional contracts in cloud manufacturing. However, the reliability of the transaction data collected from IoT devices must be ensured. This is because the data streams arrive from IIoT devices that are spread over the operating fields. Just one validation of their ownership, purchase, deployment history, etc., does not justify IIoT security, as they can be breached after the initial validation. Continuous verification of the provenance metadata captured in the IIoT network (against counterfeits and eavesdroppers) is critical to enable rapid threat detection and mitigation.

Blockchain technology provides significant advantages in terms of security, decentralization, and immutability, making it a promising solution to IIoT data management and security challenges [22],[70]. A study [70] proposed the EdgeShare framework to enhance

the overall performance of IIoT using blockchain and edgecomputing to increase the efficiency and security of data sharing across heterogeneous network domains. Similarly, Latif et al. [36] proposed an ultra-lightweight blockchain-based architecture for the access control of IIoT sensors and actuator data in an industrial environment. Several solutions have emphasized secure data storage, access control, and transaction logging. This discrepancy is particularly relevant in the context of the dynamic nature of IIoT environments and the need for timely decision making based on trustworthy data. To address this gap, Paper [23] presented challenges in provenance verification systems to accurately identify IIoT devices and build the traceability of doubtful devices in a network. These challenges include the provenance detection of bindings, fault tolerance, integrity, confidentiality verification through data, chain and origin integrity verification, access control, and protection of keys during sharing. Building on Paper [23], a study [67] mentioned the integrity challenges and added proof of work and proof of authority verifications and consensus while extracting provenance metadata from IIoT systems comprising high- and low-level information and enforcing the completeness and security of provenance metadata through enforced lineage, verifications, data point locking, parallel verifications, and privacy.

Several studies have highlighted the importance of low latency and high throughput in existing blockchain implementations to address the real-time constraints. Yu et al [70] introduced LayerChain, a hierarchical edge-cloud blockchain that efficiently reduces the block propagation time and system resource requirements, making it suitable for large-scale low-delay IIoT applications [22]. Umran et al [71] and Latif et al [22] propose using a private blockchain mechanism with a lightweight security scheme and a low-power ARM Cortex-M processor to improve computational execution and ensure secure authentication, scalability, and and lightweight architectures, to meet the high throughput and low latency requirements of IIoT. Moreover, a study [58] delved deeper into provenance security by defining the layers of audit, enforcement, and regulation planes and building a system-wide policy framework to enforce predictive auditing and governance of data flow within an application context and between application contexts. Another study [69] further emphasized the security provenance verification challenges faced by existing blockchain architectures, and suggested predictive auditing to verify compliance and identify violations retrospectively and proactively when using machine learning.

A prominent gap identified in the literature is the limitation of continuous provenance data verification, where devices constantly exchange data and operational parameters and security risks can emerge between periodic updates, which blockchain systems typically rely on for validation. Several studies have explored the use of blockchain for enhancing security and data provenance in Industrial Internet of Things (IIoT) environments. Blockchain technology offers significant advantages in terms of security, decentralization, and immutability, making it a promising solution for IIoT data management and security challenges[22], [70], [73]-[76]. However, there is a gap in addressing continuous and real-time provenance verification in the existing blockchain-based IIoT systems. Additional layers or layers of control are required to ensure the continuity of security, privacy, and trust. The solution is a continuous validation, which is possible through continuous auditing. Furthermore, to detect and conduct timely interventions, predictive capabilities of continuous auditing are required. These are mentioned in the limitations of references [15]–[19], [20], and [30]. Because IIoT devices have limited edge-computing protection, their validity requires predictive audit capabilities. This could involve developing lightweight consensus algorithms tailored for real-time provenance verification or incorporating machine learning techniques to detect anomalies and ensure data integrity in real time. In addition, the use of smart contracts, as mentioned in [70], can be extended to automate and expedite the provenance verification process in an IIoT environment.

2.4 AI for IIoT Security

The significance of artificial intelligence in enhancing IIoT security has been the focus of numerous recent studies aimed at protecting devices, networks, and applications, either at the endpoint or within cloud-computing environments. AI-driven predictive auditing solutions have been developed in various studies, as noted in references [33]–[45] and [52]–[54]. These solutions use AI techniques to proactively detect and address potential security threats prior to their occurrence. Currently, predictive audits in IIoT security are powered by machine-learning and deep-learning algorithms. AI methods are employed to process large datasets, detect anomalies, and identify patterns that conventional security systems may miss [77]. Various machine learning algorithms, such as support vector machines (SVM), random forests (RF), extreme gradient boosting (XGBoost), neural networks (NN), and recurrent neural networks (RNN), have demonstrated their effectiveness in detecting and classifying different IoT attacks with satisfactory results [78]. These techniques enable continuous system assessments to ensure adherence to security standards and improve detection and response to breaches [77]. The integration of AI with IoT security has yielded promising advancements in threat detection, predictive analytics, and overall security measures [79]. Machine learning (ML) techniques have been proven to boost the effectiveness, speed, reliability, and efficiency of cybersecurity strategies for IoT devices [80]. With AI's integration of AI into the application layer, controls are crafted to supervise and manage IIoT security both upstream (from the field to the application) and downstream (from the application to the field) within cloud computing environments [82]. As IIoT devices do not relay all their data to cloud systems because of the increasing strain on these systems, there has been a shift towards edge computing [88]. Therefore, AI in cloud computing should feature a data-analysis framework that bridges both cloud and edge computing. This approach is particularly significant for cloud manufacturing. Achieving this can be facilitated through blockchains, where peers interface with edge-computing databases. The foundational layers for authentication, authorization, and access control privileges must be established for edge computing. AI analytics should be hosted in cloud computing to integrate multiple levels of edge-computing infrastructure. Utilizing artificial intelligence to analyze behavioral data from IIoT can efficiently identify and predict misuse, anomalies, patterns, intrusions, fraud, data proliferation,

or a combination of these elements [83]-[86]. An AI-driven security layer is considered more advanced than conventional signature-based detection techniques. The success of AI relies on the purity, precision, relevance, and organization of the data, which requires comprehensive data engineering prior to analysis. For AI systems to effectively detect and predict, they must be trained using established databases that contain intrusions and anomalies. This analytical process requires the extraction and normalization of pertinent features from the data during pre-processing [87]. Algorithms used in machine learning that excel at detecting anomalies in data from IIoT devices include decision trees, random forests, k-nearest neighbors, support vector machines, and artificial neural networks. These networks incorporate features such as convolutional layers, backward propagation, and long short-term memory [81],[86]-[88]. Among these, decision trees, random forests, and artificial neural networks excel in both detection and predictive analysis, whereas k-nearest neighbors and support vector machines are predominantly effective for detection. AI systems should be engineered to autonomously recognize and foresee misuse, anomalies, patterns, intrusions, fraud, data proliferation, or any combination thereof by detecting manipulations in AI inputs compared with expected logical values [86]. To achieve this, developers should incorporate detection and prediction capabilities into a rule-based framework that is specifically designed for IoT data analysis. The range of attack complexities includes basic input manipulations and botnet attacks, extending to more sophisticated and severe threats such as dataset poisoning, algorithm poisoning, and model poisoning. AI should be proficient in detecting rule infractions, whereas developers should excel in crafting comprehensive rules that effectively address known attack scenarios.

An AI automation strategy based on rules can effectively defend against a range of attacks on IoT devices. This includes evasive attacks such as masquerading and eavesdropping, malicious code injections such as cross-site scripting and side-channel attacks, and flooding attacks [86]. However, to detect or predict more complex threats, such as counterfeit IoT, wormhole/sinkhole/link ranking, exploit-based attacks, DDoS, and injection of false sensor data, it is crucial to employ more advanced techniques [81],[87],[89],[90]. Addressing these threats necessitates the use of both supervised and unsupervised learning along with reinforcement and deep learning strategies. These advanced approaches are essential for identifying both external and internal threats. Identifying patterns might

only be achievable after extensive training of neural-network black boxes, which could include multilayered perceptrons or a combination of decision trees and random forests [81],[87],[89],[90],[91]. Therefore, it is crucial to create layers for gathering, organizing, preprocessing, and analyzing large datasets to facilitate supervised and unsupervised learning as well as reinforcement and deep learning methods in artificial intelligence [45].

Currently, there are a limited number of studies that combine AI with blockchain technology to improve IIoT security; however, it is expected that more research will be conducted in this area in the future. In the realm of AI, blockchain technology is vital for guaranteeing the collection and storage of dependable and immutable data, which is crucial for performing both detection and prediction analyses [92]-[95]. Data can be stored either on- or off-chain, but their integrity is maintained using a smart contract rule engine. Although both types of data can be encrypted, on-chain data require digital signatures from blockchain participants for authentication. Regardless of whether the data are stored on or off the chain, their trustworthiness remains high because of the chain-code validation performed by the blockchain peers in accordance with the smart contract rule engine. The application of blockchain technology may not extend to all IIoT devices in edge computing, because it is restricted to those involved in executing smart contracts. Consequently, data collection for AI-driven predictions and detection should be confined to IIoT devices within this scope, leading to the utilization of both on- and off-chain data storage. AI systems can be developed to take advantage of both the storage types. Although both on-chain and off-chain data can be used in the detection and prediction processes, testing data might be limited to on-chain sources, and decision making can be implemented on IIoT devices integrated into smart contracts. The literature review in earlier sections provides a crucial foundation for this study. The next section critically examines the knowledge gaps in existing research.

2.5 Summarizing the Gaps in Literatures

The existing literature on Blockchain Technology, Predictive Auditing, and Industrial Internet of Things (IIoT) provides valuable insights into the individual components of these technologies; however, significant gaps persist regarding their integrated application within the context of cloud manufacturing. The following sections highlight the key gaps that this study aims to address:

(a) Lack of continuous Provenance monitoring and Validation for Dynamic IIoT Tracking

In the context of IIoT, provenance refers to the systematic tracking of device interactions, state changes, and data transfers. Traditional models [17], [68], [69] focus on post-event logging, limiting their ability to detect threats in real-time. While most existing research on IIoT focuses on data collection and post-hoc analysis, comprehensive approaches that enable real-time data validation and continuous monitoring of IIoT systems are lacking. While such solutions tend to ensure data encryption and access control, they rarely allow for real-time verification of data provenance [36], [73]–[76]. Such a gap is especially apparent when we look at the changing landscape of the IIoT and the necessity for real-time decision making based on reliable data. The absence of such real-time mechanisms renders IIoT networks susceptible to cybersecurity breaches, data manipulation, and system failures, because anomalies or malicious activities are frequently detected too late to prevent operational disruptions.

This study addresses this gap by investigating the potential integration of real-time monitoring and verification mechanisms into blockchain-based IIoT infrastructure. This might include integrating machine learning techniques to help in anomaly detection and the assurance of data integrity in an updated manner.

(b) Real-Time Constraints in Blockchain for Real-Time IIoT Security

Cybersecurity threats in Industrial Internet of Things (IIoT) environments are becoming increasingly sophisticated, posing significant challenges to industrial systems. These devices are being increasingly utilized in manufacturing facilities and other sectors, such as

power grids and the oil and gas industries. There has been a sharp rise in the number of malicious PowerShell scripts aimed at CPS. Although industries that employ cloud computing may offer robust protection against these remote code execution strategies, insider traders can intentionally create vulnerabilities to facilitate successful code execution attacks. Moreover, compromised CPS devices may be deployed in critical areas prone to accidents, posing risks to lives, property, and system stability, and potentially causing physical harm to industrial machinery. Blockchains are acknowledged as tools for enhancing the cybersecurity of industrial IoT devices, particularly when they are directly integrated with ongoing manufacturing planning, operations, and control applications [22], [46], [60]. The combination of provenance with blockchain solutions is noted for its capability to monitor blockchains [14] [16], [18]. In contrast, the real-time transaction processingof blockchains remains a major challenge, particularly for managing large volumes of data generated by IIoT devices. Owing to the latency of blockchain networks, their ability to process transactions quickly remains limited in terms of fulfilling the real-time requirements of dynamic IIoT systems, wherein instantaneous data validation and transaction processing are required.

This study aims to address latency issues by investigating methods to optimize blockchain's transaction throughput and reduce its latency, making it more suitable for time-sensitive IIoT applications without compromising the integrity or security of the data.

(c) Limited Exploration of Blockchain for Preventing Cybersecurity Threats in Real-Time

The sophistication of cyberattacks on IIoT systems has increased and the potential of blockchains against cyberthreats to IIoT systems has not yet been fully considered. Although various studies assert that blockchain technology has implications for data immutability and secure transactions, a significant limitation in this context is that predicting cyber breaches when integrated with blockchain technology remains an area that has not been thoroughly explored in existing literature. An integrated approach that combines predictive anomaly detection with blockchain's immutable ledger has the potentialto improve both cybersecurity and operational performance. Numerous researchers have explored the Blockchain Technology to secure IIoT systems. However, there remains a significant lack of integration between blockchain and predictive auditing technologies

that emphasize real-time anomaly detection. Although the use of blockchain is significant in IIoT environments as it provides support fordata integrity, immutability, and provenance tracking, it has not been adequately integrated with the predictive capabilities of machine learning and anomaly detection to improve real-time security monitoring and operational performance. None of the existing studies has explored the synergyof these dominant technologies in an addressable domain, which is IIoT security.

This study bridges this gap with an integrated blockchain model that utilizes its immutable ledger in combination with predictive auditing real-time anomaly detection by providing a more robust and proactive security mechanismfor industrial IoT systems in cloud manufacturing environments.

(d) Limited Application of Blockchain for Comprehensive Traceability in IIoT Systems

Existing studies on IIoT via blockchain mainly focus on the provenance of data or authentication of the device and ignore the detailed traceability within the system and network activities of the device itself. There have been multiple studies on the use of blockchain as a means to improve both security and data provenance in Industrial Internet of Things (IIoT) environments. This enables it as an excellent candidate for overcoming IIoT data management and security issues [70], [22] due to the benefits of security, decentralization,and immutability provided by the underlying blockchain technology[73],[22], [73]. Although existing research has made significant strides in leveraging blockchain for data provenance and authentication in the IIoT, there is a gap in addressing the comprehensive traceability of device interactions. Existing IIoT security and data integrity studies lack holistic tracking of an IIoT ecosystem as a whole, which wouldbetter facilitate an understanding of the system performance and security.

The full-spectrum traceability model proposed in this study serves as an integral framework for extending the existing knowledge. Through the application of blockchain technology, the proposed research allows end-to-end tracking of device interactions, system state changes, and operational anomalies to increase accountability, transparency, and auditability incloud manufacturing systems.

Table 1: Comparative Summary of Key Blockchain-IIoT Studies and Research Advancements

Thematic Group	Representative Studies	Key Contributions	Limitations	How this study Overcomes the limitations
Security & Access Control in IIoT	Latif et al. (2021), Yang et al. (2022), Wang et al. (2023), Cui et al. (2022)	Secure data access, cross-domain authentication, blockchain-based access regulation	Focus on static authentication, no continuous monitoring or traceability of device behavior	Integrates real-time AI-driven monitoring and behavioral profiling for dynamic authentication and continuous access control using smart contracts
Consensus Optimization & Blockchain Scalability	Huang et al. (2019), Liu et al. (2019) Yu et al. (2020),	Lightweight or sharded consensus models, improved latency and scalability	Focused on network efficiency, but lack end-to-end data validation or real-time auditing	Combines blockchain with real-time predictive auditing to balance efficiency and trustworthiness in IIoT environments
Data Provenance & Traceability	Cao et al. (2019), Juma et al. (2023)	Smart contract-based traceability and transaction monitoring	Lack granularity in tracking IIoT device actions and interactions beyond static product lifecycle events.	Implements continuous data provenance and behavioral traceability across IIoT devices using blockchain logs and AI validation
AI/ML Integration for Cybersecurity	Abdullahi et al. (2022), Folorunso et al. (2024), Lloyd & Abubakkar (2024), Messinis et al. (2024)	AI-based intrusion detection, anomaly detection using ML/DL	Lacks blockchain integration; prone to false positives; little focus on transparency and accountability	Merges AI auditing with blockchain immutability, enabling transparent, explainable, and tamper-proof threat detection via smart contracts
Trust & Resilience in IIoT Ecosystems	Umran et al. (2021), Sani et al. (2019), Cai et al. (2023), Feng et al. (2023)	Low-latency private blockchains, and resilience mechanisms	Lacks integration of AI with blockchain for adaptive resilience; focus on infrastructure not threat response	Builds a hybrid AI–Blockchain framework that supports threat detection, response, logging, and self-healing using smart contracts and predictive models

2.6 Theoretical Framework

Cyber-Physical Systems (CPS) form the foundation of Industry 4.0, by integrating physical industrial processes [1]–[8]. These systems facilitate real-time data exchange between manufacturing and logistics control engineering systems. Controlled machines, equipment, and robots can occur over the internet, thereby expanding the operational domain

from a few manufacturing plants to larger geographies. However, this new design is exposed to risks because of "lack of physical touch. Controlling sophisticated critical systems remotely builds attack surfaces on the internet coverage of digitalized machines, equipment, and robots, causing cybersecurity breaches in industrial systems [26]–[27]. Existing security measures, such as encryption and traditional access control models, provide initial protection, but lack continuous monitoring. Real-time security auditing post-deployment includes Validation, Tracking, Traceability, Sensor data fidelity, accountability and liability, inter-cloud security, algorithmic transparency, errors, and malicious processing. This results in undetected breaches, unauthorized access, and threats to data integrity, ultimately compromising the manufacturing processes. To address these challenges, the proposed framework integrates provenance blockchain with AI-driven predictive auditing for real-time monitoring.

The proposed framework is designed based on three fundamental security paradigms. These were chosen to address the security, trust, and real-time monitoring challenges faced by Cloud Manufacturing (C-MFG) environments.

- ◇ **Access Control Theories** (Zero Trust, Role and Attribute Based Access Control) → To prevent unauthorized access.
- ◇ **Blockchain Security and Data Provenance** → Provide IIoT data integrity, traceability, and immutability.
- ◇ **ML-driven Predictive Auditing** → Detecting security threats in real time.

This study proposes a well-defined security framework that employs the immutability of blockchain, verification methods of Zero Trust, and real-time adaptability of ML-Driven Predictive Auditing to build an IIoT security framework with resilience. This section presents the theoretical foundation of the proposed framework by integrating the principles of cybersecurity models, access-control mechanisms, blockchain security, and real-time audit theories.

2.7 Security Models and Access Control Theories

2.7.1 Zero Trust Architecture (ZTA)

The Zero Trust Model (ZTM) redefines traditional security models, which assume that entities residing within the perimeter of a network are trusted by default. Traditional network-based security models operate under an implicit trust assumption in which registered and authenticated entities are considered trustworthy. However, this perception of the devices will not persist, even if they adhere to all procedures and key exchanges necessary for proper registration. This approach is susceptible to insider threats and unauthorized access. ZTA breaks under the principle of "Never trust, always verify," where continuous authentication and security monitoring are required for every device before access is granted to any resource. This transforms authentication, authorization, and accounting (AAA) controls from static network-based perimeters to additional capabilities for tracking and tracing movements, locations, and usage of devices, anti-counterfeit controls, and data quality controls. The diagram presented by DiMase et al. [28], redrawn and presented in Figure 6, is an example of a traditional cyber-physical security framework deployed to secure machine-to-machine communication.

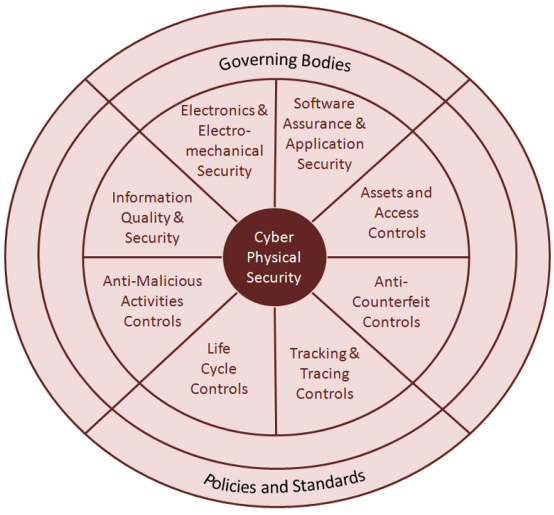


Figure 6: Redrawn cyber physical security principle (based on [28]).

The zero-trust model builds on multiple foundational security theories and frameworks that contribute to its application in IIoT security.

(a) Least Privilege Access Control (LPAC) Theory: The principle of least privilege states that individuals, software, and devices should only receive the lowest level of access required to perform their tasks, thus limiting the potential damage from security incidents [91]. Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) are widely used access control frameworks for implementing the least privileged principles in the Industrial Internet of Things (IIoT).

(b) Continuous Authentication and Risk-Based Access Models: ZTA requires continuous authentication of IIoT devices instead of one-time authentication during the first connection; thus, trust cannot be assumed through initial verification [92]. Continuous authentication is a key aspect of the ZTA when applied to IIoT environments. Meng et al.[93] proposed a zero-trust compliant protocol that eliminated the need for a trusted authority. These models evaluate different factors to determine the risk level of a user's login attempt and adjust theauthentication process [93].

(c) Zero Trust Micro-Segmentation: Microsegmentation creates security zones in IIoT networksto help stop lateral movements of attackers. All IIoT networks are divided into individual security zones in which data are protected to prevent the lateral movement of attackers in the network. This method is consistent with the defense-in-depth (DiD) approach, offering many layers of protection, which is critical for addressing the diversity of security threats inIIoT environments [94].

2.7.2 Application of ZTA in the study

ZTA limits access toIIoT devices and cloud resources for authenticated and continuously monitored entities. This ensures that only authorized users haveaccess, and minimizes the chances of data breaches in cloud manufacturing environments. Therefore, the following important security mechanisms must be implemented in IIoT environments to operationalize zero-trust principles:

(a) Continuous Identity Verification

Every IIoT device and user must continuously verify their identity using Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC). The proposed framework employs RBAC within the cloud processing and blockchain security layers,

ensuring that only authorized personnel can interact with IIoT data, configure devices, or modify the security policies. Blockchain-based decentralized identity management systems further enhance trustworthiness by preventing unauthorized identity spoofing.

(b) Dynamic Access Control and Risk-Based Authentication

Although RBAC enforces role-based permissions, it lacks contextual awareness. Attribute-Based Access Control (ABAC) extends RBAC by incorporating dynamic security attributes, such as device security posture, operational conditions, and user behavior [92]. The proposed framework leverages ABAC to make real-time access decisions for IIoT devices, thereby ensuring a granular security control based on the current security context. Access permissions are assigned dynamically based on real-time risk assessments to ensure that devices with higher risk scores face stricter authentication challenges. An access request is granted only if the authentication and security evaluation function, A(u,r,c), satisfies a predefined security threshold.

$$A(u, r, c) \geq T$$

In this equation,$A(u, r, c)$ represents authentication and security evaluation function, and u denotes the user or device requesting access, c is the contextual parameters and T represents the predefined security threshold. This ensures that the access is dynamically assessed based on real-time security metrics, thereby mitigating unauthorized access risks. Context-aware authentication mechanisms use ML-driven analytics to evaluate behavioral patterns and flag anomalies.

(c) Reduce attack surfaces

The concept of dividing IIoTnetworks into disconnected subnets prevents security incidents from cascading across an entire network. This fits the clustering architectureintroduced by Boudagdigue et al. [95], in which each cluster head controls the security of the member nodes. This segmentation can also protect essential systems and sensitive data by preventing attackers from browsing the network easily [95]. Zero-trust policies, however, function on the foundation of "never trust, alwaysverify," meaning that every access to the network has to be authenticated and authorized on an ongoing basis. Zero-trust security policies strictly enforce restrictionson unauthorized lateral movements between the security zones. This method supplements network segmentation by providing an ad-

ditional layer of security both in and betweennetwork segments. [96]. This means that if one segment is compromised, the damage is limited and zero-trust policies imply that accesswithin each segment is meticulously regulated and logged. This multilevel approach addresses the challenges posed by the extendedattack surface and the growing volume of cyber-attacks targeting industrial equipment, as pointed out by Mugarza et al. [97].

2.8 Blockchain and Data Integrity Principles

Blockchain ensures data integrity by maintaining an immutable ledger where each block is linked to the previous block using cryptographic hashing.

$$H(B_n) = H(B_(n-1)) + H(T_n)$$

Where,$H(B_n)$ denotes hash of block n, $H(B_(n-1)$ is the hash of the previous block and $H(T_n)$ represents the hash of the transaction data in the current block. This formulation ensures that each block's hash is dependent on both the hash of the previous block and the transaction data of the current block, thereby maintaining the integrity of the blockchain. This guarantees that any attempt to modify past records is immediately detectable.

Unlike conventional transactional records, these blocks are not kept on centralized ERP servers, which can be subjected to unauthorized alterations. Smart contracts are used to incorporate transactional members, establish guidelines for interactions, and verify them [2]. Smart contracts record updates for all transactions based on the transaction rules specified in the contract. These transactions will only be included in the smart contract execution records if blockchain peers provide all the transaction-specific information required by network members in the blockchain. Such rules ensure that transactions are both validated and trustworthy.

2.8.1 Application of Blockchain in the study

The proposed framework leverages blockchain to achieve the following:

(a) Smart Contract-Based Security Policies

Smart contract state rules automatically enforce security-compliance verification policies.

$V(o) \in P$. In this expression, $V(o)$ evaluates whether an operation o is valid,(o) denotes operation being performed and p is a set of permitted operations. This prevents unauthorized actions on IIoT devices and ensures compliance with the security policies. Smart contracts serve as automated security enforcers.

- Validating IIoT device interactions in real time, preventing unauthorized state changes.
- Executing risk-aware security responses, where anomaly severity determines the level of enforcement.
- Triggering compliance audits when abnormal device behavior is detected.

By dynamically adapting security enforcement policies based on an ML-driven risk assessment, the framework enhances IIoT resilience against cyber threats.

(b) Immutable Logging of Device State Changes Existing blockchain-based IIoT frameworks store every device state update directly in the chain, leading to high storage overhead and processing latency. To overcome this problem, the proposed framework employs the following steps:

- Hybrid on-chain/off-chain logging, where high-frequency device interactions are temporarily stored off-chain and securely hashed into periodic blockchain entries.
- Risk-adaptive consunsus mechanisms, ensuring that high-risk anomalies trigger immediate interventions, reducing unnecessary overhead.

This structured approach ensures that IIoT security auditing remains scalable, without compromising provenance and traceability.

2.9 ML-Driven Predictive Auditing

Although the blockchain enhances data integrity, it does not provide real-time anomaly detection. In cloud manufacturing, blockchains can be used to form trusted networks of partners that operate their assets to serve common demands and orders [54], [59], [62]. The assets can be shared and allocated to fulfill demands and orders through peer

organizations of the blockchain engaged in executing smart contracts. As the assets are privately held by blockchain peer organizations, the accountability of their reliability and operational success is based on them as per smart contracts. A blockchain state-tracking system can be used to collect data on the actions performed by assets that contribute to state changes in smart contracts.

To contribute to smart contracts in blockchains, IIoT devices should be authorized and authenticated by the peer organization's security system. While organizations attempt to authorize and authenticate each IIoT device installed in their industrial systems, there is a recognized risk of devices being replaced or existing devices being compromised for rogue purposes. In cloud manufacturing architecture, thousands of IIoT devices may be incorporated into the multiparty collaboration of manufacturers [6]–[8]. The modern conglomerate of several manufacturers contributing to large focal organizations in a cloud manufacturing setting may face several security threats caused by rogue IIoT devices [23]–[25].

At the fundamental level, the fact that needs to be appreciated is the limited influence of blockchain security on IIoT devices because the devices operate outside the blockchain network, and the blockchain smart contracts are dependent upon the sensory data collected from the IIoT devices to validate the completion of contractual terms.

The challenge of ensuring data authenticity in Industrial Internet of Things (IIoT) devices is addressed by integrating predictive capabilities into the provenance blockchain. This integration allows artificial intelligence to continuously monitor the behavior of these devices, thereby enabling real-time detection of violations. The incorporation of artificial intelligence into a provenance blockchain is essential to achieve this functionality. Predictive analytics plays a crucial role in overseeing IoT devices that have been authenticated and incorporated into cloud-based manufacturing systems. These techniques are instrumental in identifying the emergence of vulnerabilities in IoT devices, the manipulation of sensors, the transmission of erroneous sensory data, and improper utilization of assets associated with IoT data transmitters.

2.9.1 Application of Predictive Auditing in the study

The proposed framework implements predictive auditing for the IIoT security.

(a) Continuously monitor system behavior and detect anomalies in real time

ML models trained on IIoT behavioral patterns are leveraged to improve threat classification accuracy. Anomaly detection in Industrial Internet of Things (IIoT) security can be conceptualized as a time-series classification problem utilizing machine learning techniques to continuously monitor device activities in real time. This approach facilitates the early identification of noncompliant or potentially compromised IIoT devices, thereby mitigating the risks before any substantial harm can occur.

$$D = \begin{cases} 0, & \text{if } P(0 \mid N) < T \\ 1, & \text{if } P(0 \mid N) \geq T \end{cases}$$

Where D is the anomaly detection decision, $P(0 \mid N)$ represents the probability of an observation given a normal behavior model O and T is a detection threshold. This method enables the real-time identification of unauthorized access attempts.

(b) Risk Score Assessment

Predictive auditing models continuously monitor compliance with operating assets defined by smart contracts and adjust security policies in response to emerging risks. Every variable related to IIoT devices allocated to smart contracts can be monitored and assigned a risk score (low, medium, or high). A high R-value triggers security alerts, allowing for a rapid response to potential threats.

$$R = \sum_{i=1}^{n} w_i E_i$$

In this equation, R represents the risk score, E_i is the security log events and w_i denotes the weight assigned to each log type. The parameter n is the total number of different log types during the observation period. Adapting security policies dynamically based on continuous machine learning feedback.

2.10 Conceptual Framework

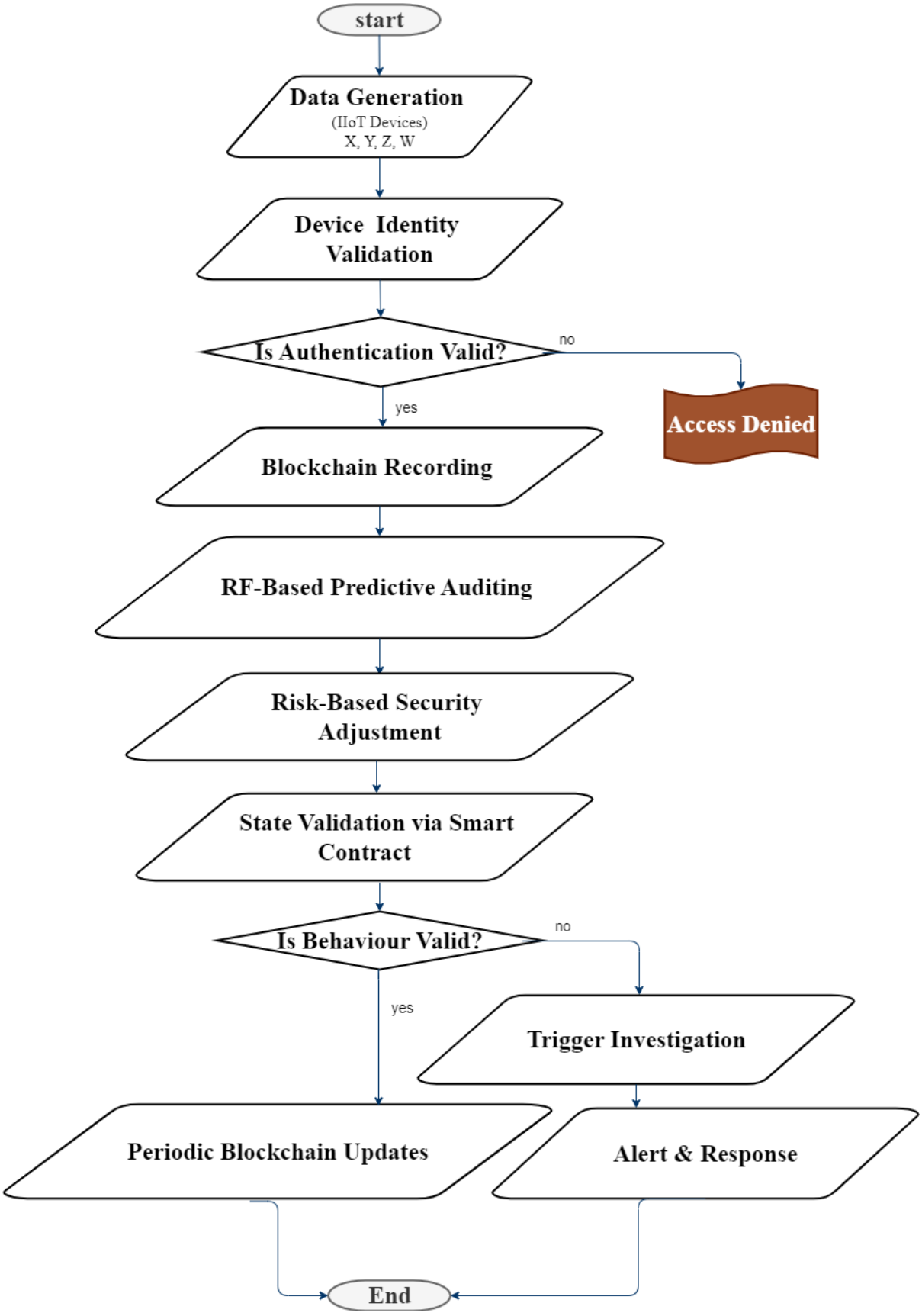


Figure 7: Theoretical Framework.

The conceptual framework for integrated Provenance Blockchain with ML-Based Predictive Auditing Framework

- **Zero Trust, RBAC and ABAC** → Secure IIoT authentication and access management
- **Blockchain** → guarantees secure data provenance and integrity.
- **Smart Contracts** → Automate the enforcement of security policies.
- **ML Powered Predictive Audit** → Offers continuous security monitoring.

The steps in Figure 7 illustrate the operation of the conceptual framework in a real-time IIoT security context. The framework workflow begins with Data Generation, where IIoT-enabled CPS devices generate real-time operational data. Device identity validation and authentication occur using zero-trust mechanisms through key exchanges with client systems and their validation using authorization records stored in the provenance blockchain. Once validated, the data underwent Blockchain Recording to ensure that the state changed in an immutable ledger. ML-driven predictive auditing continuously analyzes IIoT data for anomalies of state data transmitted by IoT devices to track their operations against the provenance rules defined in the blockchains. The Risk-based security models track the risks and adjust access control policies dynamically based on evolving threats Following anomaly detection, State Validation is conducted using smart contracts that enforce predefined operational constraints. The blockchain peer assigned to monitor the device periodically records IIoT sensor data and risk logs, and compares them against historical transaction data. If discrepancies or violations are detected, such as a device exceeding the boundary constraints, malfunctioning, or being subjected to unauthorized modifications, an investigation is triggered. Finally, Alert and Response mechanisms ensure proactive risk management. If abnormal patterns emerge, such as a device exceeding operational thresholds or behaving erratically, blockchain peer monitoring of the device flags the investigation.

To integrate Provenance Blockchain with ML-Based Predictive Auditing with the aim of building a practically viable solution, the blockchain framework is reviewed in the next section.

2.10.1 Blockchain Frameworks

Three blockchain frameworks–Hyperledger, Corda, and Ethereum–were evaluated for the proposed framework. Ethereum was avoided after the initial analysis. Ethereum is available to developers as a client machine called Ethereum Virtual Machine (EVM), which needs to be connected with the global development network of Ethereum such that EVM states can be agreed upon by all participants globally [103]. Currently, Ethereum supports "Eth" cryptocurrencies. Developers need to invest in Eth and then pay it for fulfilling requests for execution of their codes. Smart contracts can be written in Solidity and Vyper, which are Ethereum-specific languages [104]. In addition, several open-source-free tools are available for creating distributed applications [105]. The proposed framework can use one of these tools to create a provenance application, but only with the permission of the development community and pay their fees in the "Eth" currency. Ethereum is not suitable for cross-industry blockchain research in applications other than crypto-currencies, because it is tightly controlled by its development community. Its development for new industrial applications can only be performed in collaboration with other Ethereum developers who are connected to their development networks. Owing to these restrictions, the Ethereum blockchain framework is considered unsuitable for this study. This may be suitable for professional developers to collaborate and work in a highly controlled networking environment based on their mutual agreement. The other two frameworks, Hyperledger and Corda, were selected for further analyses. Unlike Ethereum, they can be viewed as generic open-blockchain frameworks that can be implemented across multiple industries for multiple applications [106], [107]. Similar to Ethereum, cryptocurrencies are supported. The advantage of Hyperledger and Corda is that they can be deployed in an isolated development environment controlled by a developer. This is because unlike Ethereum, they have native permission systems to implement and control privacy locally. Neither Hyperledger nor Corda allow unknown identities to connect and transact in the network. At the fundamental level, blockchain networks established using Hyperledger and Corda are not publicly available. They are semi-private, established with the mutual consent of the agreeing parties to form a closed group. The parties contributing to the semi-private blockchain network are peers who own the chain codes and policies governing transactions over secured channels. Secured channels constitute a fundamental fabric

of blockchain networks. Peers use chain codes as interfacing channels for asset exchange. Assets can be imagined as having business and financial value that one party can offer to another party. For example, a basket of apples sold is an asset exchange that changes hands, and the payment made for it is also an asset exchange that changes hands. All such exchanges are recorded in a ledger that is visible to all transacting parties. These exchanges are agreed on through smart contracts. The transactions recorded in the ledger are immutable and private because they are recorded in encrypted blocks that are recognized by hash functions. The transactions were verified using consensus algorithms. Subsequently, the detailed design requirements of the Hyperledger and Corda were studied [108]–[113]. The design and implementation scope include writing the contract, its states, its flows, and its integration tests. The states need to be contract state definitions to be attached to a smart contract, which, in turn, is created by declaring it a separate class [114], [115]. Opportunities to include provenance verification in the blockchain include contract states and contract verification methods. Another opportunity to add provenance verification is to add it as a flow definition and attach it to a smart contract [116]. These flows are actual transactions executed on smart contracts. Event recordings are also considered as flows from an application that captures them. After completing the contracts, flows, and states in the Corda framework, integration tests can be conducted to evaluate the features implemented in a smart contract [116]. Integration tests help test the full integrity of the flows and states declared, with their Smart Contracts identified as distinct classes. It should be noted that injecting provenance verifications as a flow definition, contract state, or contract verification method requires the generation of verification data from outside the blockchain. This is where data collection from the IIoT and inspection using a machine-learning algorithm can be designed and implemented. Understanding the overall bigger picture requires a review of the blockchain architecture explained by [117] and [119], supported by Hyperledger and Corda, as reviewed in the next subsection.

2.10.2 Blockchain Network Architecture

When manufacturers integrate their cyber-physical systems, which are essential for their production processes, into cloud manufacturing, they can connect their Enterprise Resource Planning (ERP) systems via blockchain [22]. Contemporary ERP systems have effectively interfaced with leading blockchain technologies such as Hyperledger and Ethereum. This integration further extends to linking Industrial Internet of Things (IIoT) devices and their associated cyber-physical systems to the blockchain [54], [73]. Figure 8 illustrates a design scenario in which a manufacturing network is managed using the blockchain technology. This configuration, detailed in reference [117], employs Hyperledger Fabric to develop, implement, and manage smart contracts. This design demonstrates how provenance data streams and artificial intelligence can be integrated into a blockchain.

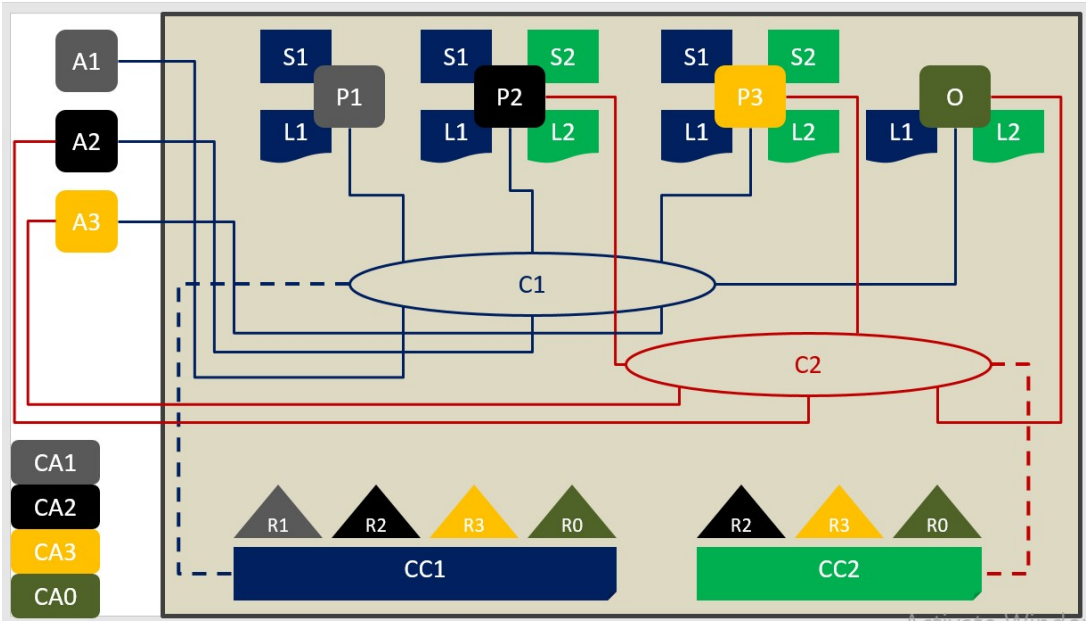


Figure 8: Blockchain Network Architecture.

The structure shown in Figure 8 was assessed based on the scenario outlined in Ref.[117]. In this context, four entities, R0, R1, R2, and R3, form a blockchain network (BN) to facilitate the signing, sharing, and management of smart contracts. R0 functions as a contracting authority, whereas the remaining entities act as contracting vendors. They agreed to establish two network channels, C1 and C2, which are governed by network configurations based on policies, specifically CC1 and CC2. Channel C1 is employed by

R0, R1, R2, and R3, whereas C2 is designated for use by R0, R2, and R3. Consequently, R1 does not participate in C2, and is thus prohibited from accessing it.

Applications A1, A2, and A3 were deployed by R1, R2, and R3, respectively, to connect to the network via the approved channels. For network interactions, R1, R2, and R3 must authorize their peers P1, P2, and P3 to operate on their behalf. R0 authorizes O to handle orders R1, R2, and R3 through peers P1, P2, and P3. Peers P1, P2, and P3 are allowed to access the BN network using cryptographic keys issued by certification authorities CA1, CA2, and CA3, respectively. The ordering authority manages C1 and C2 network channels to enable interactions with P1, P2, and P3. P1 has a restricted access to C1, whereas P2 and P3 can access both C1 and C2. When orders are placed, smart contracts are digitally signed with signatures provided by CA1, CA2, and CA3 to P1, P2, and P3 for the contracting vendors, and a digital signature from CA0 to O for the contracting authority. Furthermore, to digitally sign contracts, CA1, CA2, and CA3 issue X.509 certificates to components identified as part of organizations R1, R2, and R3, respectively. Certification authorities endorse transactions by signing them. Once signed, smart contracts are recorded in smart ledgers L1 and L2, which correspond to network segments C1 and C2. P1 maintains a copy of only L1 (as it and its organization R1 have no business connections with C2), whereas P2 and P3 hold copies of both L1 and L2. Details concerning smart contracts L1 and L2, such as the logs of tasks completed in accordance with the contract terms, are stored in state databases S1 and S2. P1 retains a copy of S1, whereas P2 and P3 have copies of S2 and S3, respectively, based on their access rights. All state database copies are synchronized. The ordering authority O does not need to maintain a copy of the state databases because R0 does not contribute to state changes. However, O can review S1 and S2 as required.

The design scenario described above demonstrates a fundamental blockchain network for creating, executing, and monitoring smart contracts and related events. This study integrated real-time provenance capture and predictive analytics using artificial intelligence, specifically by employing a random forest algorithm. The blockchain framework selected for this study was CORDA [120], which is lightweight and requires few resources. Random forests were selected because of their capability to make independent predictions about multiple variables streaming continuous datasets in a group, forming a structure

based on past records [115]. It can handle a large number of input variables to provide estimations by reading datasets on which the variables are significant. It can provide promising results, even when trained using smaller training datasets. For example, the original research by Breiman [115] used training datasets as low as 200 and 436 records to generate up to 72% accuracy. Accuracy levels can be improved significantly when structured data are used as inputs and for training. In this framework, the Random Forests algorithm is built into the cloud manufacturing controller, which has distinct rules for operations for each asset, as per the smart contracts defined in the blockchain.

Summary

This chapter provides a comprehensive review of the existing literature and establishes the theoretical framework that underpins this research. It provides deeper knowledge on the concept of provenance and predictive auditing and their usage in blockchain, forming the essential theoretical background to study real-world blockchain frameworks in the next chapter. In addition to building the theoretical knowledge needed for studying real-world blockchain frameworks, this chapter presents related studies conducted mostly at PHD levels to provide inputs for the components of the design proposed after understanding the frameworks and establishing the directions for realizing the design in an affordable experimental environment for this research. Among the related studies reviewed, few have been conducted on the design of provenance blockchain solutions. These studies were helpful for using them as baselines for designing their architectures. The original value addition from this research was proposed in the form of two added layers of machine learning and a predictive engine using the differences between the machine learning predictions and actual data streams received. All of these must be performed at the end of the blockchain, right behind the peers interfacing with the blockchain. This research was conducted as a first effort to integrate a risk management system using the prediction of provenance information by machine learning with the state transition rules engine of a smart contract loaded on a blockchain to continuously monitor the compliance of operating assets within the boundaries defined by the smart contract.

The theoretical framework synthesizes these concepts and establishes a foundation for integrating provenance blockchain with ML-driven predictive auditing. This framework was designed to mitigate unauthorized access risks, ensure real-time anomaly detection, and maintain immutable audit trials. The chapter concludes by identifying research gaps in current methodologies, including the need for improved hybrid blockchain designs by registering all operating assets through smart contracts loaded on the blockchain, and integrating predictive auditing by machine learning. It is a tightly controlled industrial cyber-security system. Further, the cyber security enhancements ensured by this system in the Industry 4.0 framework are explained in subsequent chapters.

Chapter III

Proposed Framework

3.1 Introduction

This chapter presents the methodology and proposed framework designed to address the key challenges associated with data integrity, unauthorized access, and real-time monitoring by leveraging a combination of on-chain and off-chain mechanisms. The research methodology employs Design Science Research Methodology (DSRM) to guide the design, development, and validation of the framework, ensuring that it addresses real-world problems. The proposed framework was built on a hybrid blockchain model that utilizes the Corda platform for smart contract execution and provenance tracking. This integration ensures that all device transactions are logged securely while maintaining the system efficiency through off-chain data storage.

3.2 Research Philosophy

This study adopts pragmatic learning as both inductive and deductive, which is well-suited for problem-solving in real-world contexts. Pragmatism emphasizes practical applications and focuses on developing solutions that are both theoretically sound and empirically validated [118], [119]. This is consistent with the principles of Design Science Research Methodology (DSRM), which focuses on developing and assessing artifacts to tackle identified issues by refining a conceptualized design of provenance within an ex-

isting blockchain framework and incorporating AI for monitoring and decision-making. A machine learning algorithm utilizing "Random Forests was designed to predict the numerical values for the operating parameters and identify the risk levels based on the boundary parameters. These risk levels were added to the event records to verify ongoing work and completions according to the smart contract terms. Consequently, operations teams that monitor these records can observe risk levels and examine the specific IIoT devices involved. Because event data might be gathered from a collection of IIoT devices, the entire group may need to be investigated until the investigators identify the problematic IIoT device.

As this is an original approach to addressing the research problem [148], it requires a repetitive cycle of development, refinement, and testing by simulating scenarios of provenance data anomalies using simulated production data in logistics processes until the final outcomes are deemed satisfactory. This study focused on the iterative process, which is a defining feature of DSR. With this methodology in mind, a pragmatic philosophy was chosen and both qualitative and quantitative methods were employed for data collection and analysis [148]-[153]. Given the focus of this study on solving operational challenges in IIoT systems, particularly in terms of security and traceability, a pragmatic approach ensures that this research contributes directly to addressing the issues faced by practitioners in cloud-manufacturing environments. The framework developed as part of this research will undergo continual testing and refinement to address the gaps in the continuous security, privacy, and trust of IIoT-enabled CPS devices working as machines, equipment, robots, carriers, vehicles, and other assets allocated to smart contracts under execution.

This study further aligns with the Design Science Research Methodology (DSRM), which emphasizes the creation and evaluation of artifacts that offer practical solutions to identified problems [150]-[152]. In this context, an artifact is the integrated blockchain and ML framework, designed to address the security, transparency, and operational continuity challenges associated with IIoT-enabled systems. The research approach is solution oriented, focusing on the design of an artifact capable of improving cloud manufacturing security.

The objectives of this study were as follows:

(a) Optimization of the provenance blockchain for continuous monitoring, traceability, and visibility in IIoT environments.

(b) Develop a framework that integrates a provenance blockchain and predictive auditing to proactively detect and mitigate security risks in IIoT-based cloud manufacturing.

(c) Evaluate the effectiveness of the integrated blockchain and predictive auditing framework in addressing security, traceability, and transparency challenges while maintaining operational continuity.

3.3 Methods

This study employs mixed research in which qualitative and quantitative methods are combined to form a hybrid method [118], [119]. The mixed-method design allows for a comprehensive investigation that combines artifact development (qualitative) with a system performance evaluation (quantitative). The learning approaches of inductive and deductive are mixed following the qualitative and quantitative methodologies, respectively, and they were followed in the methods. The theoretical interpretations of the literature review outcomes for designing artifact development were inductive, and the system performance evaluation conducted was analyzed deductively. This study utilized the DSRM method suggested by Peffers et al. [121] and Hevner et al. [120].

3.3.1 Qualitative Methods

Artifact Design and Development: The conceptualized design and development of the blockchain and ML framework are the primary focus of qualitative research. The framework was designed on the basis of theoretical foundations and an existing body of knowledge. This requires refinement through several rounds of iterative design and refinement occurs through pilot implementation and continuous evaluation. The artifact design process follows several iterations and refinements to achieve artifact effectiveness [122]. As illustrated in Figure 9, the DSRM process model [121] comprises six activities spanning the entire research process from initial motivation to final communication. The Research Entry Point varied according to the specific study [121]. Research can be

initiated through four primary approaches: commencing with a specific problem, concentrating on a particular objective, emphasizing design and development, or being driven by the needs of the client or context. In this study, we adopt a "problem-centered initiation," wherein we identify an issue within the literature to evaluate its significance and explore it throughout the research process. As illustrated in Figure 9, this approach suggests that the research process will proceed sequentially, beginning with the initial task: "identify the problem and motivate."

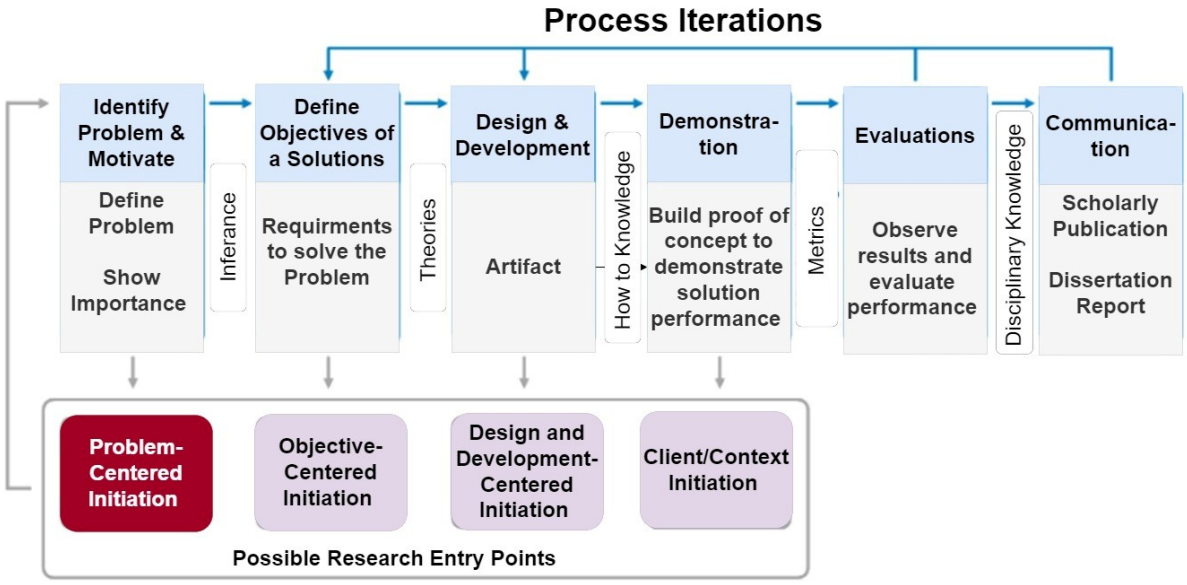


Figure 9: Design science research methodology process model [56].

3.3.2 Quantitative Methods

Experimental Evaluation: The study used an experimental evaluation method in a controlled environment. This was mainly performed during the design cycle, where the artifact was assessed and refined using the DSR guidelines, which cover the design, testing, and iterative refinement stages. A quantitative method is used to evaluate the effectiveness of the integrated framework. Quantitative data were generated by simulating intermittent anomalous data values of parameter variations at different levels injected into the data streams to best represent the performance of the artifact. The effectiveness of the system was evaluated using key performance indicators such as prediction accuracy, blockchain-logging latency, risk detection rates, and response times.

3.4 Data Collection

The primary data for this study was generated from a simulated IIoT environment. IIoT devices (representing real-world industrial machines and sensors) will be simulated to generate data streams that will be monitored and logged by the integrated blockchain and ML-based predictive auditing system [123], [124].

3.4.1 Data Sources

This study relied on data generated from a simulated IIoT manufacturing environment to ensure controlled evaluation of the proposed framework [125], [126]. **The primary data sources included the following**:

(a) **IIoT Device Data**: Data generated by IIoT devices, including sensor readings (Locations and weight) were collected to serve as inputs for the blockchain and ML auditing system.

(b) **Blockchain Data**: Provenance data, including transaction records and logs of state changes, were collected from the blockchain.

(c) **ML-based Predictive Auditing Data**: Predictive models generate data on the probability of system risks, including security breaches and system anomalies.

3.4.2 Data Collection Process

- **Real-Time Data Collection**: The system continuously collects data to simulate the real-time operation of the IIoT devices. Data are generated and fed into both the blockchain and ML auditing systems.

- **Simulated Data**: The data will be simulated using Python scripts of IIoT devices, allowing controlled testing of the integrated system under various operational conditions. These Conditions are defined in the smart contracts.

3.4.3 Experimental Setup

The experimental setup consisted of simulating IIoT devices in a cloud manufacturing environment using two primary components [127].

Provenance Blockchain: A fully operational runtime of the Corda framework was set up to record and validate the state changes of IIoT devices, providing an immutable record of device activities. The smart ledger blockchain invokes investigations by peers accessing the smart contract to automate the validation and logging of the state transitions in the blockchain.

ML-Based Predictive Auditing: Machine learning model was conceptualized using random forest algorithms at the core to predict potential security threats and operational anomalies. The Random Forest model is deployed off-chain, analyzes the streams of data from IIoT devices, and forecasts potential risks. While Random Forest (RF) is inherently a batch learning algorithm, its integration within real-time Industrial Internet of Things (IIoT) systems poses challenges related to responsiveness, data drift, and dynamic behavioral patterns. To overcome these limitations and meet the operational demands of cloud manufacturing environments, this study implemented an adaptive RF strategy based on the principles introduced in our previous work [26], which proposed a hybrid ML and provenance blockchain framework for risk mitigation. This approach comprises two interdependent layers: (i) sliding window accumulation and retraining, (ii) shadow model architecture.

(i) Sliding Window Accumulation and Retraining

- Rather than retraining the model after each individual data point, the system uses a sliding time window to accumulate and audit data, specifically spatial coordinates (x, y, z) and load weights w from IIoT-enabled forklifts. A new batch is formed after a predefined number of events (1,000 interactions). These data were then labeled and used to update the training dataset.

- Retraining is either scheduled at new data thresholds or triggered when the model performance metrics drop below a threshold. This approach ensures a balance between model responsiveness and computational stability, a technique that is widely

applied in time-series adaptation models [173].

(ii) Shadow Training Architecture and Adaptive Updating The shadow model architecture runs concurrently with the production pipeline, allowing for asynchronous and non-disruptive model updates. This component continuously aggregates new labeled behavior data derived from smart contract validation and predictive audit outcomes.The retraining of the RF model was triggered based on performance degradation or new data thresholds:

- Scheduled retraining: Every 2-4 simulated hours or after 10,000 new labeled records
- Adaptive retraining trigger: F1-score falling below 0.85 on recent test batches

The updated model is validated offline and only hot-swapped into production if it outperforms existing models. This methodology follows common MLOps practices, where shadow models ensure robust deployment cycles [174].

Integration of Components: While the ML model conducts instantaneous evaluations of potential risks, the rules governing blockchain contracts are configured as predictive audits, permitting updates solely when the ML's risk predictions align with established parameters. This integration enables continuous monitoring, risk mitigation, and real-time audit-logging.

3.5 Data analysis

The data analysis focuses on evaluating the effectiveness of the integrated framework in terms of security, traceability, and operational continuity. The performance of anomaly detection by machine-learning algorithms is measured using four parameters: true positives, false positives, true negatives, and false negatives [129]–[131]. A true positive (TP) anomaly exists and was successfully detected. A false positive (FP) is a nonanomaly (benign) that is erroneously detected as an anomaly. True negative (TN) is a non-anomaly (benign) that is detected as a non-anomaly (benign). A false negative (FN) is an anomaly that has been erroneously detected as non-anomaly (benign). There are two additional parameters: the total number of positives (P) and the total number of negatives (benign instances, N). These six parameters were used to generate the following four performance metrics: Overall Accuracy, Precision, Recall, and F1 score. They are defined as follows [129]–[131]:

3.5.1 Evaluation Metrics

Several evaluation metrics were used to assess the integrated framework.

- Overall Accuracy $=\frac{(TP+TN)}{(P+N)}$
- Precision $= \frac{(TP)}{(TP+FP)}$
- Recall $=\frac{(TP)}{(TP+FN)}$
- F1 Score $= 2 \times \frac{Precision \times Recall}{Precision+Recall}$

The F1 score is the harmonic mean of precision and recall. By solving the following equation, the final equation for the F1 score is as follows:

F1 Score $=\frac{2TP}{2TP+FP+FN}$

Precision, Recall, and F1 scores were generated for every test case, whereas accuracy was generated for all cases. However, there exists a method for calculating the overall F1 score [131]. The formula used was as follows:

$F_{1_o} = \frac{\sum_{i=1}^{c}(w_i f_1^i)}{\sum_{i=1}^{c} i}$

Where

- F1O is Overall F1
- w_i is the number of test cases in scenario
- $f_1^i$is the average F1 score for individual classes
- $c$is Weight (number of cases) of the ith class (number of cases)

A similar formula can be used to calculate the overall precision and recall of the model, which reflect the anomaly detection rate of the model.

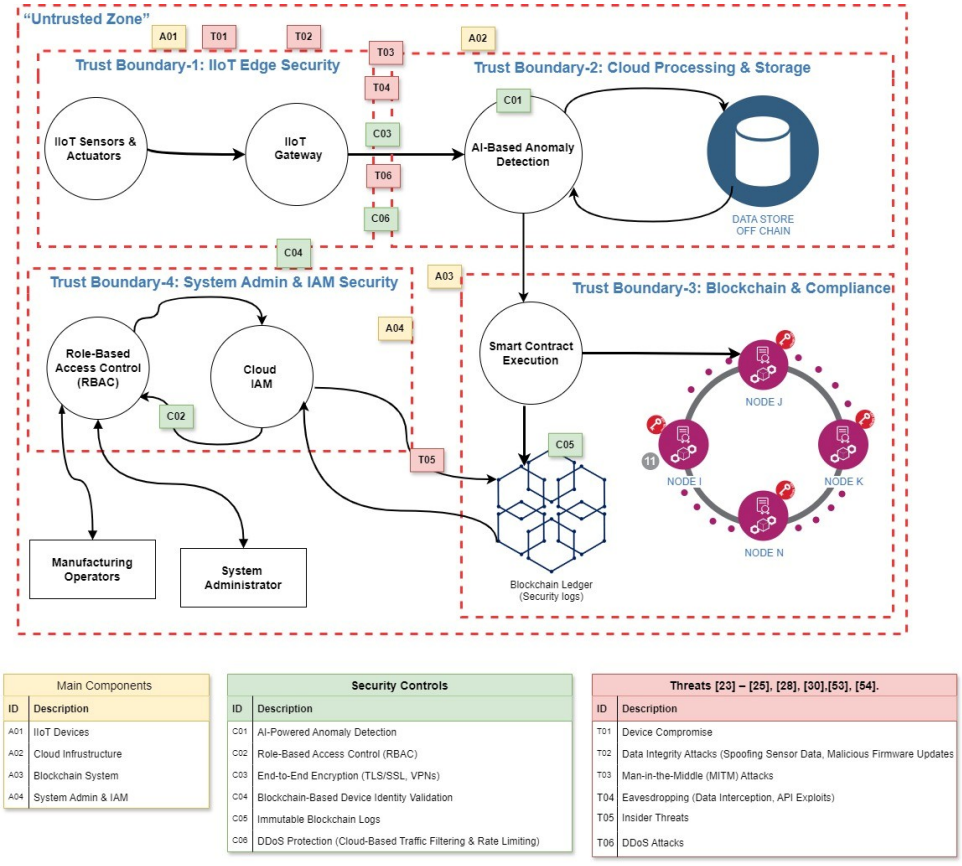
Traceability: Completeness and accuracy of provenance data captured on the blockchain.
Tracability =

$$(NumberofDeviceActionsTracked)/(TotalNumberofDeviceActions)100$$

Blockchain Latency: Time taken to log device state changes to the blockchain.

3.6 Threat Model

The threat model illustrated in Figure 10 describes the possible security threats, vulnerabilities, and attack vectors for the system. In addition, it allows the pinpointing of risk that is foundational, and allows for an understanding of threats against which the proposed framework must be defended. Appropriate mechanisms are encoded in the framework to mitigate all identified threats, thereby preventing potential adversarial actions against the system.



Main Components	
ID	Description
A01	IIoT Devices
A02	Cloud Infrastructure
A03	Blockchain System
A04	System Admin & IAM

Security Controls	
ID	Description
C01	AI-Powered Anomaly Detection
C02	Role-Based Access Control (RBAC)
C03	End-to-End Encryption (TLS/SSL, VPNs)
C04	Blockchain-Based Device Identity Validation
C05	Immutable Blockchain Logs
C06	DDoS Protection (Cloud-Based Traffic Filtering & Rate Limiting)

Threats [23] – [25], [28], [30],[53], [54].	
ID	Description
T01	Device Compromise
T02	Data Integrity Attacks (Spoofing Sensor Data, Malicious Firmware Updates
T03	Man-in-the-Middle (MITM) Attacks
T04	Eavesdropping (Data Interception, API Exploits)
T05	Insider Threats
T06	DDoS Attacks

Figure 10: Threat Model.

3.6.1 Assumptions of the Proposed Framework

The effectiveness of the framework is based on several assumptions that define the security model and the operational boundaries.

(a) IIoT Devices Are Trusted at Initial Registration: IIoT devices are assumed to be trusted at initial registration and adhere to accepted industry standards for

security, interoperability, and data integrity.

(b) Security Best Practices: It is assumed that system components (IIoT devices, cloud infrastructure, blockchain, and audit systems), as well as the communication between them, follow industry-standard security practices.

(c) Trust in Blockchain Mechanisms: A blockchain network is inherently assumed to be trusted because it operates on a definitive consensus mechanism that prevents any unauthorized individual from validating and committing transactions.

(d) Trusted Cloud Infrastructure: The cloud infrastructure used for off-chain storage and processing is assumed to be trusted, which prevents unauthorized modifications at the infrastructure level.

(e) Predictive Auditing System Is Continuously Updated: The predictive auditing system is assumed to be updated continuously with machine learningmodels, whereas predictive models can detect new types of anomalies when encountered in real time.

(f) Strong Authentication and Access Control Are Enforced: All user interactions with the system are expected to utilize strong multi-factor authentication (MFA) and role-based access control (RBAC) to safeguard against unauthorized access.

(g) Real-Time Data Availability: The system operates in a real-time context in which data collection, processing, and decision-making are performed continuously. This assumes low-latency operations for the IIoT and predictive auditing systems.

3.7 Framework Overview

The proposed framework for this research is presented in Figure 11 to implement the continuous security, privacy, and trust of IIoT-enabled CPS devices working as machines, equipment, robots, carriers, vehicles, and other assets allocated to smart contracts under execution. The framework integrates a provenance blockchain with a cloud manufacturing controller system with predictive auditing capabilities. This integrated framework combines real-time device state tracking, ML-driven predictive auditing for early detection of threats, and proactive security enforcement and optimizes blockchain storage to minimize transaction overhead while preserving traceability and data integrity.

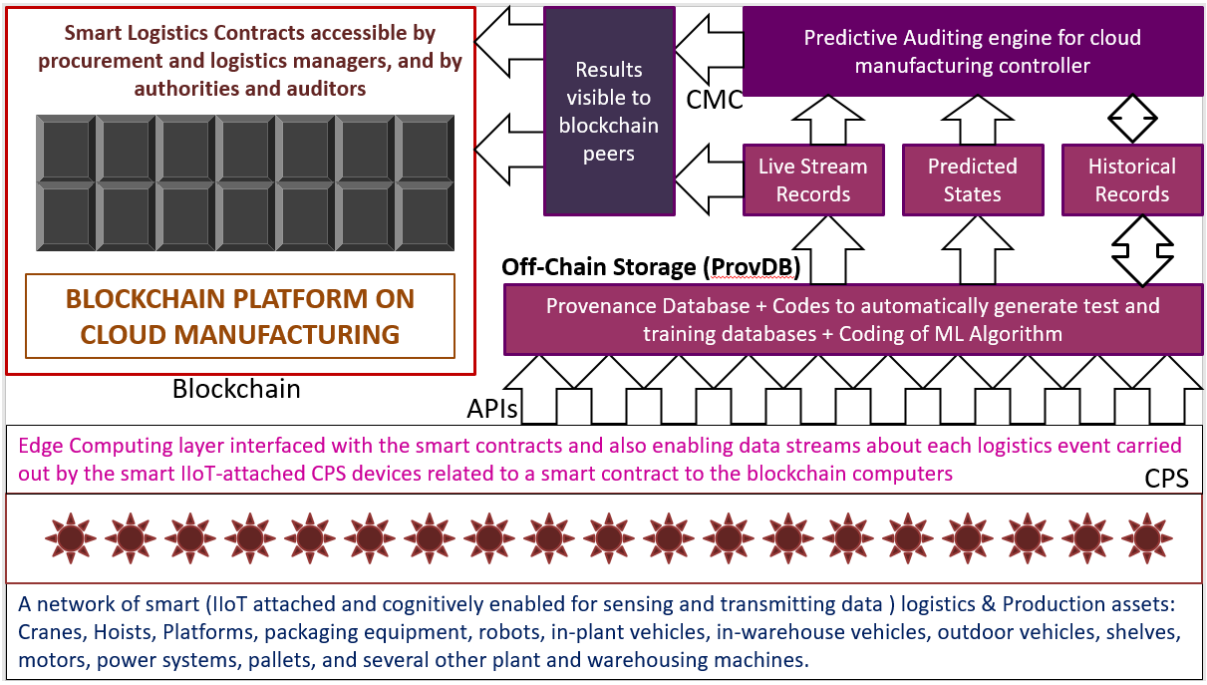


Figure 11: Proposed Framework.

The proposed framework comprises IIoT-enabled manufacturing smart logistics and production assets digitalized using IIoT attachments, such that the assets can collect data from their sensors and send them to cloud manufacturing controllers through the Internet. Examples of these assets include Cranes, Hoists, Platforms, packaging equipment, robots, in-plant vehicles, in-warehouse vehicles, outdoor vehicles, shelves, motors, power systems, pallets, and several other plants and warehousing machines. These assets must be allocated to smart contracts to execute predefined tasks to be monitored and controlled by a

cloud-manufacturing controller. CPS devices can send data directly to application Programming Interfaces (APIs) over the Internet. Off-Chain Storage (ProvDB) captures the state changes in CPS devices while executing tasks assigned by smart contracts. Instead of storing all raw data in a chain, ProvDB is used for off-chain storage, containing detailed provenance data, whereas blockchain retains hashed summaries. The machine learning models were tasked with learning from the ProvDB training database and using the test database to arrive at the predicted values of the next state and were fed to the CMC for comparison with the historical time series data. A cloud manufacturing controller (CMC) contains a predictive auditing engine for cloud manufacturing controllers based on smart contracts. The predictive auditing engine is tasked with comparing the latest state data, predicted values, and historical time-series data and classifying each transaction as low-, medium-, or high-risk, triggering automated responses for mitigation. The predictive auditing engine continuously learns from both real-time and historical data, and adapts to evolving attack patterns and system anomalies. CMC implements an interval logging mechanism, periodic hashing, and anchoring data on the blockchain to ensure its integrity. All results generated by CMC are visible to the blockchain peers for transmission into the blockchain smart contract execution blocks as status updates. A blockchain maintains an immutable ledger of device state changes and event logs. If the results are rejected by the blockchain, blockchain peers must raise alerts to conduct investigations on doubtful CPS devices.

3.8 System Architecture

The proposed system architecture presented in Figure 12 consists of five interdependent layers, each designed to handle specific tasks within the framework. These layers ensure seamless data collection, anomaly detection, blockchain-based provenance tracking, smart-contract execution, and real-time alerts for security and compliance.

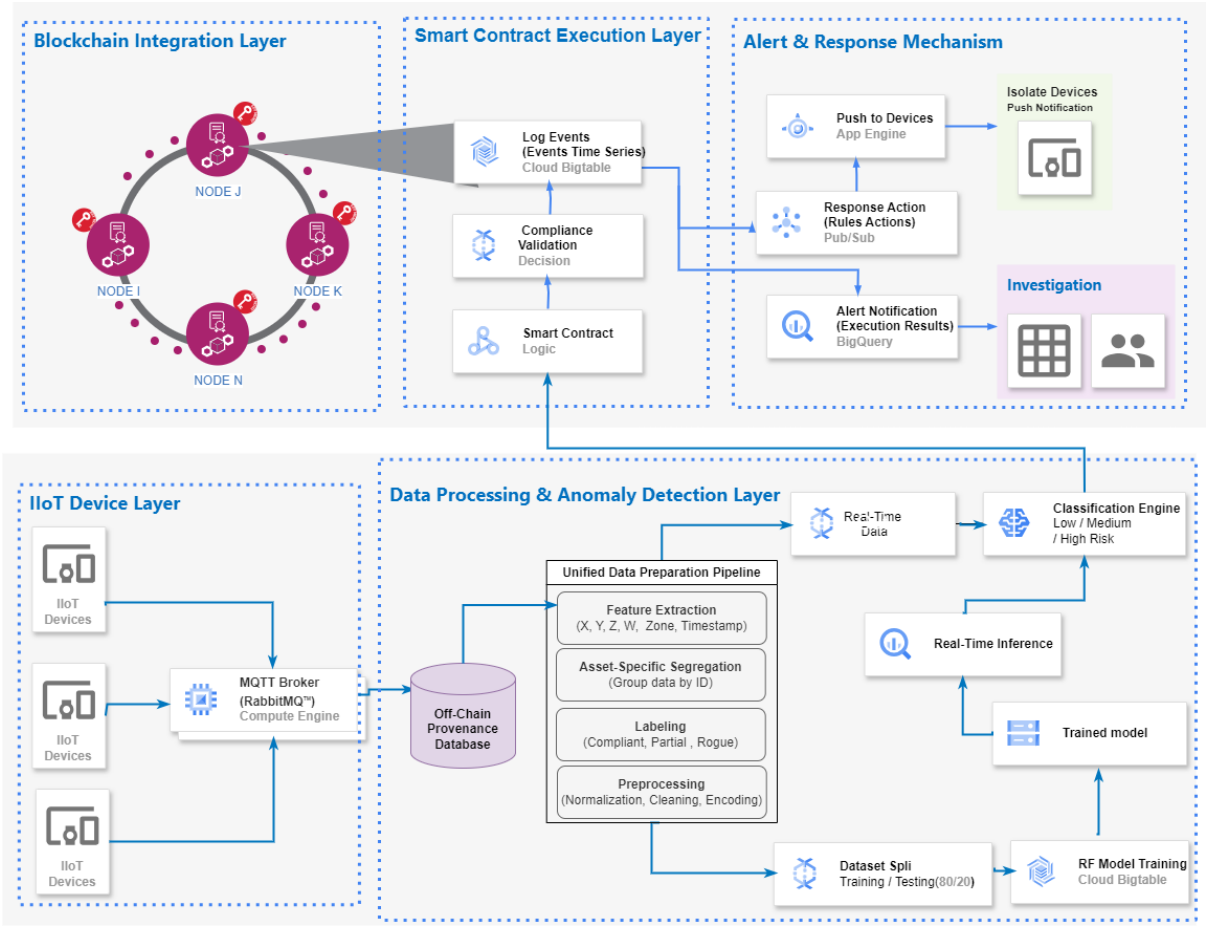


Figure 12: System Architecture.

3.8.1 IIoT Device Layer

The IIoT Device Layer serves as the foundation of the framework, where IIoT-enabled sensors continuously generate real-time data through API interfaces that are enabled on the internet for cloud manufacturing. IIoT assets must be allocated to smart contracts to execute predefined tasks to be monitored and controlled by cloud manufacturing. The MQTT broker is employed as a lightweight messaging protocol to facilitate real-time data

transmission between the IIoT and cloud manufacturing. This component addresses RQ1 by continuously logging IIoT device states off-chain

3.8.2 Data Processing and Anomaly Detection Layer

This layer applies ML models to refine, analyze, and assess incoming data for potential risks and anomalies. This layer detects anomalies and calculates the risk scores of behavioral anomalies based on signatures in the data streams collected from the processes. This component addresses RQ2 by detecting potential anomalies in IIoT device behavior using a trained Random Forest model, thereby enabling real-time anomaly detection and proactive threat mitigation. It also contributes to RQ3 by generating risk scores that inform downstream blockchain validation and consensus decisions, ensuring data integrity and operational continuity.

3.8.3 Blockchain Integration Layer

The blockchain layer ensures that device states are recorded immutably. Each device interaction was stored as a blockchain transaction. The transactions are stored in nodes in the form of shared states that are visible to participants involved in the blockchain [135]. When a state is changed, it is accepted only when the majority of nodes validate the change in the state. The transactions are grouped in the form of "chained blocks" a protected cryptographically using hash function, which not only identify the transactions but also links them with their previous transactions thus ensuring seamless integration of the sequence of transactions [136], [137]. This component addresses RQ1 by providing a secure and immutable ledger for provenance tracking, using interval-based commits to optimize for performance and scalability in IIoT-enabled Cloud Manufacturing environments. It also addresses RQ3 by ensuring data integrity and traceability through cryptographic validation and decentralized recordkeeping, reinforcing trust in manufacturing operations.

3.8.4 Smart Contract Execution Layer

The Smart Contract Execution Layer governs compliance verification and state transitions within the framework, ensuring that devices adhere to pre-established operating

boundaries. This layer operates as follows.

(a) **Smart Contract Logic:** State transitions are validated based on the computed risk score of their operational boundary breach defined in the smart contract at the time of asset registration in the blockchain. In a smart ledger, the state transitions of smart contracts are tied to risks, enabling blockchain peers to swiftly recognize and address major deviations.

(b) **State Change Decision:** If device is compliant, its state is updated on the blockchain. If any of the device's state values fall outside the allowed boundaries, the smart contract will reject the transition, preventing the update from being recorded and invoke further investigation

This layer ensures that only authorized and compliant devices can operate, thereby preventing unauthorized asset movements and operational breaches. This component addresses RQ3 by validating IIoT device transactions using programmable smart contracts that enforce compliance with expected behavioral norms. Transactions flagged as anomalous by the ML engine are rejected, ensuring that only trustworthy data is committed to the blockchain, thus maintaining system security and operational resilience.

3.8.5 Alert and Response Mechanism

When anomalies are detected, alerts are triggered and blockchain peers initiate investigations. Smart contract updates help in invoking alerts about contractual breaches, such that essential corrective and preventive actions can be taken proactively. The Alert and Response Mechanism ensures real-time monitoring and proactive security responses that trigger automated actions such as sending alerts or updating compliance logs before they fully materialize, thereby reducing the risk of damage or data loss. This component addresses RQ3 by dynamically adjusting blockchain consensus mechanisms based on the ML-generated risk level of each transaction. It accelerates validation for low-risk transactions (using PoA) while requiring enhanced scrutiny for high-risk cases, thus balancing security and performance continuity in the IIoT ecosystem

3.9 Data Flow Diagram

The data flow of the proposed framework follows a sequential process in which each layer of the system interacts to achieve secure, compliant, and transparent IIoT operation. The data flow diagram is illustrated in Figure 13, where the IIoT devices stream

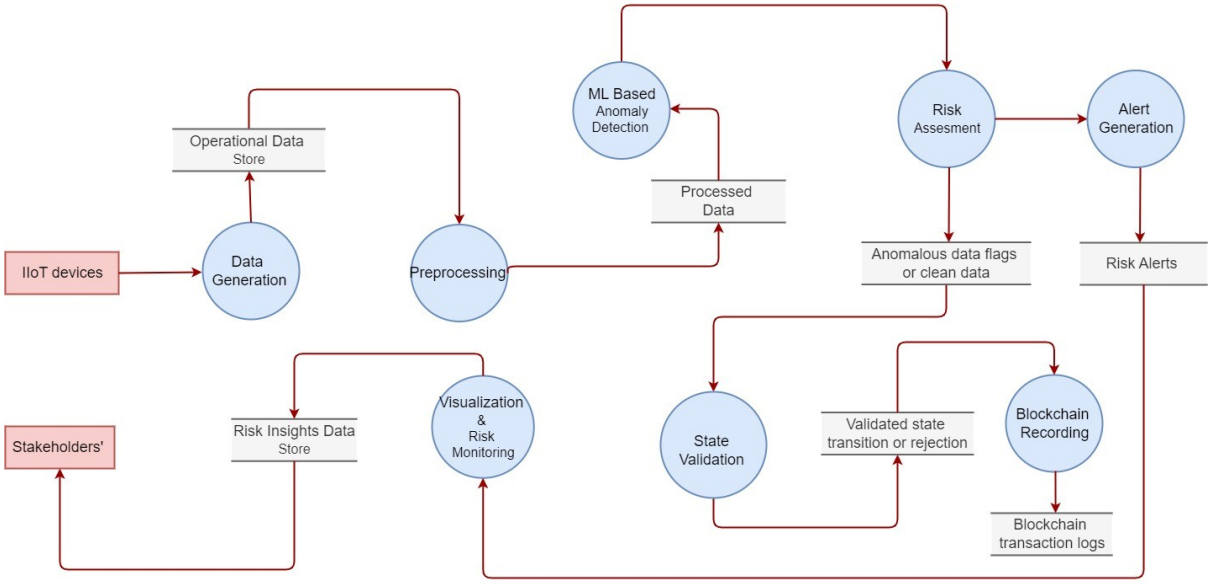


Figure 13: Data flow Diagram.

operational data related to the IIoT device provenance parameters allocated to smart contracts. The dataset was processed using an artificial intelligence system specifically designed for anomaly detection. Within this framework, machine-learning models monitor the operational parameters delineated in smart contracts and identify deviations from these parameters through predictive auditing facilitated by artificial intelligence. The anomaly detection process consists of sub-range validation, historical comparison, and risk classification. Measured values that exceed the set thresholds or differ significantly from normal behavior are given a risk score (NONE through HIGH), and the system determines whether further intervention is required. Machine learning predictions can be based on continuous learning from data streams flowing from IIoT operation fields.

Once analyzed, the provenance data are subjected to state validation through smart contracts stored in blockchains, ensuring that all transactions comply with predefined operational constraints. Smart contract state rules can be defined to enforce policies on an entire cloud-manufacturing network. The state changes in the smart ledger of smart

contracts were linked to risks so that blockchain peers could detect high deviations and take actions in a timely manner.

When anomalies or unauthorized activities are detected, the system automatically triggers alerts and response mechanisms, based on the severity of the identified risk. Any anomaly causing negative execution of smart contracts is detected through machine learning and traced to the device using provenance data. Negative execution breaches the blockchain state transition rules; hence, transactions are rejected, thereby promoting investigation. The framework ensures that all flagged anomalies are logged immutably, maintaining an auditable trial for the compliance, security, and continuous monitoring of IIoT assets within the system.

3.10 Algorithms

This section presents the main algorithms used to develop the framework, which include continuous monitoring, predictive risk detection, and secure provenance tracking.

3.10.1 Algorithm 1

This algorithm addresses the need for continuous monitoring of IIoT devices by logging off chains in real time, whereas an optimized interval-based blockchain enhances security and reduces the blockchain overhead.

Let $D = d_1, d_2, \ldots, d_n$ be a set of IIoT devices. S_d the state of device d at time t.
$P(d, t)$ be the provenance log of d at time t.
Denote the blockchain record at time t as $B(t)$

Algorithm Steps:

Algorithm 1 Continuous Provenance Logging and Interval-Based Blockchain Commit

Input: S(d) state of the IIoT device, Monitoring interval T, Blockchain B

output: Provenance log P(d,t), updated blockchain B(t)

1: **Initialize ProvDB and Blockchain**

P(d,t) ← Initialize ProvDB

B(t) ← blockchain ledger

2: **State Retrieval and Off-Chain Logging**

Retrieve the real-time state S(d) of each device d in D and store it in ProvDB

3: **Calculate cryptographic hash H(P(d,t)) of the provenance log after a predefined monitoring interval T,which is the threshold time interval.**

Append H(P(d,t)) to the Corda blockchain B(t+1)

Clear temporary logs after committing to blockchain.

4: **Repeat Process**

Logging and committing system status changes at regular intervals.

3.10.2 Algorithm 2

This algorithm leverages machine learning (ML) models to predict real-time potential anomalies using the behavior of IIoT devices. The algorithm sends alerts as soon as an anomaly is detected, and records the event on the blockchain for later reference, thereby establishing a proactive approach to security.

$X = [x_1, x_2, \ldots x_3]$ is the input feature set extracted from the IIoT device behavior.
$Y = [0, 1]$ be anomalous status, Y=0 normal; Y=1 anomalous.
f(X)→Y is a random Forest based predictive function.
$P(Y = 1X) \in [0, 1]$ is the probability of occurrence of an anomaly

Algorithm Steps:

Algorithm 2 ML-Driven Predictive Anomaly Detection with Corda Integration

Input: IIoT device data, X, pre-trained ML model, M, anomaly threshold,T

Output: Anomaly detection Y, risk probability $P(Y = 1|X)$

1: **Data Collection**

Real-time data X from the IIoT sensors were continuously collected.

2: **Data Preprocessing**

Normalize and denoise the collected data for anomaly detection.

3: **Prediction**

Use the preprocessed data X as input into the ML model to check if there is likelihood of anomaly happening M(X) $\rightarrow P(Y = 1|X)$

4: **Anomaly Detection and Response**

Trigger an alert and log anomaly in provenance $P(Y = 1|X) > T$

The anomaly is notified to blockchain peers as a record for further investigation.

Trigger an alert and log anomaly in provenance

5: **Repeat**

Continuously monitor and detect anomalies in real-time.

3.10.3 Algorithm 3

This algorithm takes advantage of Corda smart contracts to validate the behavior of IIoT devices before committing provenance data to the blockchain. When a transaction occurs, it is first processed through a smart contract that checks for potentially malicious behavior, flagging it when an anomaly is detected. The smart contract rejects transactions and alerts the network for manual review, thereby ensuring that only compliant data are recorded.

Let SC the Corda Smart Contract T_xX be the validation function such that $P(d,t)$ be the provenance log of d at time t.
If device behavior is normal,$V(T_x) = 1$ (valid). For anomaly detection, $V(T_x) = 0$ (invalid)

Algorithm Steps:

Algorithm 3 Smart Contract-Based Dynamic Validation and Anomaly Response

Input: Provenance transaction Tx, anomaly flag Y, blockchain B

Output: Validated blockchain commit $B(t+1)$

1: **Transaction Extraction**

Provenance data $P(d,t)$ are extracted from each incoming transaction T_x.

2: **Anomaly Validation**

Anomaly status Y retrieved from the ML model.

3: **Smart Contract Approval**

If Y = 0 (normal behavior), the SC authorizes transaction $V(T_x) = 1$ and the data are committed to the blockchain.

If Y = 1 (anomaly detected), the SC rejects transaction $V(T_x) = 0$, and issues an alert for further analysis.

4: **Repeat**

Verification and authorization of subsequent transactions according to device behavior

3.10.4 Algorithm 4

Unlike static consensus models, this algorithm implements dynamic consensus, meaning that they change the way they approve transactions, depending on the risk assessments of each transaction in real time. Transactions with lower risk levels are processed faster, whereas higher-risk transactions require broader validation from their blockchain peers.

Let $R(t)$ be the risk score of a transaction.
Let C be the consensus threshold.

Algorithm Steps:

Algorithm 4 Dynamic Consensus Mechanism for Risk-Based Blockchain Commit

Input: Risk score $R(t)$, blockchain B

Output: Adaptive transaction approval

1: **Risk Calculation**

The risk score $R(t)$ for each transaction is calculated using the ML-based anomaly detection output.

2: **Transaction Approval**

If $R(t) < C$, approve the transaction by using a fast PoA mechanism.

If $R(t) > C$, validation using multiple Blockchain validators is required for high-risk transactions.

3: **Commit Transaction**

If approved, the transaction is added to Blockchain $B(t)$.

Table 2: Summary: Algorithm–Component–Research Question Mapping

Algorithm	Functionality	Research Question(s)
Algorithm 1	Real-time provenance logging and periodic blockchain commit	RQ1
Algorithm 2	AI-based anomaly prediction and blockchain-integrated alerting	RQ2
Algorithm 3	Smart contract validation of device behavior	RQ3
Algorithm 4	Risk-based adaptive blockchain consensus	RQ3

3.11 System Design

The design integrates off-chain provenance logging with ML-driven anomaly detection, which can interface with a smart ledger blockchain to enforce security policies by using corda-based smart contracts. The off-chain layer in the design has two core components: machine learning and predictive auditing engine. The machine learning component was designed to learn from the data streams of IIoT devices used in cloud manufacturing. At every receipt of the data block (an entire row in the database), the random forest machine learning was implemented to predict the next data block. The predictive auditing engine was designed to identify risk levels based on a comparison between machine learning predictions and the actual data block received. The risk levels are fed to the blockchain's smart ledger, which records events related to the smart contract being executed. Blockchain was designed to enforce different actions on peers at each risk level. The framework is shown in Figure 14, which comprises the nomenclature taken from Figure 8.

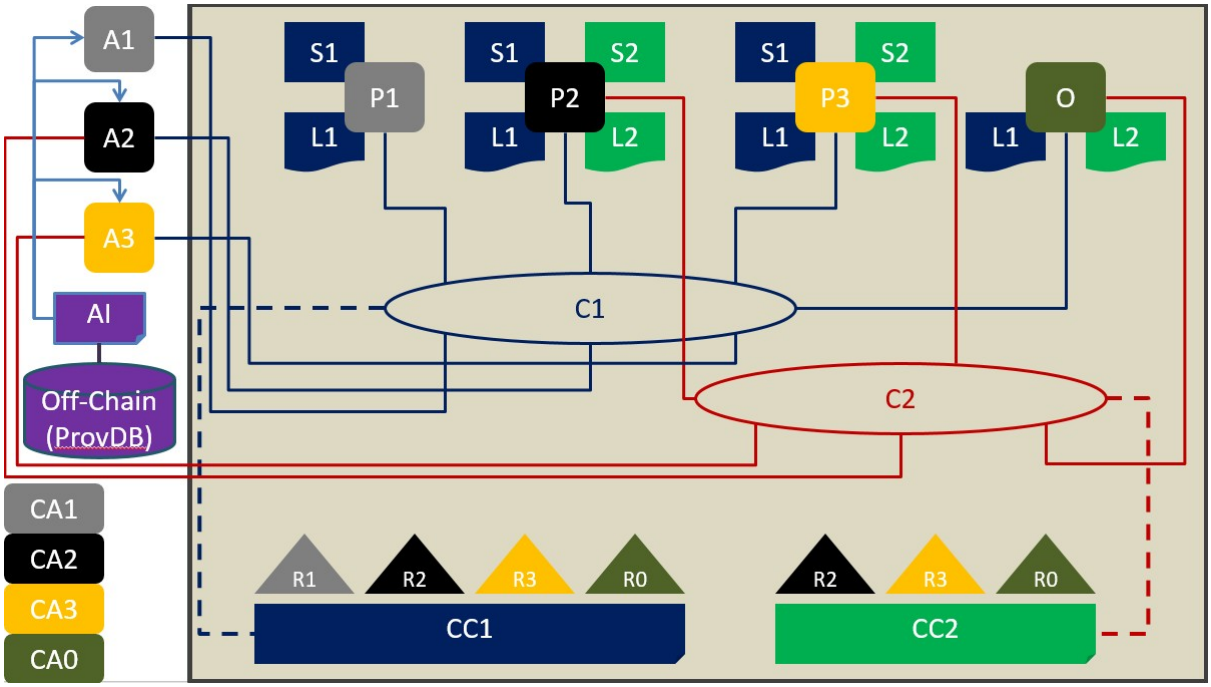


Figure 14: System Design.

As illustrated in Figure 14, the primary systems encompassing state databases S1 and S2 undergo regular updates upon the completion of events, as delineated by smart contracts within smart ledgers L1 and L2. Applications A1, A2, and A3 affiliated with organiza-

tions R1, R2, and R3, respectively, oversee the state changes in S1 and S2 on behalf of R0 through their respective smart contracts. Consequently, administrators A1, A2, and A3 operating outside the blockchain framework are responsible for ensuring the accuracy and reliability of event data recorded in the S1 and S2 state databases within the blockchain. Therefore, A1, A2, and A3 are integral to the development of robust security protocols for the proposed solution. Applications A1, A2, and A3 incorporate API interfaces that enable the reception of data from Industrial Internet of Things (IIoT) devices integrated into active processes. These applications are engineered to register IIoT devices by assigning unique hexadecimal keys to facilitate the data transmission. The data collected from these IIoT devices were used to update the states within the state databases of the blockchain. Once authorized, IIoT devices are expected to provide precise updates on events from ongoing operations, which can be used to modify the states in the blockchain's state databases. Provenance blockchain control has been proposed to ensure the consistent reliability of IIoT devices. Ideally, the unique hexadecimal keys assigned to IIoT devices for data transmission should serve as a proof of provenance. However, this method represents a singular form of control and does not guarantee continuous reliability, particularly in the presence of insider threats.

To ensure continuous verification of trustworthiness, a predictive auditing layer was created to build and use its own database, known as the Provenance Database (ProvDB), to conduct training and testing cycles. Random forest was used as the machine learning algorithm. The model was trained using historical data from ProvDB, which contains IIoT inputs from processes carried out to meet the conditions of smart contracts stored in L1, L2, and L3.

3.11.1 Design Analysis

Figure 15 presents a simplified version of the system design intended to enhance the comprehensibility of the complex elements depicted in Figure 14 and facilitate the implementation of the system. Within the domain of cloud manufacturing, IIoT devices can be compared to PLCs equipped with IP communication capabilities, enabling them to connect either to edge computing servers or directly to the internet [58], [138], [139]. These devices are integrated into cloud manufacturing systems that execute smart contracts

that are stored in a blockchain. They function as on-chain IoT devices and are directly managed by applications (A1, A2, and A3) under the control of the contracting parties. These devices record all the process events identified by the "provenance variables of interest" in an off-chain database. The provenance variables of interest are predefined within the blockchain rule engine and associated with the execution of smart contracts.

This study envisions a warehouse scenario in which forklifts are rented via a smart contract to operate within specified physical boundaries that smart contract owners must not exceed. The streams of location and weight data are treated as dynamic provenance sources.

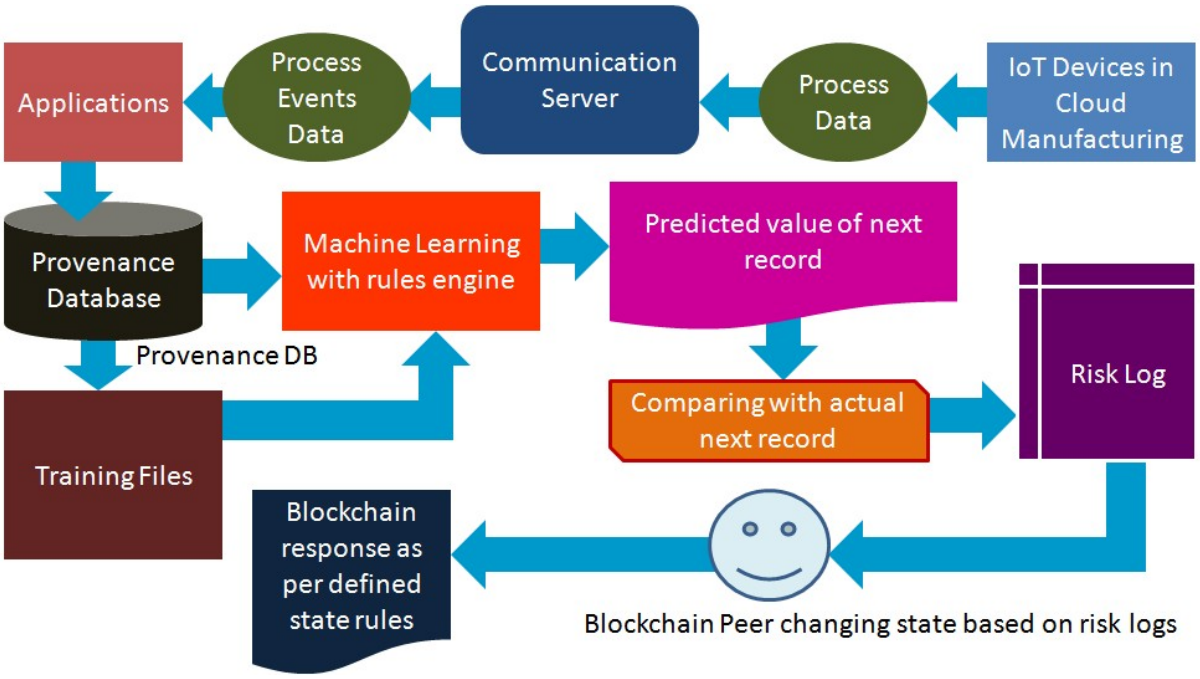


Figure 15: Design Analysis.

The attributes in question were delineated within an experimental framework wherein the Industrial Internet of Things (IIoT)-enabled forklift-loading machines were allocated to a warehousing smart contract functioning within a substantial multistory warehouse infrastructure. The monitored attributes include the movements of IIoT devices within the warehouse premises and the weight borne by each IIoT-enabled forklift asset.

A data management system systematically retrieves organized and structured records from an off-chain database to facilitate the training of a machine-learning algorithm. Through the application of supervised learning, the machine learning model is capable of

forecasting the "predicted value of the next record," which pertains to the forthcoming states of the variables of interest associated with smart contracts. The predictive auditing engine assesses the time-series patterns of the most recent predicted values of the variables against the actual values obtained. Concordance between these values indicates an absence of risk. Conversely, any discrepancy between the predicted and actual values of the variables may imply risks that could undermine trust in the IIoT devices.

3.12 Implementation

The implementation is realized using Corda, a distributed ledger technology (DLT) platform, to enforce on-chain provenance logging and access control, while Off-Chain ML models analyze transaction patterns and detect anomalies in real time. The system is composed of the following key Off-Chain and On-chain Components

3.12.1 Off-Chain Components Implementations

The off-chain component was implemented as a provenance predictive auditing system by inspecting the data streams sent by the IIoT devices that recognized provenance identification provided by unique hexadecimal keys allocated by applications A1, A2, and A3, as shown in Figure 14. Predictive analytics was implemented using a random forest machine learning algorithm and a time series-based engine. The machine learning algorithm made predictions for the next dataset based on learning from the data streams coming from the API clients (simulated as IIoT devices), and the time-series engine created risk logs by inspecting the differences in data between the predicted and actual datasets obtained from the API clients. The state of a specific IIoT device at time can be defined as

$$X_t = [x_t, y_t, z_t, W_t]$$

Where:

- x_t, y_t, z_t: represents the spatial coordinates of the device in a geo-fenced area at time t. These coordinates are used to identify the location of the device in specific areas (loading area, storage,dispatch area, etc.)

- W_t: represents the weight or load associated with the device at time, as recorded

statistics/operational data show how much loading the device carries.

The design of the predictive auditing architecture is illustrated in Figure 16. The ML design was conceptualized using random forest machine learning algorithms at the core. A Random Forest is formed as a collection of decision trees, also known as random trees. Random forests were selected because of their capability to make independent predictions regarding multiple variables streaming continuous datasets in a group, forming a structure based on past records [117].

The random forest algorithm was developed by Breiman in 2001 [117]. Random forests can be viewed as a collection of decision trees used as independent classifiers. Each decision tree can make its own predictions by generating branches based on simple Yes and No decisions. In experiments by Breiman in 2001, training datasets were built from 214 to 2310. In addition, he used three large datasets–7291, 4435, and 15000–to study the effects of large data sizes. They found that training data sizes above 435 resulted in low error rates (less than 10%) with large data sizes, which did not make much of the difference (up to 80% accuracy). However, to achieve accuracies greater than 80%, the data size must be significantly increased. Therefore, the size of the training dataset used in this study was 152400 records per asset.

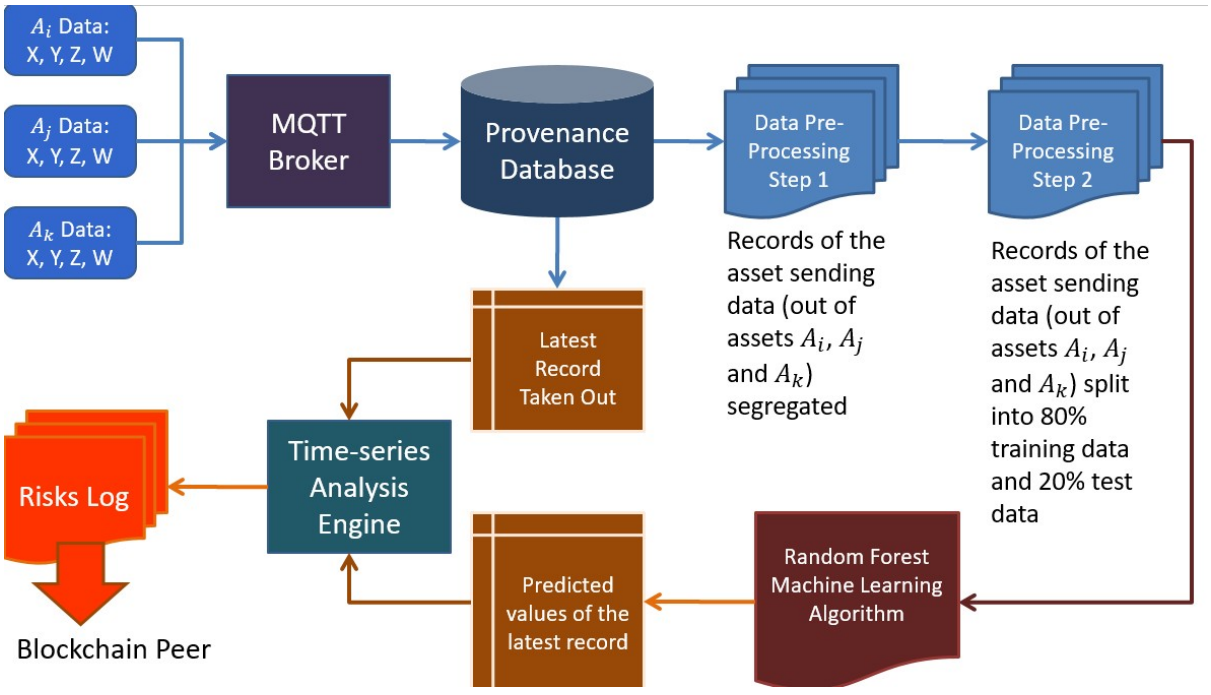


Figure 16: Predictive Auditing System Design.

Unlike prior approaches that primarily rely on high-level policy logs, inferred access control events, or abstracted device actions, which often fail to capture the nuanced operational behavior of IIoT devices in context, this study emphasizes the limitations of such indirect methods in detecting critical anomalies or behavioral deviations [153]. To address this gap, the proposed framework introduces a multidimensional behavioral capture mechanism that enables explicit, real-time monitoring and classification of device-level activities. In our simulated industrial environment, each IIoT-enabled forklift was instrumented with sensors to continuously stream raw, low-level data—comprising spatial coordinates x_t, y_t, z_t) and load weight W_t). These features collectively represent the dynamic physical state of the device at any given time. To enhance contextual awareness and traceability, the data were geofenced according to predefined functional warehouse zones—including storage, loading, and restricted areas—thereby enabling the generation of precise zone-transition events that reflect the spatial-temporal behavior of each forklift with high fidelity. These enriched logs are subsequently processed by a predictive auditing system that is implemented using a Random Forest classifier in conjunction with a time-series engine. The model is trained on historical labeled behavior to classify current states into categories, such as fully compliant, partially compliant, or rogue. Discrepancies between predicted and actual data were recorded as potential risks, each annotated with a compliance-based score [154]. This explicit behavioral model has three significant implications.

- ⋆ **Context-Aware Enforcement:** By incorporating real-time behavioral data, smart contracts can enforce policies based not only on user roles, but also on actual device behavior, thereby enhancing the granularity and effectiveness of access control mechanisms [153].

- ⋆ **Enhanced Anomaly Detection:** The system's capability to identify deviations in physical behavior, as opposed to relying solely on log-based heuristics, facilitates a more precise and timely detection of anomalies. This is essential for maintaining operational integrity in Industrial Internet of Things (IIoT) environments [155].

- ⋆ **High-Fidelity Operational Traceability:** The extensive logging and analysis of device behavior enable detailed forensic audits, offering comprehensive visibility of events and facilitating adherence to security standards such as IEC 62443.

The first step in the implementation involves collecting real-time data from the IIoT devices. In this study, the predictive auditing capability was implemented using data generated from MQTT API clients configured as IIoT sensors with transmitters. Applications A1, A2, and A3 in figure 16 have API interfaces on which IIoT devices deployed within operational processes are integrated. The role of these applications begins with the registration of IIoT devices by assigning them unique hexadecimal keys and categorizing them into a specific operational zone. Apache ActiveMQ was used as the MQTT broker to simulate a continuous real-time data transfer.

An MQTT server is installed and configured to receive transmissions on topics related to the assets assigned to the smart contract. Applications A1, A2, and A3 subscribe to relevant topics for each asset attribute. The IIoT data transmission was simulated using an MQTT-based messaging system. Several asset-specific topics were created in the MQTT broker to simulate the forklift operations in a warehouse environment. The concept of topic data is the specific context in which data are transmitted. In this study, the topic is data collected on attributes related to the IIoT asset engaged in a smart contract tracked through the blockchain. The attributes were defined in a scenario in which forklift-loading machines were assigned to a smart warehousing contract operated within a large multistory warehouse infrastructure. The monitored attributes are the movements of assets within the warehouse premises. The weights loaded on the assets of the simulated IIoT devices (publishers) were as follows:

String DATA = "X-axis movement in meters ";
String DATA2 = "Y-axis movement in meters
String DATA3 = "Z-axis movement in meters
String DATA4 = "asset loading weight in KG ".
String DATA5 = "asset ID
String DATA6= "yes"; Asset active/inactive Boolean
The string body = ""; combines all the above information and sends it to the MQTT broker.

Figure 17 illustrates that the entity relationship between MQTT topics, publishers, and subscribers is "many-to-many," meaning that multiple publishers can transmit data to a

single topic and multiple subscribers can access data from various topics. This "one-to-many" relationship between assets and MQTT topics.

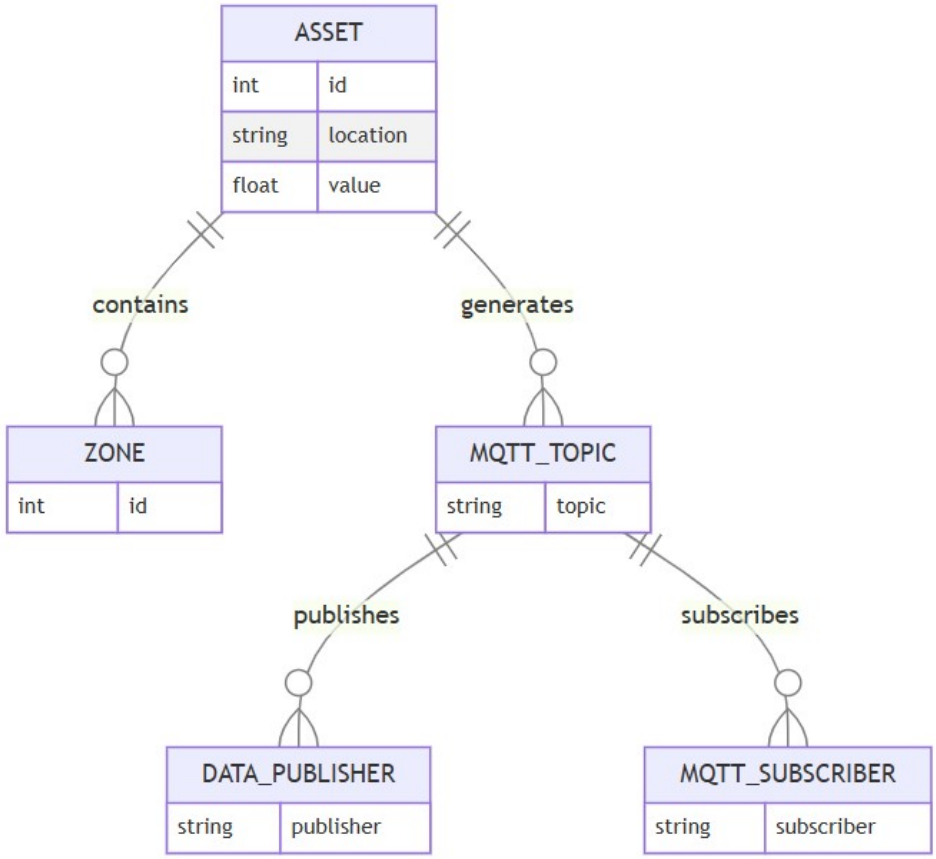


Figure 17: Entity relationship Diagram.

Each IIoT asset, represented by A_i, transmits data to structured MQTT topics, reflecting the hierarchical organization of warehouse zones and assets. Zones Z_j are predefined geo-fenced areas mathematically represented to ensure a systematic asset arrangement:

$$Z_j = \{X_{\min} \leq X(t) \leq X_{\max},\ Y_{\min} \leq Y(t) \leq Y_{\max}\}$$

Each IIoT-enabled forklift is assigned predefined movement boundaries and weight limits to reflect realistic constraints in a warehouse setting. The index of each asset within a zone is determined by

$$A_i \in Z_j \iff X_i \in [X_{\min}, X_{\max}]$$

Each asset transmits a data packet $D_i(t)$ containing the spatial coordinates $X_i(t)$,$Y_i(t)$,$Z_i(t)$ and weight $W_i(t)$.

$$Z_i(t) = f(_i(t), Y_i(t), Z_i(t), W_i(t))$$

In the diagram illustrated in Figure 17, data publishers (IIoT devices) transmit the data collected from assets to relevant MQTT topics. In this study, data from topic publishers were generated and transmitted using an Apache ActiveMQ MQTT Broker server. Applications A1, A2, and A3 employ MQTT interfaces to receive provenance data from Industrial Internet of Things (IIoT) devices connected to the forklifts. Data were produced using the publisher's module. The transmissions from the publisher (simulated IIoT devices) receive a subscriber to an MQTT broker, which acts as a listener. The data flows from the MQTT topic publisher (simulating the IIoT devices) module comprise stable data values of parameters constituting the events of a process. Intermittent anomaly values of the parameter variations at different levels were simulated in the data streams. In this study, the decision was made to refrain from using real IIoT devices within a laboratory setting, as the primary aim was to detect anomalies in the data they produced rather than focusing on the engineering of the IIoT devices themselves. In practical industrial settings, the publisher is integrated as firmware within physical IIoT devices, allowing data for the topic publisher to be automatically generated by industrial sensors embedded in these IIoT devices.

The real-time data collected from IIoT devices are temporarily stored off-chain to avoid overloading the blockchain and to ensure efficient data handling. Data were transmitted to ProvDB from Apache ActiveMQ, formatted as topic publisher data by a publisher module, with entries corresponding to the structure of the ProvDB database. The initial column contains the device keys necessary for registration, whereas the other columns display the numerical data gathered from the sensors during operational processes. The numerical data gathered from the sensors were transmitted by the topic publisher to subscribed listeners (A1, A2, and A3).

The provenance database was assigned to the outputs of the listeners (A1, A2, and A3). This is the "ProvDB.csv", in which the data sent on IIoT assets are stored. The attributes of all IIoT assets can be segregated and appended when the data are received from the Publisher through the MQTT broker, thus creating new rows of data being received. Upon receiving MQTT messages, the cloud manufacturing applications (A1, A2, and A3) temporarily buffer incoming data and segregate data points (location updates and weight measurements) for each asset over the defined time intervals. The device state

changes were captured and logged at predefined intervals as follows:

$$T_next + T_current = Delta(T)$$

Where T_next is the next logging timestamp, the $T_current$ represents the Current timestamp and $Delta(T)$ is the Predefined logging interval.

The ARFF format file "ProvDB" can be appended when the data are received from the Publisher through the MQTT broker, thus creating new rows of the data being received. The data were stored in the Provenance database (ProvDB), defining Asset ID as the primary classifier. A decision tree acts as a node in a data tree. In this study, SQL databases were used to simulate off-chain data storage for all records from field and individual asset-specific databases, which were later split into training and test databases.

The collected data were segmented by device IDs so that each device's operational data could be analyzed independently. The pre-processing step segregates the dataset of assets or depends on the latest data received. It builds the ProvDB database, segregates the training databases from ProvDB for each asset, and appends them with the latest data to generate prediction data for each asset by learning from the training data specific to the asset. The implementation simply identifies the data tree stored in the provenance database that is activated for training. To obtain highly reliable results [128], approximately 14200 records per IIoT were entered into this file to create a training database for machine learning. To operate the machine learning algorithms, the training dataset files were generated separately for each IIoT asset. The entries in the training dataset are clearly marked as anomalies and benign records. The data were stored in the provenance database under defined definitions, indicated by X (X coordinate), Y (Y coordinate), Z (Z coordinate), and W (weight).

In the next step, a random forest machine learning algorithm was trained. Random forests can be viewed as a collection of decision trees used as independent classifiers. Each decision tree can make its own predictions by generating branches based on simple Yes and No decisions. For example, it will split a dataset into two and mark a Yes for data above the splitting point and a No for data below it. The random forest algorithm creates an ensemble of all decision trees generated by a dataset such that their predictions can be combined to generate a singular and more accurate prediction. The random forest

algorithm randomly generates inputs following the simple formula Log2M + 1, where M is the serial number of the input (or simply an integer).

The process events from the IIoT device were scanned to capture behavioral provenance from traces of the operational histories of the simulated IIoT devices. The traces contain the latest MQTT records. Several rules were defined to split the ProvDB file into separate files comprising data for each IIoT asset. These files were further split into the training and testing datasets. After splitting the datasets, a new random tree was generated. Finally, the training data split was applied to predict the generated random tree. The machine learning model selects the training data based on Asset Id given in the "ProvDB." and appends them continuously as more data arrive from the publisher's execution. The data in these files were used in machine learning as training data for prediction. The latest data (approximately 20%) were used as test data, and the remaining data (approximately 80%) were used as the training data.

The split adapts to new real-time data patterns, allowing the model to evolve as more data is collected. Training data records were used per asset to capture all the relevant features and relationships within the data space. As this is a well-designed engineering process that follows some formula of operation, the predictions are expected to follow predictable time-series patterns. The predictions are made asset-wise such that the risk logs can be validated for the exact asset ID.

Referring back to Figure 16, the predicted value and the latest record were compared using a predictive auditing engine.

$$u_i(t) - P_t(t) > \epsilon(t)$$

The predicted state of a device $P_t(t)$ is compared with the actual observed state $u_i(t)$, and any deviation beyond a predefined threshold $epsilon(t)$ is flagged as an anomaly, which triggers an investigation. $epsilon(t)$ is the dynamic threshold that adjusts based on historical data.

In addition to the Random Forest, a time-series analysis was incorporated to track the temporal progression of the device states over time. Machine learning was applied to interpret ongoing data streams and anticipate forthcoming data sequences. A predictive auditing rule was established to evaluate the expected versus actual occurrence of the

subsequent data combinations. Alerts were generated to log the risks associated with four variables: movement along the X-axis, movement along the Y-axis, movement along the Z-axis, and weight being lifted. It was expected that the predictive auditing engine would not be affected by sudden anomaly injections because it would take into account hundreds of previous data rows for its prediction. This provides an opportunity for anomaly detection as there were differences between the actual values and values predicted by machine learning. These differences enforced the logging of risk levels in predictive auditing logs.

A dynamic risk score R(t) is calculated based on the deviation between the predicted and observed behaviors, considering historical trends and contextual factors:

$$R(t) = f(\Delta(D), H(t), C(t))$$

Where$\Delta(D)$ denotes the change in device state over time.$H(t)$ is the historical state of the device and $C(t)$ is the context, including conditions that could influence device actions. Based on the computed risk score, the devices were classified into different risk levels.

- NONE: No anomaly detected.
- LOW: Minor deviation detected.
- MEDIUM: Moderate deviation, requiring investigation.
- HIGH: Significant anomaly, requiring immediate action.

This dynamic risk-based classification ensures that the system adapts to the evolving nature of device operations. The computed risk score R(t) is categorized into discrete risk levels using thresholds $\epsilon_1(t)$,$\epsilon_2(t)$, $\epsilon_3(t)$

$$\text{Level(R)} = \begin{cases} NONE, & \text{if } R(t) < \epsilon_1(t) \\ LOW, & \text{if } \epsilon_1(t) \leq R(t) < \epsilon_2(t) \\ MEDIUM, & \text{if } \epsilon_2(t) \leq R(t) < \epsilon_3(t) \\ HIGH, & \text{if } R(t) \geq \epsilon_3(t) \end{cases}$$

Each Industrial Internet of Things (IIoT) device is assigned a security level, categorized as NONE, LOW, MEDIUM, or HIGH risk. These risk levels are determined based on the physical locations where the devices are authorized to operate and the maximum load capacity each device can manage. In practical applications, a multilevel warehouse may have forklifts restricted to designated areas. If these forklifts exceed their assigned boundaries, they risk encroaching on zones designated for other forklifts, potentially resulting in accidents. Furthermore, there may be issues in assigning forklifts to tasks that they are not equipped with, such as handling weights beyond their capacity. The severity of the boundary violation determines the risk level. The machine learning and predictive auditing engine developed in this study is designed to forecast the subsequent state values for each dataset received and then compare these predictions with the actual data to document any risks.

3.12.2 Implementation of On-Chain Components

(a) Provenance Log State (IOUState)

The Provenance Log State defines the attributes of each IIoT transaction, ensuring that the asset metadata of each IIoT asset, such as coordinates (X, Y, Z), weight, and risk classification, are immutably recorded on the blockchain. This implementation ensures the following attributes directly in transaction records that asset movements, operational loads, and security risk levels are recorded immutably on the distributed ledger so that the system can trace asset behavior across the ledger.

IOUState($s\% = y$, $s\% = Z$, $s\% = Weight$, $s\% = IIoTassetId$, $s\% = Risk$, $s\% = Sender$, $s\% = Reciever$, $s\% = linearId$", x, y, z, weight, assetId, risk, sender, receiver, linearId)

The key attributes of IOUState include the following.

- X, Y, Z: Positional coordinates tracking IIoT asset movement.
- Weight (W): operational weight of the asset.
- Risk – Security classification (e.g., NONE, LOW, MEDIUM, HIGH).
- AssetId: Unique identifier for the asset.
- sender, receiver – Parties involved in the transaction.
- linearId –encrypted ID of a CORDA transaction

(b) Integration with the Provenance Log Schema

The Provenance Log Schema (IOUSchemaV1.java) links the IOUState variables to the on-chain database schema, ensuring alignment between ledger storage and smart contract rules. IOUSchemaV1 is used to declare the exact columns and table names that will be referred to and in which the data will be stored. These column and table names are declared with variables and are used in the remaining variables. This schema ensures that all transaction data adhere to a structured format, making it easier to query and verify the provenance data. Figure 18 presents a flow chart illustrating the structure of

logging provenance data through smart contracts.

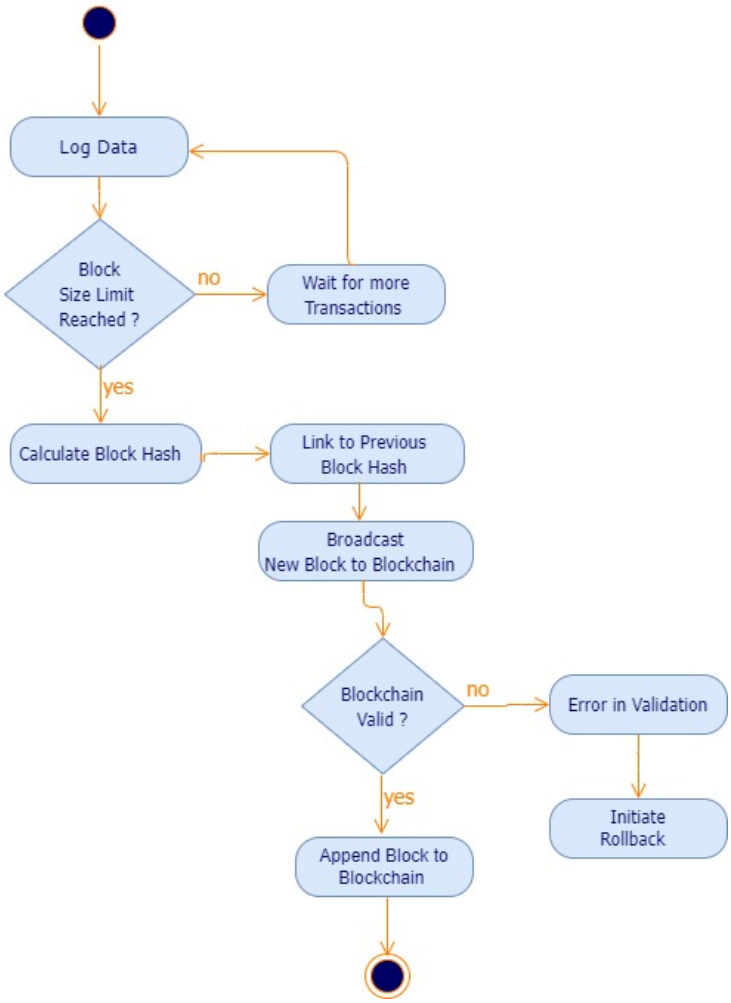


Figure 18: Logging provenance data.

To ensure that the off-chain logs remain tampered with, a Merkle Tree is computed over the latest IIoT data in the off-chain ProvDB. Each block in the blockchain contains multiple transactions. Before adding them to a block, they underwent Merkle Tree hashing.

- Each transaction (T1, T2, T3, and T4) is hashed individually (hash (T1) = SHA-256(T1)).
- The resulting hashes are combined and re-hashed in pairs - Hash(T12) = SHA-256(Hash(T1) + Hash(T2)),

This process continues until a single Merkle Root is formed.

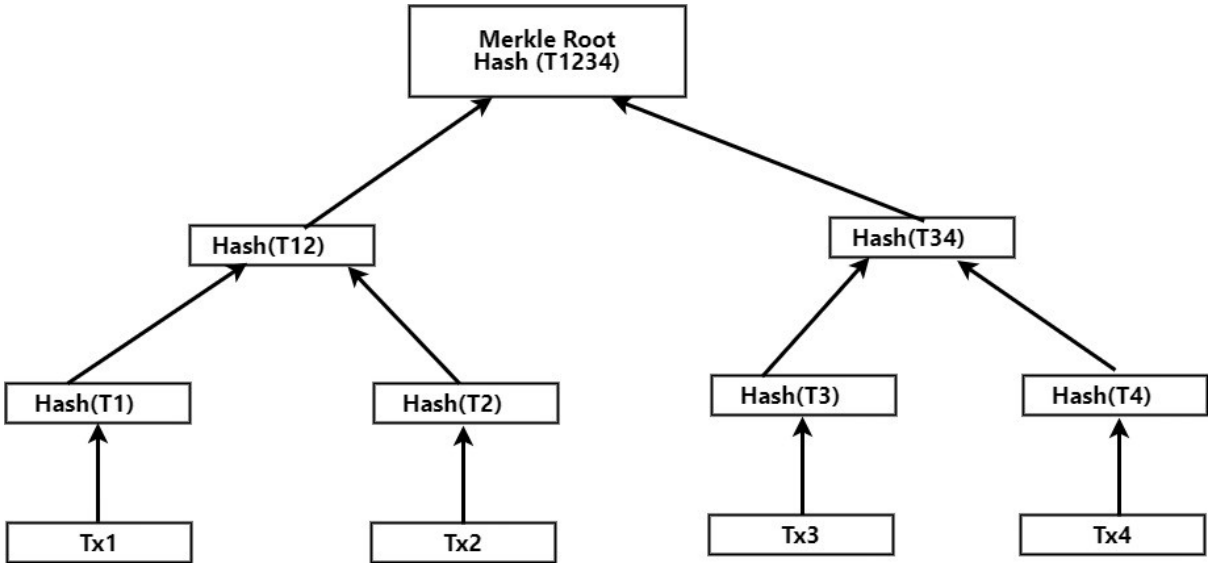


Figure 19: Merkle Tree hashing process.

The final Merkle Root is a single hash that stands for every transaction in a block. Each hash is based on its child nodes; therefore, tampering is easy to perform. If any transaction changes, the relevant hash changes and the entire tree changes up to the root.

Each new transaction in the blockchain is cryptographically linked to a previous transaction through its hash, thereby forming a verifiable and secure audit trail. Every block contains metadata with a hash of the current and previous blocks, thereby forming an irreversible chain of blocks. This means that once a block is added to the chain, it is very difficult to change any of the data, because it would invalidate all subsequent blocks and be detected by anyone looking at the chain. The linkage process is based on the following principles. Each block has hash "H(Bn), " which is produced by

- The hash of the previous block (H(Bn-1))
- The hash of the transaction data of the latest block, H(Tn)

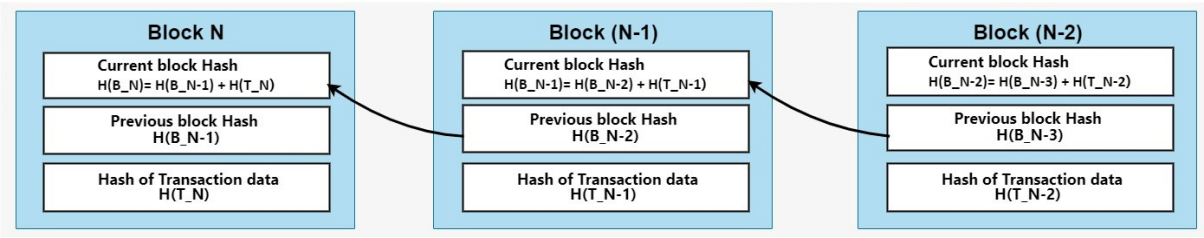


Figure 20: Cryptographic linkage process.

Each block contains a pointer to the hash of the previous block, H(Bn-1). If any data in a previous block are altered, its hash will change, invalidating all the subsequent blocks. A hybrid blockchain database was established by integrating on-chain and off-chain storage, leveraging Merkle Trees for efficient verification. When a blockchain peer requests provenance verification, the system retrieves the Merkle Proof (hash chain leading to Merkle Root). Merkle Proof was used to verify that the data were not altered. If the computed root matches the on-chain Merkle Root, the data are considered to be authentic.

3.12.3 Provenance Log Contract (IOUContract)

The Provenance Log Flow determines the procedure and when a specific task should be performed to ensure that the transaction is valid.

(a) Enforcing Transaction Rules

The Provenance Log Contract (IOUContract) which verifies whether IIoT data meets predefined conditions before allowing it to be recorded on the blockchain. Each IIoT-enabled forklift device's operational constraints are defined in the smart contract at the time of registering the assets in the blockchain. Contract verification methods are designed to reject transactions with values exceeding acceptable operational thresholds.

(b) Operational Boundary Conditions

If any of the device's state values fall outside the allowed boundaries, the smart contract rejects the transition and prevents the update from being recorded. This condition ensures that any action taken on the device is only valid if the device is within its geo-fenced location and its operational parameters are within acceptable ranges.

$$ValidateState(d_i) = \begin{cases} Allow(d_i), & \text{if } d_i, state \in S \textbf{(acceptable ranges)} \\ Reject(d_i), & \text{if } d_i, state \notin S \end{cases}$$

The validation logic ensures that only authorized transitions occur, while ensuring compliance with IIoT governance policies.

- $X, Y, Z \leq$ Maximum Limit **(ensures that asset movement is within operational limits)**

- $X, Y, Z \leq W_maxKG$ **(prevents overloading that could cause equipment failure)**

The oversight of smart contracts is centered on two fundamental quality objectives: the precise allocation of IIoT-enabled forklifts to their appropriate zones and weight capacities, and the strict enforcement of these forklifts operating within their assigned boundaries, thereby preventing unauthorized entry into zones designated for other forklifts unless a formal operational reassignment has occurred.

(c) Security Constraints

The smart contract continuously monitors the operational status of the device and verifies whether it meets the compliance information of the operating assets. Devices assessed as low or no risk were allowed to transition to new states without restrictions. However, devices flagged as medium- or high-risk by predictive auditing systems are restricted from performing state updates. In such cases, the smart contract automatically halts state updates and flags the device for further stakeholder investigation. Figure 21 presents the flow chart illustrating the structure of Enforcing Transaction Rules

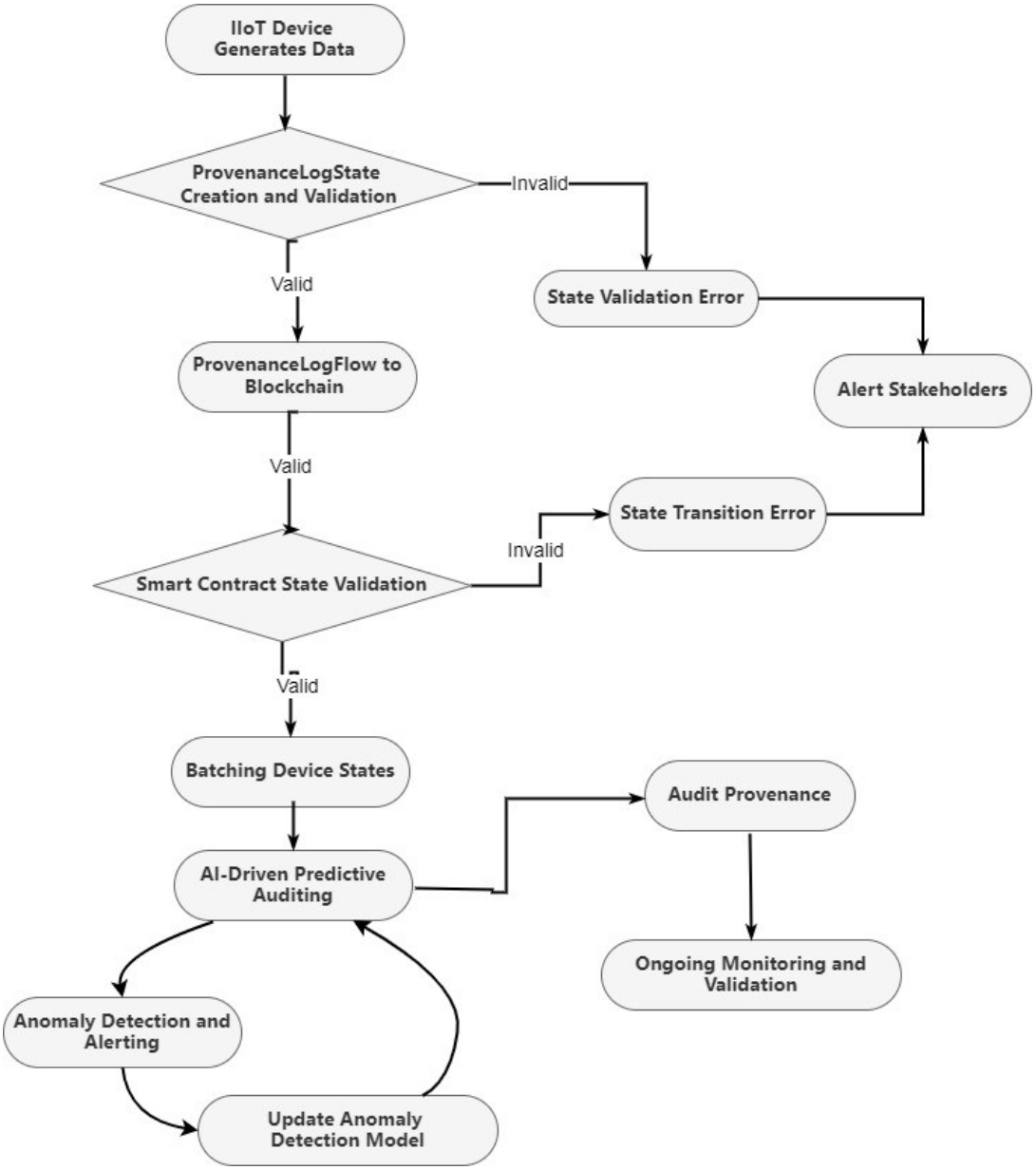


Figure 21: Smart contract state transitions process.

The state objects and smart contract validation processes ensure that state transitions occur only under predefined safe conditions to identify deviations from the expected behavior, enabling the proactive mitigation of potential security risks. Smart contracts enforce the following security policies based on the risk assessment provided by the AI-Driven Predictive Auditing model.

- A CORDA smart contract can undergo a state change when the risk level is classified as either NONE or LOW.

- In cases where the risk level is determined to be MEDIUM or HIGH, resulting in a restriction on state changes within the CORDA smart contract, direct the blockchain peers to investigate irregularities detected in the monitored IIoT devices.

Risk levels were determined by assessing the extent to which the constraints were breached. In cases where a forklift is identified as breaching X or Y, the risk can be classified as LOW because it may have been temporarily relocated from the warehouse to a parking spot for cooling purposes or to undergo necessary maintenance and repair. If there is a violation of the Z-axis constraints, which will occur alongside breaches in the X- and Y-axes because of the need to remove and relocate the forklifts via ramps, the recorded risk level will range from MEDIUM to HIGH, contingent on the extent of their displacement.

3.12.4 Provenance Log Flow

The Provenance Log Flow determines the procedure and when a specific task should be performed to ensure that the transaction is valid. It rejects values above the specified maximum limits, thus making the blockchain a control system that can reject illegal values and prompt blockchain peers to initiate investigations and take corrective actions to bring values within the allowed ranges. The Provenance Log Flow automates the transaction lifecycle, embedding real-time security validation and provenance auditing in the process. The flow consists of the following steps.

1. **GENERATING TRANSACTION:** A blockchain peer (initiator) creates a transaction with the Provenance Log State.

2. **VERIFYING TRANSACTION:** The transaction is validated against rules in the Provenance LogContractact.

3. **SIGNING TRANSACTION**: The initiating party (sender) digitally signs the transaction.

4. **GATHERINGS SIGNATURES:** Other authorized signers provide signatures.

5. **FINALISING TRANSACTION:** The transaction is finalized and becomes an immutable ledger record.

Figure 22 presents a flowchart of this process, illustrating how Corda's notary service and AI-based anomaly detection intervene when security breaches are detected.

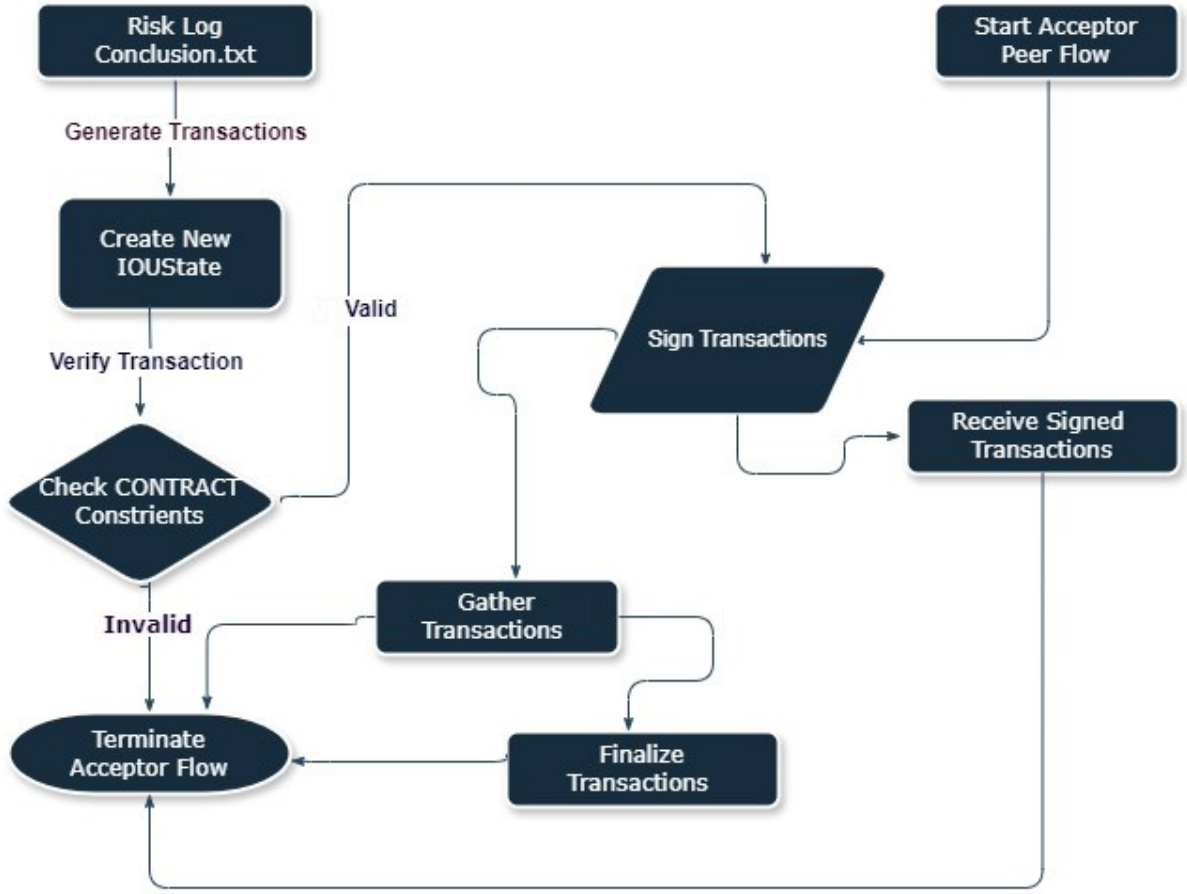


Figure 22: Provenance Log Flow.

3.12.5 Contract and Flow Testing

(a) Smart Contract Validation (ContractTests)

The ContractTests file consists of more conditions and rules for a transaction to follow to ensure that the Provenance Log Contract (IOUContract.java) correctly enforces all defined security constraints. During the transaction process, tests are performed in which the file undergoes different conditions given in the code. The key validation cases include the following.

- Rejecting invalid asset movements.
- Enforcing weight constraints.

- Ensuring risk-level compliance.

Figure 23 presents a UML class diagram, illustrating Contract and Flow Testing integration with other components

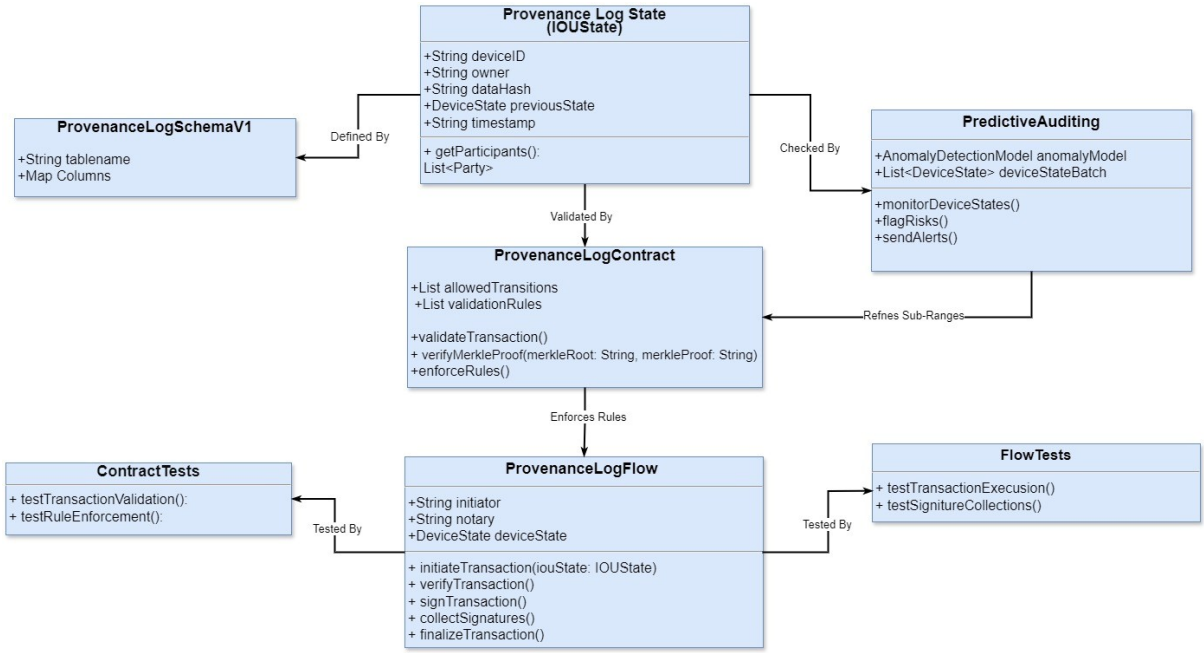


Figure 23: Contract and Flow Testing.

(b) Transaction Execution Testing (FlowTests)

The FlowTests file verifies the correct execution of the Provenance Log Flow (Flow), ensuring that:

- Transactions satisfying all security conditions are accepted and recorded.
- Unauthorized transactions are rejected before finalization.

3.12.6 Smart Contract Versioning and Policy Evolution

Given the evolving nature of operational policies in industrial environments, such as risk thresholds, access control rules, and compliance standards, this framework embeds a version-controlled smart contract architecture within Corda. This feature enables flexible policy governance without disrupting the system continuity.

- **Version Identification:** Each smart contract is deployed with a unique version

identifier, allowing legacy and updated policies to coexist.

- **Immutable Audit Trail:** On-chain logging of contract deployment events creates an immutable audit trail that supports the transparent tracking of policy modifications over time.

- **Context-aware vocation:** The framework allows context-aware contract invocation, where the active contract version is selected based on the actor role, operational context, or specified upgrade flags.

Implementing smart-contract versioning is crucial for maintaining system adaptability and compliance. Corda supports contract upgrades through mechanisms such as hash and signature constraints, allowing for controlled and flexible updates to smart contracts [156]. Furthermore, Corda's introduction of the zone constraint in Version 3 allows for implicit upgrades of contracts, enabling nodes to individually upgrade contracts without requiring simultaneous updates across the network [157].

3.12.7 Integration of Off-Chain and On-Chain Components

The integration of AI and blockchain allows for the combination of real-time anomaly detection, data provenance tracking, and predictive auditing, thereby ensuring enhanced security in IIoT environments. This study integrated the prediction of provenance information using machine learning with the state transition rules engine of a smart contract loaded on a blockchain to continuously monitor the compliance of operating assets within the boundaries defined by the smart contract. Figure 24 illustrates the structural relationships and interactions among the components of the framework.

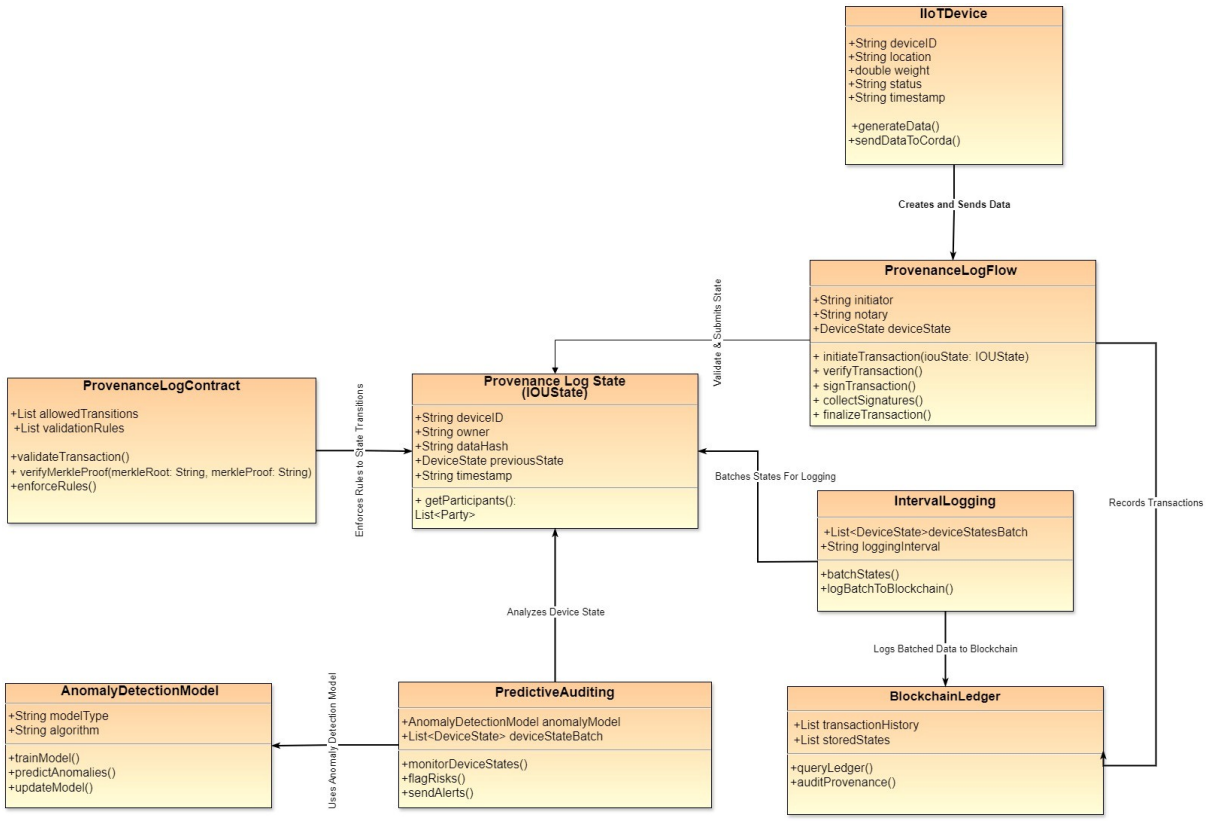


Figure 24: Components of the framework.

IIoT devices collect data from processing events and transmit them to the processing system. The device state was created from the incoming data and validated using DeviceState. Blockchain state tracking assesses predefined rules in DeviceContract to determine the validity of state transition. DeviceFlow initiates the transaction,signs it, and logs it into a Blockchain Ledger as soon as it has been validated. DeviceContract ensures that the business rules prevent an invalid state transition. It checks the state change

against the valid transitions and validation rules. The system aggregates the provenance data into a batch for storage efficiency on the blockchain and then logs the batch into BlockchainLedger. The device states were continuously monitored using predictive auditing. The AnomalyDetection Model was then trained on the collected states to detect possible anomalies and flag those that crossed the risk thresholds. The alerts were sent to stakeholders for data investigation and remediation through predictive auditing. The system queries the content of the Blockchain in Ledger to obtain the historical provenance of the device states. Thus, all changes can be audited, thereby facilitating traceability and accountability. Anomalies can be retrained in the anomaly detection model if they are frequent or if new patterns can be detected.

3.12.8 Design Trade-Off: Scalability versus Intelligence and Security

The integration of predictive auditing and blockchain validation presents inherent trade-offs in terms of scalability. Real-time AI inference requires substantial processing capacity, particularly within high-frequency Industrial Internet of Things (IIoT) environments. Simultaneously, CORDA's distributed architecture, although optimized for enterprise use, imposes latency overheads owing to consensus mechanisms and smart contract execution. To address these challenges, the framework employs a layered and modular architecture: ML computations are conducted off-chain using parallel batch processing, only validated outcomes are committed to the blockchain, and smart contract logic is confined to access control decisions to mitigate the load. This design ensures that real-time performance is maintained at the edge, whereas security and auditability are assured on the chain. Nevertheless, this approach prioritizes trust and accuracy over horizontal scalability, indicating that future optimization is essential for deployment on an industrial scale.

3.12.9 Design Trade-Off: Immutability versus Auditing Flexibility in a Scalable System

The integration of real-time predictive auditing with blockchain-based enforcement presents a fundamental architectural trade-off between immutability, a core principle of blockchain technology, and the adaptive flexibility necessitated by machine learning (ML) anomaly

detection. This dichotomy is particularly challenging in the context of the Industrial Internet of Things (IIoT), where systems must manage high-frequency data, evolving operational profiles, and low-latency decision-making [158]. To address these tensions, the proposed framework employs a layered architecture that separates compute-intensive auditing tasks from tamper-resistant controls and logging functions. Specifically:

- Predictive auditing was employed as an off-chain module, wherein a Random Forest classifier continuously assessed real-time data from IIoT-enabled forklifts. This approach enables the framework to perform high-frequency machine-learning inference without causing congestion in the blockchain network [159].

- The logic of smart contracts is deliberately limited to enforcement decisions rather than computational logic. This approach is consistent with CORDA's enterprise principles, which prioritize deterministic contract execution and efficient notarization [160].

To maintain the immutable nature of the blockchain while permitting adaptive corrections, the system utilized an append-only correction approach. Misclassifications or false positives are neither erased nor modified; rather, they are rectified by recording subsequent compensatory events that are also regulated by smart contracts. This approach preserves the integrity of historical audits and allows for traceable reversals, thereby enhancing transparency and accountability [161]. Furthermore, these corrective measures are integrated into a model retraining loop, allowing the auditing engine to adapt to the operational dynamics. Over time, this feedback mechanism enhances the model's generalization capabilities, thereby reducing the incidence of false alarms and improving the overall reliability in diverse IIoT environments [162]. Nonetheless, the modular design of the framework presents scalability tradeoffs. Frequent AI-based inferences coupled with the enforcement of smart contracts can place substantial demands on computational and communication resources in large-scale deployments. To facilitate broader industrial adoption, future extensions may investigate distributed edge analytics, smart contract orchestration layers, and lightweight consensus algorithms optimized for IIoT [163].

3.13 Deployment Structure of the Framework

The framework was deployed in a simulated cloud manufacturing environment that involved the following key nodes and components:

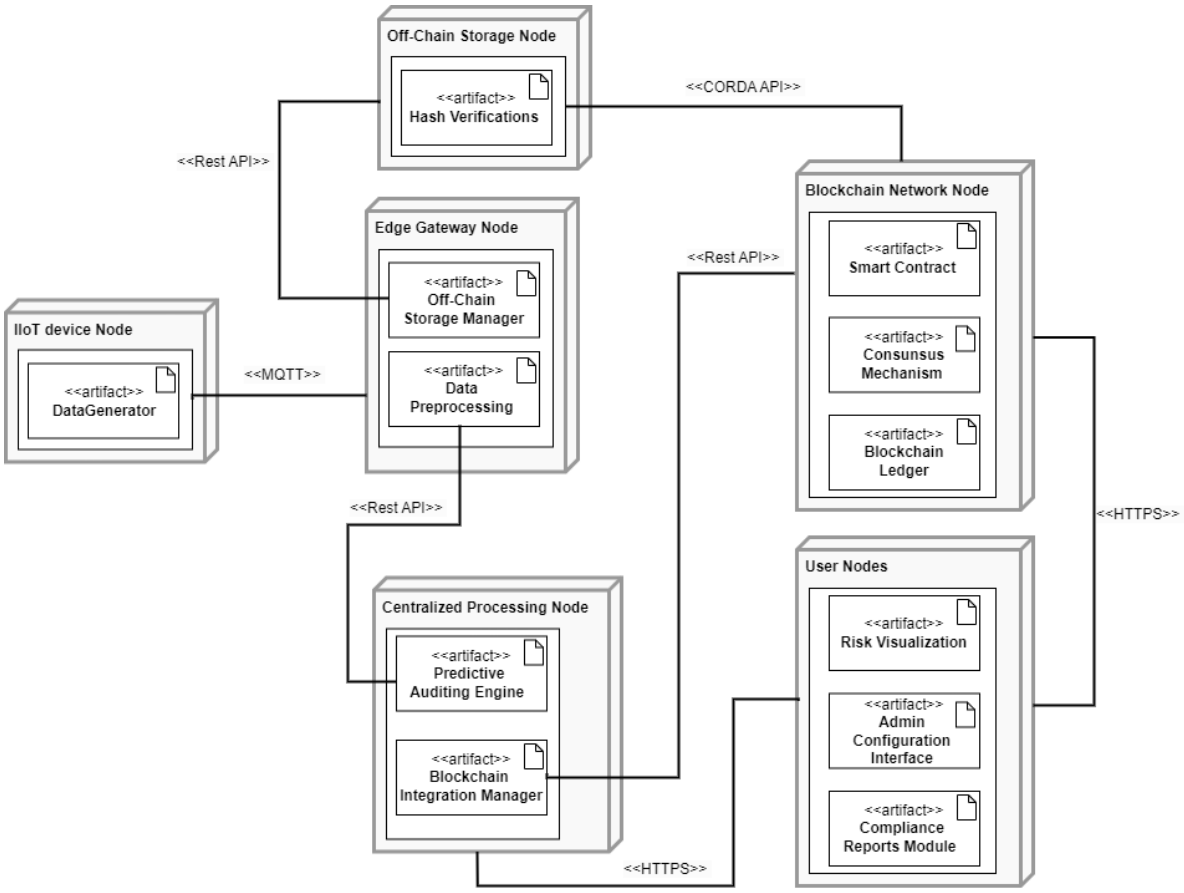


Figure 25: Deployment Diagram.

3.13.1 Communication Workflow

(a) IIoT Device → Edge Gateway

The workflow is initiated using IIoT devices that send real-time information to an edge gateway via MQTT protocols. Sensors that gather essential operational data (location coordinates (X, Y, Z) and weight (W)). First, the Edge Gateway, which resides at the network edge in the geo-fenced environment, was used for processing.

(b) Edge Gateway → Off-Chain Storage

A Real-Time Monitoring Module processes the received data from the Edge Gateway, which forwards the processed data to the Off-Chain Storage, where a large volume of data can be managed more efficiently. The off-chain allows the operational data to be stored without polluting the block space, and subsequently reduces the process load from the blockchain. These processed data are then periodically hashed and transferred to the blockchain to guarantee data integrity.

(c) Edge Gateway → Centralized Processing Node

Processed data are sent from the Edge Gateway to a centralized processing node using REST APIs. Predictive Auditing and Blockchain Network Management. Centralized Processing aggregates data from multiple edge gateways, runs machine-learning models that evaluate risks, and raises red flags when it senses that security is at stake.

(d) Centralized Processing Node → Blockchain Network

The validated data are securely hashed and sent to the Blockchain Network from the Centralized Processing Node. The blockchain acts as a decentralized ledger to provide an immutable log of events and transactions. The system dynamically adjusts the consensus process based on the risk level, allowing high-risk data to be immediately validated.

(e) Blockchain Network Off-Chain Storage

The Blockchain Network interacts with Off-Chain Storage to cross-validate off-chain hashes against on-chain hashes periodically, using all blockchain nodes.

(f) Centralized Processing Node → AI Predictive Auditing Model

The Centralized Processing Node, embedded with an AI Predictive Auditing Model, monitors incoming data in real-time to detect anomalies, evaluate operational risks, and predict threats. The model was trained to analyze the input data and identify anomalies or deviations from the expected behavior.

(g) User Interface Nodes Processing Nodes

User interactions occur via secure User Interface nodes, hosting dashboards, and command-line tools on the Centralized Processing Node. The system uses HTTPS over TLS for

secure data transfer and role-based access control (RBAC) to protect sensitive operations. External Users (Portal Users) have limited access, primarily read-only access to system reports and analytics, whereas administrator/operator nodes can configure the parameters for the system and monitor alerts for compliance.

3.13.2 Private Cloud Setup (VM-Based)

The system processed real-time data streams from up to 50 simulated IIoT devices, producing a data point of 256 bytes per second. Increments of 10 s were sent for anomaly detection (in batches of 500 points). These are resource allocations for Cloud Instance 1, real-time monitoring and anomaly detection, cloud instance 2, Blockchain Node Management and Predictive Auditing.

(a) Cloud Instance 1 (Real-Time Monitoring and Anomaly Detection)

Cloud Instance 1 is responsible for handling real-time data output using simulated IIoT devices, conducting anomaly detection, and managing the storage of noncritical data in Off-Chain Storage. It communicates with Cloud Instance 2, which is responsible for transmitting the processed data for the validation and verification processes.

Resource Configuration:

I. Compute Resources:

   a. **vCPUs**: 6-8 virtual CPUs: Optimized for real-time processing of up to 50 devices running anomaly detection algorithms. Additional CPU threads were provisioned to accommodate the peak loads incurred owing to the state changes or simultaneous anomaly detection.

   b. **Memory (RAM):** 24-32 GB: Adequate for use as a buffer and processor for real-time data streams, running detection algorithms in real time, and handling data deduplication and compression.

II. Storage:

   c. **SSD Storage:** 1-2 TB (essential for logging and processing real-time operational data efficiently). SSDs are preferred for lower latency and faster access

for immediate analysis).

d. **Off-Chain Storage:** 5-10 TB (This is scalable storage to save raw IIoT data on any intermediary data storage or offloading on the blockchain. This allows the frequent data generated by the devices to be managed.

III. Storage Management:

a. **Hashing Mechanism**: Data batches are streamed via Apache Kafka, which hashes them and prepares them for logging on the blockchain.

(b) Cloud Instance 2 (Blockchain Node Management and Predictive Auditing)

Cloud Instance 2 is responsible for managing blockchain nodes and integrating AI models to perform anomaly detection, predictive audits, and logging data into the blockchain.

Resource Configuration:

I. Compute Resources:

c. **vCPUs:** 10-16 virtual CPUs: Dedicated resources to run blockchain nodes, maintain a distributed ledger, run machine-learning models, and conduct predictive audits.

d. **Memory (RAM):** 40-64 GB: It offers sufficient computing capability to run ML models, process blockchain transactions, and execute auditing procedures.

II. Storage:

a. **SSD Storage:** 2 TB: Supports fast write andread operations for transaction logs on the blockchain, ML model outputs, and predictive audit results.

III. AI and Predictive Auditing

a. **AI Model Execution**:1-2 GPUs for compute-intensive AI models for anomaly detection and predictive analytics.

3.13.3 Testing and Validation

Multiple testing phases were used to ensure the functionality of the framework. This included unit tests, integration tests, and system simulations, all of which helped to confirm that the system behaved as intended under a range of conditions. The system was tested in a simulated warehouse environment, which allowed us to assess the real-time performance as well as to determine the efficacy of providing IIoT devices, predictive auditing, and blockchain capabilities

3.13.4 Tools and Technologies

Several key technologies and tools are used to realize the proposed framework, including IIoT simulation tools, private cloud platforms, machine-learning frameworks, and blockchain technologies.

(a) IIoT Device Simulation and Data Generation

(b) Simulated IIoT Devices

Simulated IIoT Devices were created with an Apache ActiveMQ MQTT Broker server running on a dedicated local host port (representing a separate API interface) and an API client programmed in Java representing the MQTT transmitter to the Apache MQTT Broker server). MQTT clients were simulated as IIoT devices using Python-based scripts to emulate the behavior of IIoT devices and to generate data streams. The MQTT Broker server was configured using Apache ActiveMQ 5.15.0.

(c) Private Cloud Infrastructure

The framework was deployed in a private cloud environment using virtual machines (VMs) to manage and process IIoT data. Two native Linux configurations are interconnected using a Wireless LAN router. Both instances ran an Ubuntu 20 (minimum version) Linux distribution. They were configured with Java JDK 8 runtime, because the Corda framework uses an old version of the spring framework that does not support anything beyond Java JDK 9. Hence, a safe choice was the Springring Framework using Java JDK 8. The full framework was configured using the Maven software.

(d) Real-Time Data Processing and Predictive Auditing

Random Forest Machine Learning (ML) algorithms were used to identify anomalous patterns in IIoT data streams. The package used for coding the machine learning code was Weka [91], which was built in Java 8. Time-series forecasting models (LSTM) were used to identify patterns and trends in the historical and real-time data. A Predictive Analytics Algorithm provides provenance data and alerts based on the predicted data. Software programming and runtimes were conducted using the Java 8.

(e) Blockchain Technology

The blockchain component ensures data integrity, transparency, and secure logging of IIoT device states.

- Corda Platform: Corda Community edition 4.10 was selected because of the ease of configuration and conservation of hardware resources. Corda can be installed without Docker and Kubernetes in a core Java development environment using a suitable IDE (Integrated Development Environment). The IDE used was IntelliJ Idea 2023.2.2 (Community Edition) Runtime version: 17.0.8+7-b1000.22 amd64. The IDE was used to import Java 8 packages, develop Java 8 codes, and program and operate the runtime of the Corda code. Corda works with Java 8; therefore, Java 8 JDK (Java Development Kit) was selected for this study.

- Smart Contracts: On the CORDA platform, smart contracts are utilized to verify and ensure compliance with state changes and transactions. To define the rules governing smart contract states in Corda, six Java files were configured: IOUState, IOUContract, IOUSchemaV1, ContractTests, Flow, and FlowTests. The smart contract under examination in this study was designated as the IOU. During the configuration of these files, a database file named "iou.changelog-v1. xml" is automatically generated. This file functions as a change-log database and records state changes within the IOU smart contract. It is imperative that all variables defined in the Schema and other files, such as States and Flow, have corresponding entries in the change-log database.

(f) Off-Chain Storage and Data Management

- **Off-Chain Storage:** SQL Databases: To handle the off-chain storage of raw and processed data records, SQL relational databases were used for temporary storage of the raw data generated by the IIoT devices before they were hashed and logged onto the blockchain.

- **Hashing Mechanism:** SHA-256: Hash data before logging onto the blockchain, ensuring integrity.

3.14 Ethics Considerations

Research ethics have several dimensions. The dimensions are mostly related to access to respondents and data, access to content, interactions and behavioral aspects, originality of findings and analysis, and protection of intellectual property rights [128]. Ethics are often practiced to solve the conflict between the researcher's desires and the possibilities available. This basic theory is applicable to all the research steps. This study did not involve interactions with human participants. It is based on designing and realizing the blockchain framework, which also has ethical implications. This category of research comes under design studies; hence, ethical conduct in design-based research applies [132]. A fundamental ethical consideration is that design studies must be socially responsible. Design studies are a practice-based approach to educational enquiry [147]. Designs studied for educational inquiry have found ways in the commercial world to lead to the production of products and services used by people. Hence, we should ensure that the design created does not cause any form of harm to the end customers or should be biased to deliver results at the cost of harming other results already useful to the users [148] – [150]. At the engineering level, designs should comply with the health, safety, and security requirements of the end customers. Furthermore, we should be very careful in selecting the technologies to be used for shaping designs into actual systems, products, and services in the educational and commercial world [151]–[153]. Designs should be economical, people-friendly, environment-friendly, harmless, safe to use, and protect vulnerable people. In addition, designs should respect originality and comply with the rules of patents and intellectual property rights. These ethical aspects were carefully analyzed while designing the framework. It was ensured that there would be no harm to people or the environment caused by the developed and tested framework. In terms of provenance

data privacy, the framework captures extensive metadata, including timestamps, asset movement histories, and AI-generated compliance scores to ensure operational traceability. To protect privacy:

- All device identifiers were hashed and anonymized prior to logging.
- No user credentials or personal data were stored.
- Role-based actor identifiers (e.g., forklift, supervisor) were used without linking to specific individuals.

The system assumes that identity management and consent protocols are implemented at the organizational level. This means the framework provides privacy-aware design scaffolding, but not a complete regulatory compliance mechanism such as GDPR enforcement. It is acknowledged that the current version lacks a dedicated privacy engine and does not support data erasure, minimization, or user-level access control. As such, it may not fully meet data protection mandates in privacy-sensitive deployments. These enhancements are necessary for deployment in regulated industrial ecosystems such as multi-tenant clouds, cross-border supply chains, or sectors subject to stringent data governance requirements.

Chapter IV

Experiments and Results

4.1 Introduction

This chapter presents the detailed results related to the testing and validation of the proposed framework. The evaluation was centered on a geo-fenced warehouse to replicate real-life scenarios and test the functionality of the framework under controlled conditions. The experiments aimed to prove the ability of the framework to ensure data integrity, traceability, and operational security while also demonstrating its effectiveness in mitigating security threats in Industrial Internet of Things (IIoT)-driven C-MFG environments.

4.2 Experimental setup

Experimental setup for evaluating the proposed framework to ensure traceability, security, and efficiency in a simulated IIoT environment. The experiment simulated a cloud manufacturing system that required strong monitoring, integrity, and anomaly detection mechanisms to ensure operational freedom. The experiment was designed to test the feasibility of the proposed method for achieving real-time data traceability, preventing unauthorized data manipulation, and efficiently handling different device operational states.

4.2.1 Hardware and Software Infrastructure

The experimental environment consisted of cloud instances, simulated IIoT devices, and blockchain-based provenances. The setup adheres to international safety and system integrity standards, specifically ISO 3691 [165] for industrial trucks and IEC 61508 for safety-related programmable systems [166].

(a) Cloud Instances and Virtual Environment

The framework was deployed on two cloud-based virtual machines (VMs) to simulate the IIoT security architecture, emulating real-time data generation and transmission pipelines common in IIoT-integrated manufacturing environments [167], [168].

1. **Cloud Instance 1: Real-time IIoT Data Collection and Monitoring**

   - Off-chain storage is implemented for the Apache ActiveMQ MQTT Broker running on a dedicated local host port (representing a separate API interface) and an API client representing the MQTT transmitter to the Apache MQTT Broker server.
   - Hosts the API client (simulated as an IIoT device) that generates real-time data.
   - Run-predictive auditing system (ML-based).

2. **Cloud Instance 2: Predictive Auditing Blockchain Logging**

   - Hosts operational runtime of the blockchain framework (corda-based).
   - Logs device state changes onto the blockchain for provenance tracking.
   - Enforces smart contracts for compliance verification.

(b) IIoT Device Simulation

1. ***Simulation Environment***

   The environment was simulated using a multistory warehouse, where each forklift machine was assigned a fixed boundary. The warehouse is divided into three zones: zone 1, zone 2, and zone 3, each with specific operational constraints, such as weight

limits and location coordinates (X, Y, Z). These zones serve as controlled areas where the behavior of the forklifts is monitored in real time to ensure compliance with predefined operational rules. A simple schematic of this process is shown in Figure 26.

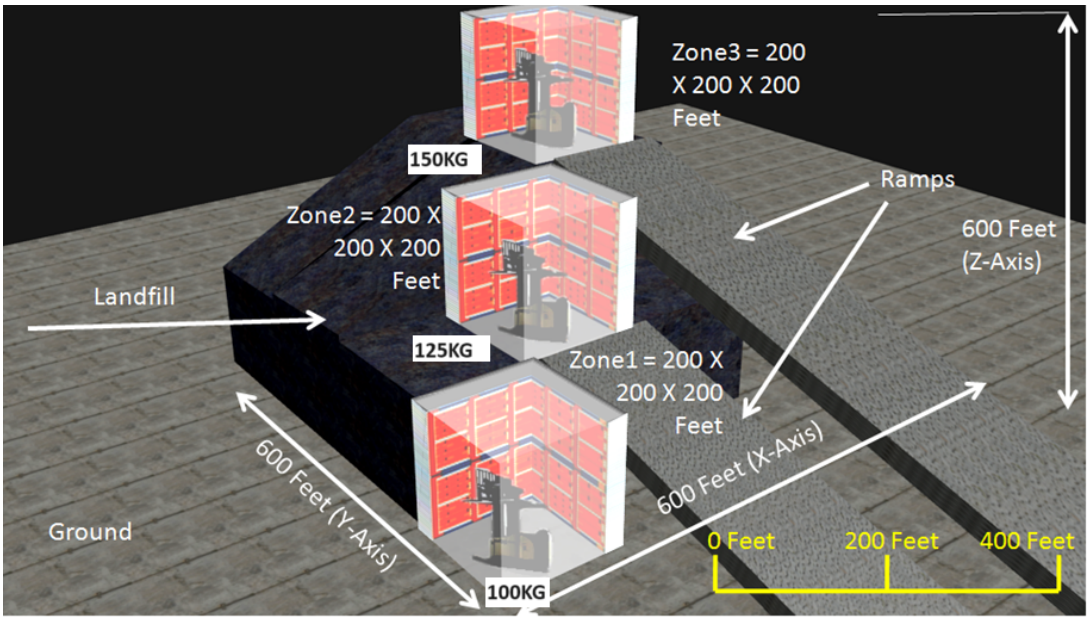


Figure 26: A simple schematic of warehouse environment.

Configuration of Operating Zones:

The experimental setup involved a scenario in which Industrial Internet of Things (IIoT)-enabled reach truck forklifts, designed for high-rise warehousing with vertical storage capabilities, were allocated to three distinct operating zones within a warehouse. Each zone measured 200 fts in width, height, and length. Although these zones were adjacent, they were not interconnected. During the testing phase, these zones were designated as Zone1, Zone2, and Zone3. Zone1 was located at ground level, Zone2 was positioned on a landfill approximately 200 feet high, and Zone3 was situated on a neighboring landfill approximately 400 feet high. The experiment was conducted considering the average dimensions of a medium-sized warehouse.

2. ***Simulated device attributes***

   - Device ID (unique identifier for each IIoT unit).
   - Location Coordinates (tracking device movement in warehouse).

- Weight Sensor Data (weight measurements to detect anomalies).
- Device Interaction Logs (capturing all state transitions).

These features enabled effective behavior modeling and real-time anomaly classification, consistent with IIoT and predictive analytics research [169], [170]. As the experiment was conducted in a simulated environment, real IIoT devices were emulated using a Python-based simulation tool. The simulated IIoT devices (API client or topic publisher) generated values within a predefined range for each zone and device using the API client for device tracking and security monitoring. The generated data are stored off-chain in real time and hashed/logged into the blockchain at fixed intervals. The testbed comprised twenty virtual forklifts, each equipped with embedded simulated IIoT modules, operating across three logical zones. The virtual forklifts were programmed to exhibit distinct behaviors—compliant, partially compliant, and rogue—in accordance with defined boundaries.

(c) Blockchain Configuration The blockchain network was configured using Corda with a Java JDK 8 runtime. The setup was followed to encode ProvDB as a separate application connected to the main smart ledger of the blockchain, where the IIoT device states were recorded at defined intervals to optimize storage and maintain traceability.

Blockchain Parameters:

- Consensus Mechanism: Notarization (ensures that transactions are validated).
- Data Structure: Batches of hashed device states are stored in a chain.
- Item Smart Contracts: Enforce predefined operational constraints (compliance) before transactions are allowed.

(d) ML-Based Predictive Auditing System

An ML-based auditing system is deployed to enable real-time anomaly detection. The system uses a Random Forest with time-series analysis to classify device states into NONE, LOW RISK, MEDIUM RISK, or HIGH RISK categories.

- Training Data: Generated IIoT logs with labelled anomalies.

- Device-Movement Patterns.
- Sudden weight fluctuation.

4.2.2 Simulation Scenarios and Data Generation

The framework was evaluated under three simulated scenarios, each reflecting a different device compliance level in a warehouse environment.

(a) **Scenario 1:** Compliant Behavior (None or Low Risk)

This scenario illustrates standard operations under ideal conditions, where the IIoT-enabled reach truck forklift devices follow designated paths (X, Y, Z), carry authorized loads (W) measurements, and create transactions according to the established guidelines. It acts as a reference point for evaluating the system performance during regular operational circumstances.

- The ML-based predictive auditing system assigns the risk to be either at NONE or at LOW
- Smart contracts allow device state changes to be logged into CORDA.

Expected Outcome: Transactions are recorded compliant, and no security alerts are raised.

(b) Scenario 2: **Partially Compliant Behavior (Medium Risk)**

This scenario illustrates the IIoT-enabled reach truck forklift devices display noncritical deviations, such as slight changes in their routes or imbalances in loads, which are common in real-world operations. This scenario assesses the system's ability to identify and validate behaviors that are ambiguous or borderline and require more sophisticated decision-making.

- The ML-based predictive auditing system detects anomalous trends and flags the device either at MEDIUM or at HIGH RISK
- The smart contract temporarily blocks state changes until the issue is resolved.

Expected Outcome: Alerts are generated, and the blockchain peers to conduct investigations about the anomalies evident in the IIoT devices being monitored.

(c) Scenario 3: **Non-Compliant Behavior (High Risk)**

The IIoT-enabled reach truck forklift devices deliberately perform unauthorized activities such as tampering with sensor data, entering prohibited areas, or initiating unauthorized transactions. This scenario represents insider threats, device breaches, or cyber-physical attacks, testing the ability of the framework to withstand and maintain trust boundaries.

- The ML-based predictive auditing system decide the risk as either at MEDIUM or at HIGH level consistently.
- The smart contract prohibits state changes and triggers investigations by blockchain peers.

Expected Outcome: Device is blocked, and an audit log is generated, preventing security breaches.

To capture a broad array of use cases relevant to secure IIoT-enabled cloud manufacturing, these scenarios were meticulously selected to strike a balance between technical feasibility, behavioral authenticity, and threat representation.

4.2.3 Data Volume and Class Distribution

A total of 457,200 labeled records were generated from the simulation runs:

- Normal (compliant) behavior: 374,320 records.
- Partially compliant (grey-zone): 191,440 records
- Rogue/attack behavior: 191,440 records

Each record captured behavioral data and risk indicators, reflecting real-world IIoT data structures. Rogue behaviors were synthetically injected using rule-based violation patterns such as:

- Unauthorized zone entry.

- Payload overloading beyond safety thresholds.
- Unstable or erratic navigation behaviors

These behavioral deviations were modeled based on industrial safety violation taxonomies [171], [172], ensuring semantic realism. For each experimental iteration, a randomly sampled test batch of 5000–7000 records was used to evaluate model consistency, generalizability, and traceability performance.

4.2.4 Evaluation Metrics

The performance of the framework was assessed using quantitative evaluation metrics. These metrics align with the core objectives of the system: secure operation, real-time responsiveness, traceability, and resilience.

(a) **Validation Accuracy (%)**

- Measure the correctness of behavior classification using the ML auditing model (true positives and true negatives).

Relevance: This study demonstrates the predictive capability of an auditing engine.

(b) **Precision and Recall (%)**

- Precision: quantifies the number of correctly identified flagged anomalies.
- Recall: measures the proportion of all anomalies that were successfully detected.

Relevance: These metrics assess the effectiveness of a system in proactive anomaly detection.

(c) **Traceability Rate (%)**

- Indicates the proportion of all actions (valid or invalid) that were recorded and verified on the blockchain.

Relevance: This reflects the transparency and visibility of IIoT operations across all scenarios.

(d) **Average Latency (s)**

- Capture the system's response time from event detection to logging or transaction resolution.

Relevance: Critical for time-sensitive industrial operations requiring real-time decision-making.

(e) **Transaction-rejection rate (%)**

- Reflects the system's ability to block unauthorized or invalid operations through smart contract enforcement.

Relevance: Gauge the effectiveness of blockchain-based validation under threat conditions.

(f) **System Uptime and Resilience**

- Observe whether the framework maintains continuous operation despite behavioral anomalies.

Relevance: Confirm the system's stability in the face of cyber-physical disruptions.

Together, these metrics provide a multi-dimensional view of performance, covering both detection capability and operational continuity.

4.3 Experimental Execution

4.3.1 Testing Procedure

1. **Forklift initialization:** Each asset is assigned a predefined path.
2. **ML model activation:** Real-time movement predictions were generated.
3. **Real-time anomaly detection:** Deviations were flagged based on predictive modeling.
4. **Blockchain validation:** Transactions committed to traceability.

5. **Risk classification:** Alerts were triggered based on the assigned risk levels.

4.3.2 Experimental Iterations

This study defines an experimental run as a complete execution cycle wherein an IIoT-enabled forklift is tracked through movement, classified through risk,and finally logged through a blockchain [150], [151]. Overall, there were 150 recorded experimental runs for the three scenarios.

Each run involves:

- **Data ingestion:** The system receives simulated location and weight data.
- **Risk classification:** The ML model predicts the next expected value and assigns different risk levels (NONE, LOW, MEDIUM, and HIGH).
- **Blockchain logging:** This framework verifies compliance and log-state changes.

Each run provides a self-contained evaluation of the system performance under the given conditions. Because the parameters (X, Y, Z, and W) are different physical quantities (coordinates and weight), which have different units and ranges, the normalization process of min-max was used for the visualization, so all parameters were scaled between 0 and 1. Therefore, we chose min-max normalization because this method helps retain the relative relationships that exist in the dataset [152] while mapping all values into a limited scope within the range of [0,1]. The normalization was computed using the following mathematical expression:

$$E = X_norm = X - X_min/(X_max - X_min)$$

Where X is the original value, and X min and X max are the minimum and maximum values of a given parameter, respectively. The dataset was independently normalized for each of the parameters (X, Y, Z and W)

4.4 Results and Analysis

The following sections present the results of an experimental evaluation designed to assess the efficacy of the proposed ML-Blockchain integrated framework in ensuring traceability, security, and anomaly detection in an IIoT-enabled industrial environment.

4.4.1 Scenario (1): Fully Compliant Forklifts

The primary objective of this study is to evaluate the accuracy of the system under normal operating conditions. This serves as a baseline for understanding how well the ML model predicts movements, and whether blockchain validation operates without conflicts when an asset follows the expected behavior.

(a) **Experimental Conditions**

- IIoT-enabled reach truck forklift devices were simulated to operate strictly within their predefined zones while adhering to all movement boundaries and weight threshold constraints.
- Real-time sensor data were continuously compared with ML-predicted values to assess the correlation accuracy using historical data to log the risks.
- No blockchain validation disputes were observed as all state transitions were aligned with the expected operations.

(b) **Operational Compliance and Risk Analysis**

In this scenario, the IIoT enabled the reach truck forklifts that were operated following the ongoing smooth trends reflected in the historical data. The ML model effectively predicted movement patterns with minimal deviations between the expected and actual sensor readings. The blockchain smart contract validates movement and weight compliance before state changes are permanently recorded. Figure 27 shows the normalized plots of the simulated values of X, Y, Z, and W, and the predicted values.

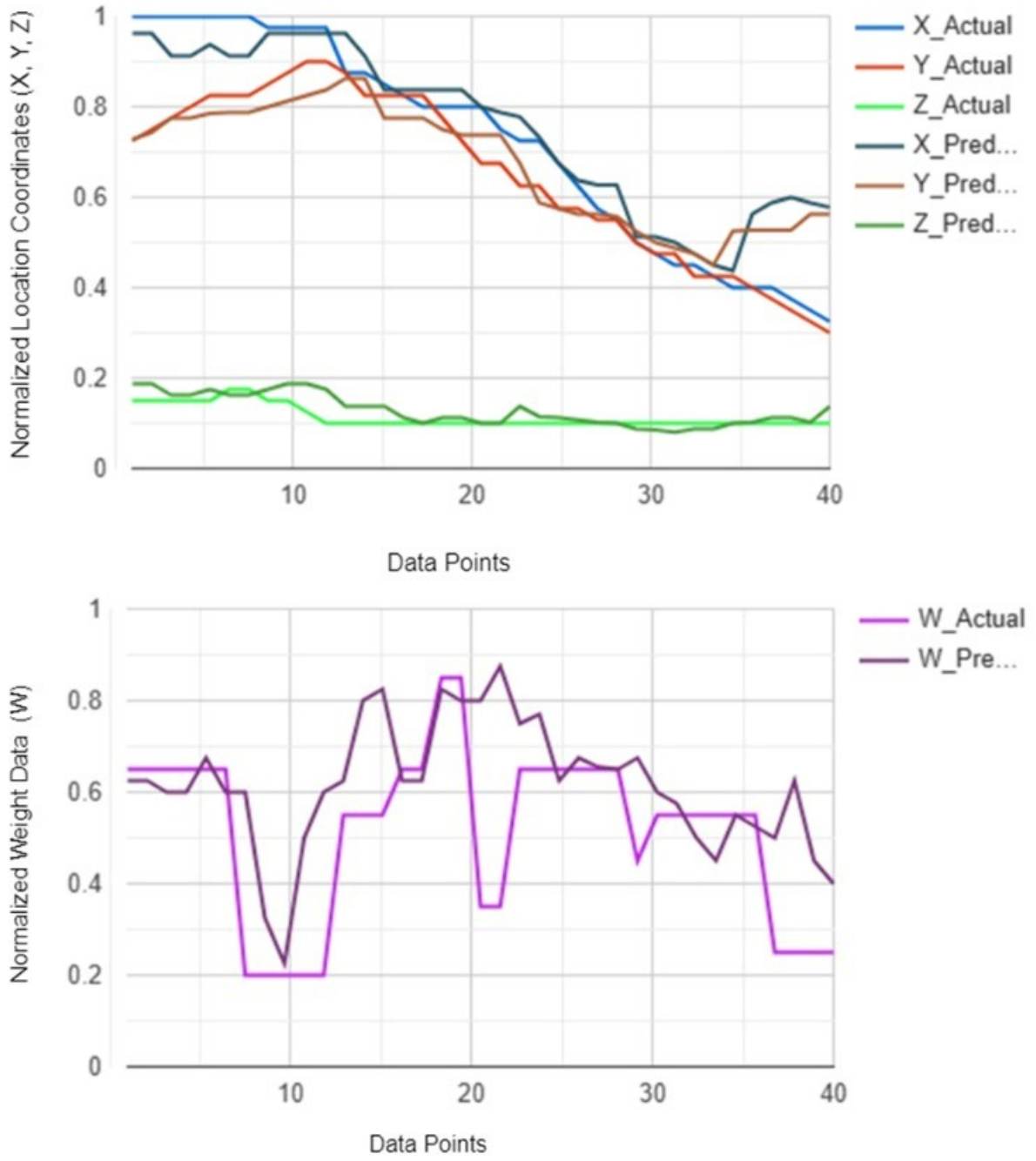


Figure 27: Actual versus Predicted tests for IIoT devices that are fully compliant to the smart contract rules.

As the historical data have 152400 records per asset showing smooth trends marked as "yes" as well as anomaly appearances marked as "no," smooth ground movements of the reach truck forklifts are expected to be predicted accurately. The plots in figure 27 for the fully compliant forklift scenario show a high degree of visual alignment between actual and predicted values. The two curves exhibit similar directional movements, rising

and falling in tandem, indicating that the model successfully tracks the overall behavioral pattern of forklifts operating within the expected norms. Correlation values (Corr X: 0.95; Corr Y: 0.93; Corr Z: 0.77; Corr W: 0.52) reflects co-movement but not how close the predicted values are to actual ones. However, the narrow gaps between the lines indicate that the model not only recognizes the trend but also achieves reasonable proximity in magnitude, reinforcing its reliability under stable, rule-compliant conditions. More definitive performance insights are provided through classification metrics, which directly evaluate the correctness of predicted class labels.

(c)**Risk Observations**

The risk classification module assessed all recorded movements and categorized them into four levels: NONE, LOW, MEDIUM, and HIGH. **Risk Distribution:**

- **NONE or LOW risk:** 94.3% of the recorded logs, indicating that the forklift followed the expected movements with high accuracy.
- **MEDIUM risk:** 4.6% of cases are primarily due to minor operational variations such as brief halts or temporary slowdowns.
- **HIGH risk:** 1.1% of cases are attributed to planned asset reallocation rather than an actual security violation.

The predictive auditing system conducted real-time risk profiling, and most recorded events were NONE or LOW risk levels, indicating minimal deviations from expected behavior such that the provenance flow transaction was initiated in the CORDA smart contract to update the state change in the smart ledger of the smart contract successfully.

Figure 28 illustrates a successful blockchain transaction related to a smart contract. The blockchain validation success rate was 100%, confirming that all the logged events were legitimate and secure. No unauthorized state changes occurred, reinforcing the transparency and trustworthiness of provenance records. In this scenario, the smart contract constraints allow it to accept the X, Y, Z, and W values of all the IIoT-enabled forklift devices within the acceptable range. Furthermore, the constraints allow only no risk or a low risk of transaction initiation. In this scenario, all forklift constraint values are

within the limits; hence, the transaction is executed successfully. Finally, the transaction is signed digitally.

Figure 28: Blockchain Validation success(experimentation).

(d)**Performance Metrics**

The performance of the predictive model based on the decision tree in the random forest algorithm was validated by plotting four performance parameters: precision, recall, F1 score, and traceability. The Figure 29 presents the metrics for performance of the prediction model in Scenario (1)

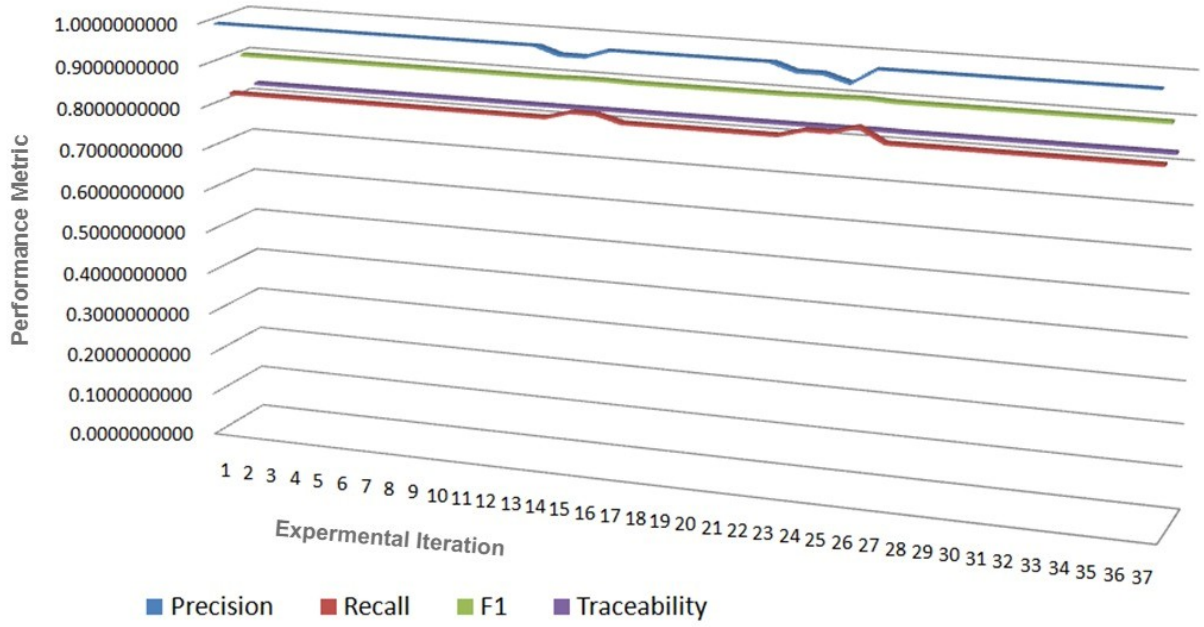


Figure 29: The metrics for performance of the prediction model of the Scenario (1).

The curves of the performance metrics follow linear straight lines without any noticeable

degradation in the performance. Precision was maintained at 100% in most cases except for a few instances, whereas recall and traceability were maintained at 82%. As recall represents the anomaly detection rate, an 82% rate indicates that the model is capable of detecting more than 80% anomalies. The F1 score formed a straight line of approximately 90%, and was maintained throughout the testing period.

(e)**Analysis and Interpretation**

The high correlation values for the X-axis (0.95) and Y-axis (0.93) movements confirmed that the ML model was highly effective in predicting the forklift trajectory and ensuring that the IIoT-enabled forklift devices in Scenario 01 followed predefined paths without deviation. However, the slightly lower correlation for the Z-axis movements (0.77) suggests that the vertical displacement was somewhat less predictable than the ground-level movements because they changed abruptly during warehousing operations. This could be due to slight variations in warehouse floor levels. In practice, risk levels do not change suddenly. Instead, as IIoT assets progress along their designated routes, the shift becomes more gradual. Consequently, experiments were performed by adhering to the paths outlined in the scenario. When data are inputted into a blockchain, blockchain peers should not immediately draw conclusions. They must closely monitor these patterns to assess whether genuine risk is present. ML's use of predictive analytics aids in minimizing unnecessary false positives. For instance, if there have been numerous instances of course changes or reallocations in the past, they will eventually be incorporated into predicted values. Consequently, significant variations between the predicted and actual values occurred only when unexpected outliers appeared in the data streams. For example, if an IIoT-enabled forklift has never been moved to a different zone, but has been removed for cooling or repairs multiple times before, the predictive values will only indicate a breach when the Z-value exceeds its usual operating range. Despite these minor variations, the blockchain validation process reinforced trust in the recorded data by confirming that all the logged events were legitimate. The results demonstrated that the IIoT-enabled forklift devices in scenario 01 maintained full compliance while highlighting areas where predictive accuracy could be further refined, particularly for weight and vertical movements. In addition, the performance metrics can be improved by increasing the size of the training dataset.

4.4.2 Scenario (2): Partially Compliant IIoT enabled Forklifts

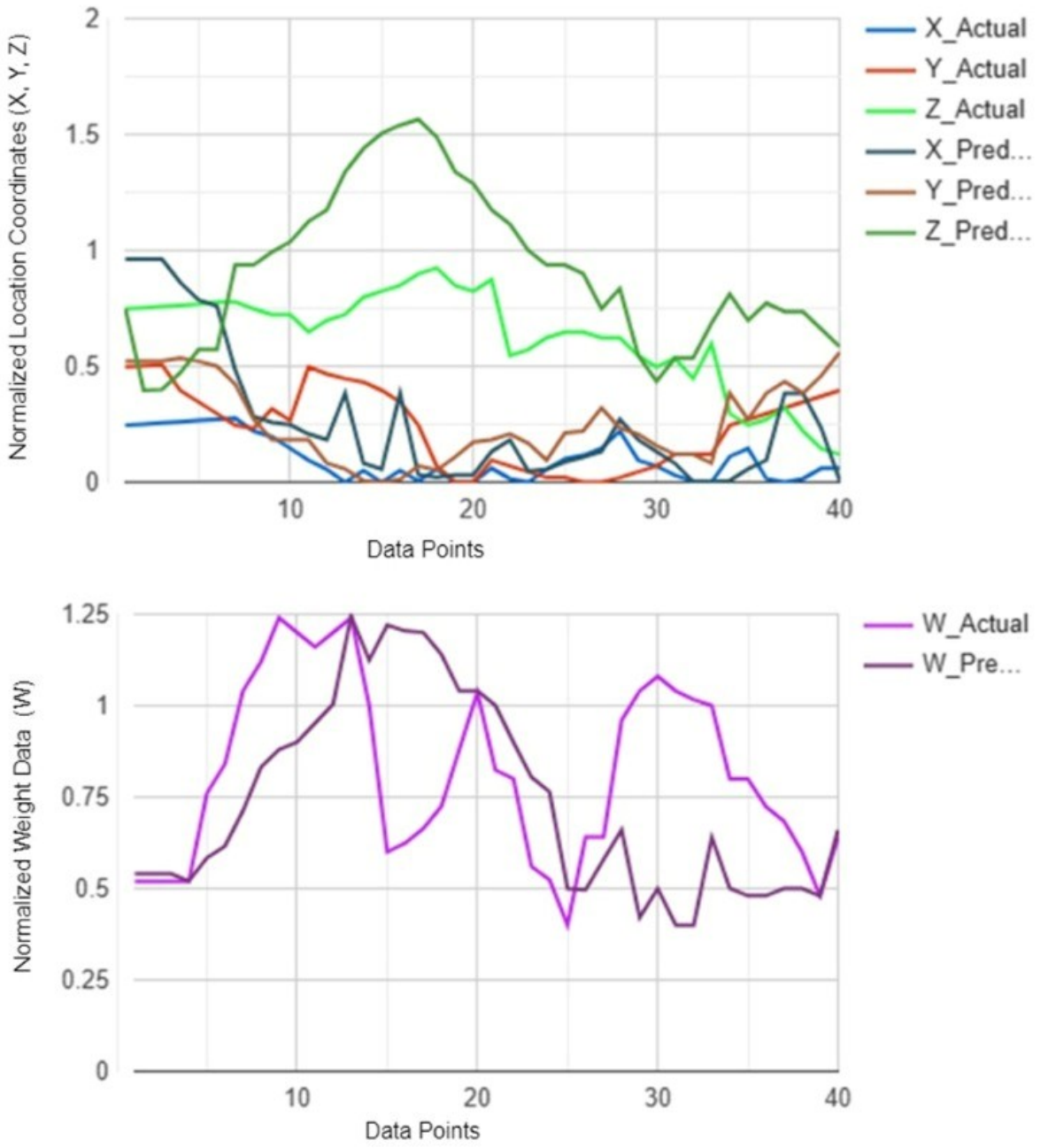


Figure 30: Actual versus Predicted tests for IIoT devices that are partially compliant but was suddenly assigned a radically different task in the smart contract not predicted by the ML.

This scenario evaluates the anomaly detection capability of the system when an asset exhibits minor operational deviations.

(a)**Experimental Conditions**

- IIoT-enabled forklifts occasionally moved outside their assigned range, but did not breach security policies significantly.
- Weight inconsistencies were observed because of occasional load misreporting.

In the partially compliant case, the graphs in figure 29 show a moderate trend similarity between the actual and predicted data. The model appears to capture the general direction of change, but with more visible deviations, indicating that the asset does not perform as per smart contracts. The graphs show instances in which the predicted values diverge significantly from the actual values, indicating a loss in predictive precision during inconsistent behaviors. Therefore, the correlation values (Corr X: 0.80; Corr Y: 0.37; Corr Z: 0.52; Corr W: 0.27) is useful in confirming that the model still captures general behavior patterns, but should be interpreted alongside other metrics.

(b) **Operational Compliance and Risk Analysis**

Scenario 2 was used to simulate inconsistencies, in which IIoT-enabled forklifts were simulated to divert from their intended paths. Such divergences may be the result of manual reassignment or route optimization, which is not reflected in warehouse management. The ML model attempts to predict movements and weight changes while continuously comparing them with real-time sensor inputs. Figure 30 presents a comparison of the actual and predicted values for the IIoT-enabled forklifts, illustrating the extent of deviations and their classification.

(c) **Risk Observations**

- o MEDIUM risk: 27.3% of recorded cases, primarily caused by lateral deviations from the expected movement paths.
- o HIGH risk: 9.6% of cases in which deviations significantly exceeded operational thresholds.

In this scenario, the IIoT-enabled forklifts are predicted to be partially compliant, but deviate significantly from the ongoing trend, thus causing a series of MEDIUM risks and two HIGH risk logs, in addition to the LOW risk logs expected from compliant IIoT assets. This was because the forklift changed its course from its predicted track several times, and intermittently returned to its original predicted path. The results indicate that unexpected lateral shifts (Y-axis) were the primary contributors to increased risk levels, aligning with the ML model's sensitivity to deviations from predefined pathways.

The predictive model identified several instances of MEDIUM risk primarily caused by deviations from the expected movement path. The IIoT-enabled reach truck forklift devices are partially compliant (i.e., have operational errors), and the predictive auditing system determines the risk as either at the MEDIUM or HIGH level consistently, such that the state change in the CORDA smart contract is prohibited, instructing the blockchain peers to conduct investigations on the anomalies evident in the IIoT devices being monitored.

(d) **Performance Metrics**

Figure 31 shows the performance of the model as reflected by the precision, recall, F1, and traceability metrics. The precision returned to 100%mfor the majority of the cases, except for a few where it was reduced to the range of 96%–98% depending on the severity of the deviation. The recall (anomaly detection rate) was approximately 82.1% in most cases, 83.7% in two cases, and 85.4% in one case. The F1 score returned to 90.2% in the majority of cases and 90.4% in some cases, indicating a high reliability in anomaly detection. This indicates very high confidence in the predicted values. Traceability returned to 82.1% in almost all cases. They reflect the consistency of the performance of the decision trees in the random forests algorithm.

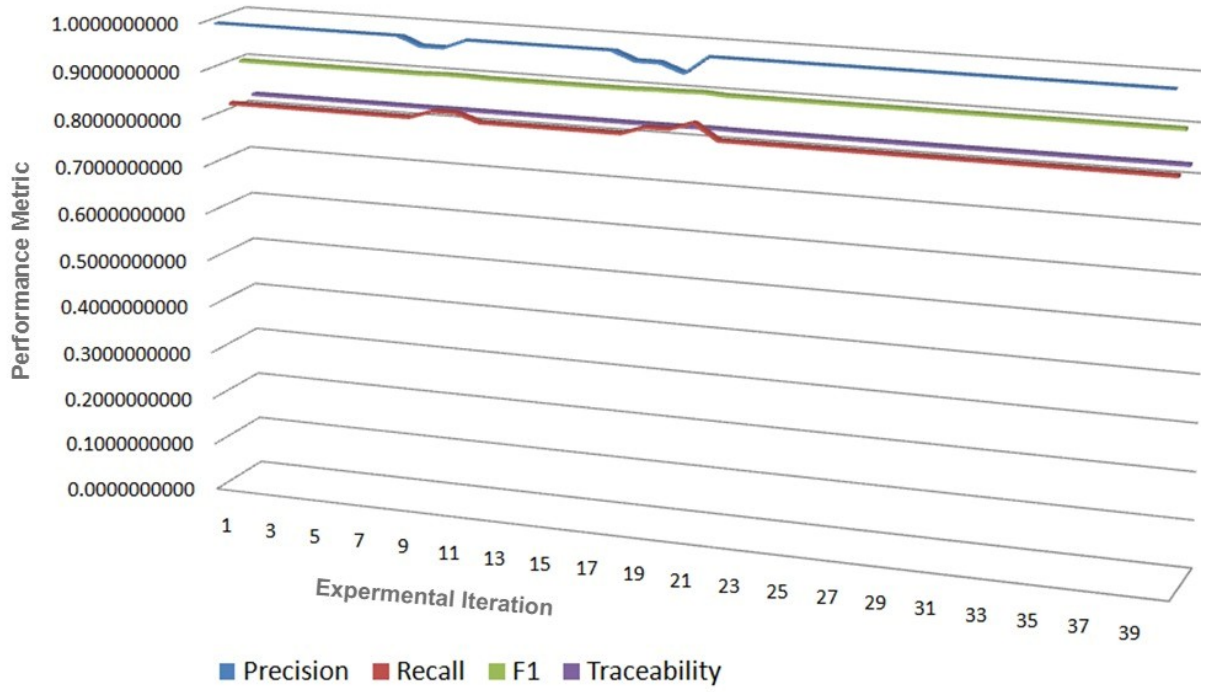


Figure 31: The metrics for performance of the prediction model of the Scenario (2).

(e) **Analysis and Interpretation**

The IIoT-enabled reach truck forklift device behavior reflects real-world scenarios in which forklifts are manually reassigned or undergo operational adjustments that are not immediately updated in the Enterprise Resource Planning (ERP) system. The correlation in Y-axis movements (0.37) was significantly reduced, suggesting unexpected lateral shifts possibly due to unplanned navigation changes, external obstructions, or human intervention. The Z-axis correlation (0.52) also indicated unpredictability in the vertical displacement, which could be due to varying load-handling patterns. The weight correlation (0.27) was significantly lower than that of the IIoT-enabled reach truck forklift devices in Scenario 1, suggesting frequent inconsistencies in the weight sensor readings, possibly owing to irregular load placements or inaccuracies in real-time weight capture. The variations in the Z and W data show that the forklift is routinely loading and unloading materials, although not as per the predictions, and also at locations not predicted by the ML. Despite these deviations, the anomaly detection model effectively classified irregular movements, achieving an F1 score of over 90% which confirmed a high confidence level in detecting operational deviations. The blockchain validation mechanism ensured that

all risk events were recorded and investigated by blockchain peers, as the smart contract routine highly deviated from the other routines that were reflected in the differences between the ML predictions and actual behaviors. There are significant differences between the actual and predicted values, indicating that the asset does not perform as per the smart contracts. In this scenario, the blockchain will refuse to update prompting for investigation and return the asset to full compliance with smart contract rules, reinforcing the framework's ability to track and analyze forklifts that do not strictly adhere to predefined movement paths.

4.4.3 Scenario (3): Compromised/Rogue IIoT enabled Forklifts

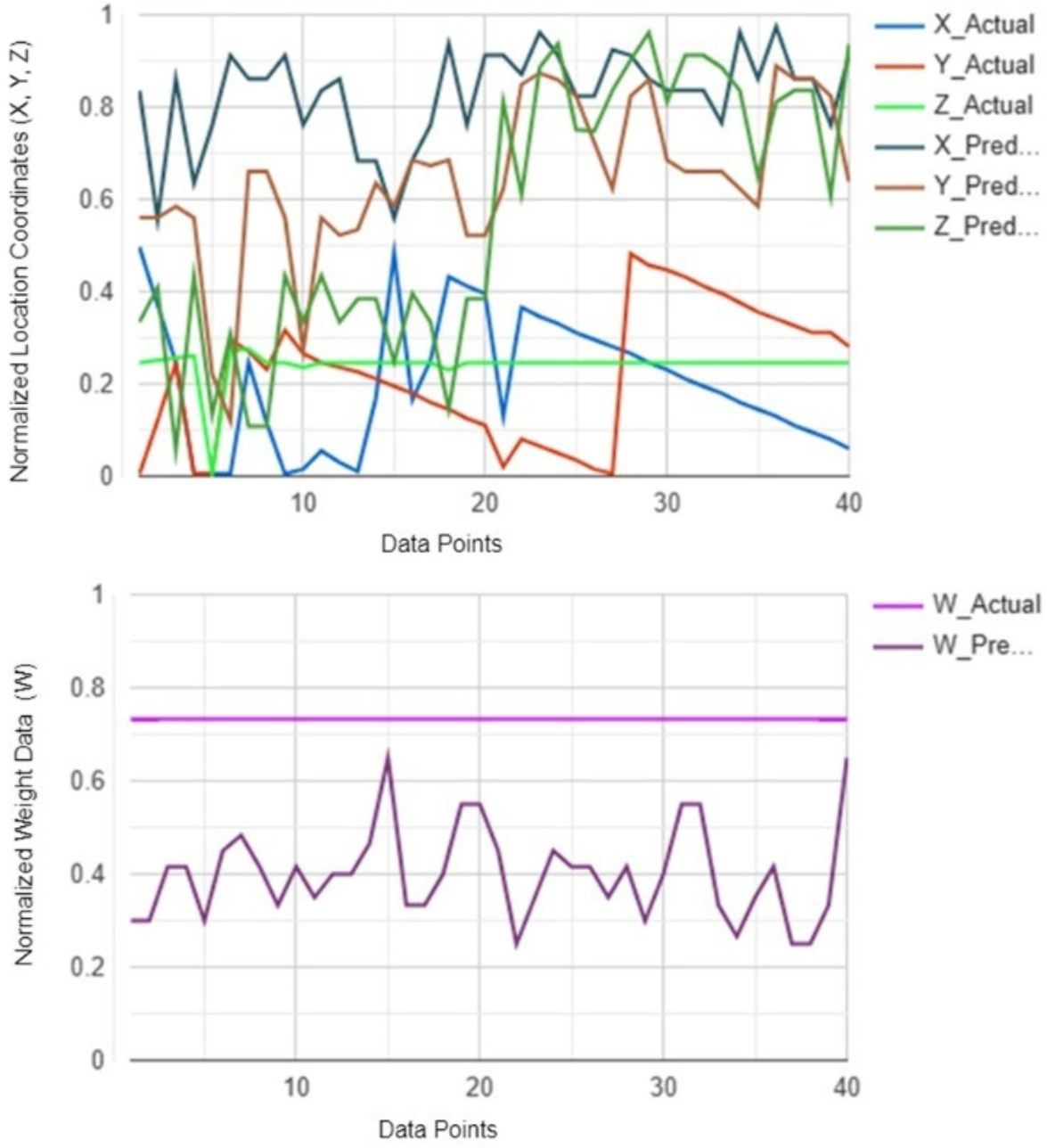


Figure 32: Figure 32: Actual versus Predicted tests for IIoT devices that were non-compliant to the smart contract rules.

This scenario evaluates responses to compromised or rogue forklifts.

(a) **Experimental Conditions**

- IIoT-enabled forklifts exhibit erratic movement beyond assigned zones.
- The assets failed to register the expected weight changes, indicating possible sensor tampering.

For the rogue scenario, the plots in figure 32 reveal visible divergence between the actual and predicted values during several anomalous intervals. This is also reflected in a lower correlation coefficient (Corr X:-0.02; Corr Y: 0.15; Corr Z; 0.20; Corr W: 0.20), indicating reduced directional alignment due to unpredictable or adversarial forklift behavior. In this case, the correlation coefficient signals only a partial trend relationship and should not be interpreted as evidence of an accurate prediction. Rather, it highlights the inherent challenge of predicting compromised behavior and the need for additional indicators to fully assess model trustworthiness under anomalous or hostile conditions.

(b) **Operational Compliance and Risk Analysis**

Scenario 3 was simulated as rogue or compromised devices that followed a path completely different from the predicted path. Unlike scenario 2, which indicated operationally permitted deviations, The IIoT-enabled reach truck forklift device movement was erratic, crossing predefined safety zones and ignoring loading constraints and inconsistencies in sensor readings. The ML-based predictive auditing model continuously monitored the forklift state and the blockchain-based system captured every deviation in real time, ensuring that deviations could be flagged for immediate investigation.

(c) **Risk Observations**

- MEDIUM risk: 18.2% before escalating.
- HIGH risk: 81.8% sustained throughout the experiment.

The rapid escalation from MEDIUM to HIGH risks demonstrated the effectiveness of the framework in detecting security threats. Initially, the risks started with the MEDIUM logs. However, as the IIoT-enabled forklifts continued to operate outside the expected

parameters, the risk assessment model escalated the classification to HIGH risk, which persisted until the end of the test, except for the last log in the MEDIUM test. This scenario returned mostly MEDIUM and HIGH risks because of significant differences between the actual and predicted values. It may be noted that this forklift did not change any weight or z-axis movements (indicating loading and unloading of weight) but simply moved randomly in the warehouse. The CORDA does not accept this level of risk for this smart contract, and hence, has rejected the transaction instructing the blockchain peers to conduct investigations about the anomalies evident in the IIoT devices being monitored throughout this scenario, as shown in Figure 33.

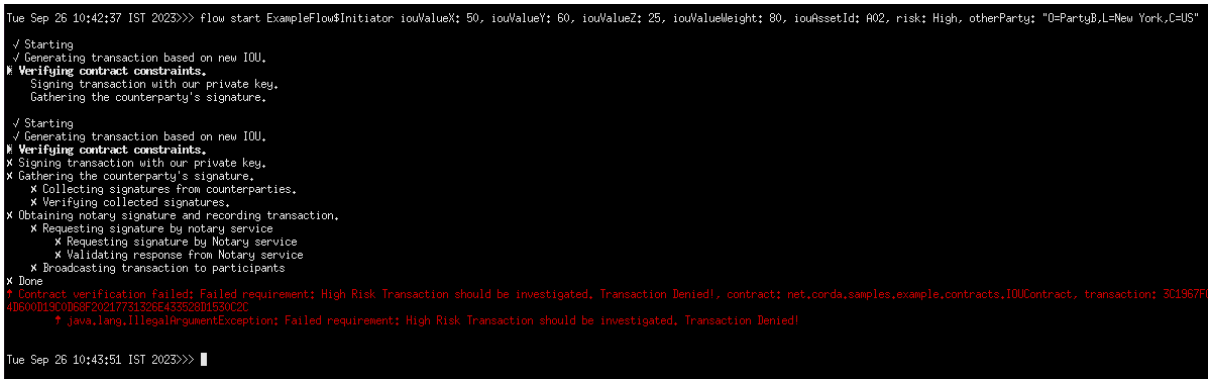


Figure 33: A rejected medium and high risk transaction (Experiments).

Scenario 3 exhibited significantly higher risk levels than scenarios 1 and 2. This sustained high-risk classification indicated severe compliance failure, such that the state change in the CORDA smart contract was prohibited, warranting immediate intervention.

(d) **Performance Metrics**

The precision, recall, and F1 scores reflected high confidence in the predicted values. The precision was 100%, except in two cases. The recall was 0.82 in the majority of cases, except for five cases when it reached a higher value of 0.83. The traceability was also 0.82 for the majority of the cases, except for three cases where its value increased to 0.83. The latency of the model in generating the results for these cases is 3–5 s. This performance is expected to improve powerful hardware resources in cloud computing.

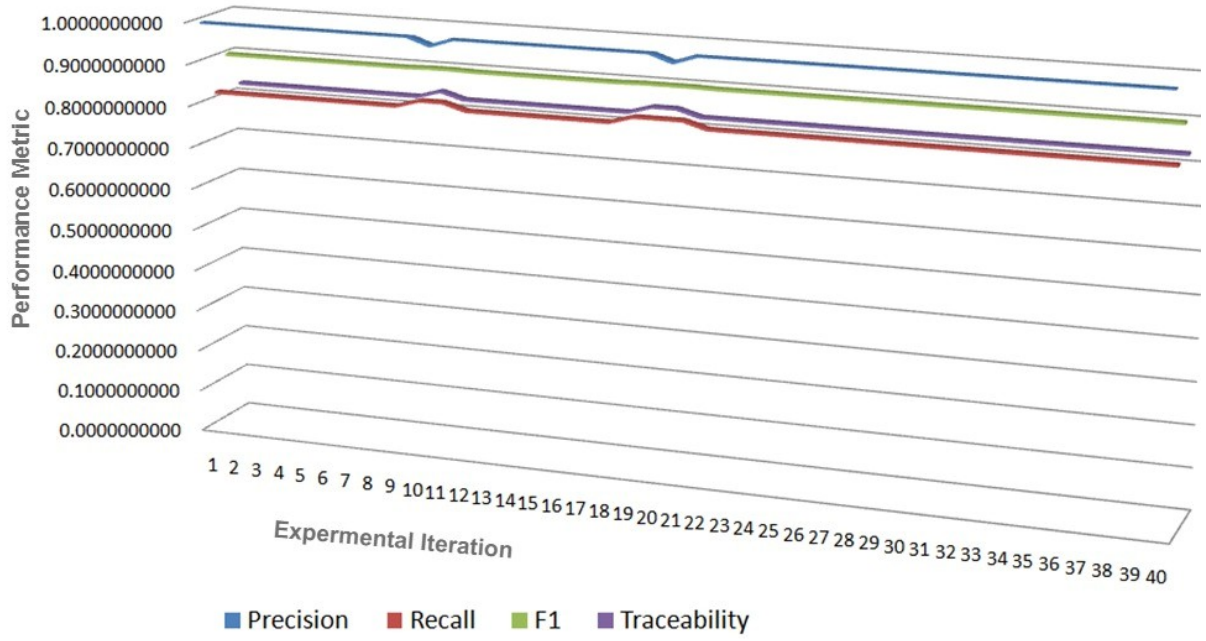


Figure 34: The metrics for performance of the prediction model of the Scenario (3).

(e) **Analysis and Interpretation**

The erratic movements of IIoT-enabled forklifts followed a path completely different from the predicted path. The lower correlation values across all movement dimensions, particularly the Y-axis (0.20) and Z-axis (0.15), indicate highly unpredictable behavior that deviated significantly from both historical trends and ML-predicted values. The weight correlation (0.12) further suggested unreliable load data, reinforcing the need for thorough inspection and remediation.

This behavior reflects two situations.

- **Cybersecurity Threat:** The forklift's unexpected movement patterns could indicate unauthorized control by an external entity, suggesting a cybersecurity breach. This indicates that the forklift is compromised and is used as a weapon by the hacker.

  This behavior indicates that the asset may be under the control of an insider trader who bypasses the industrial connectivity with these assets. Such behavior may also be possible if the asset has been compromised by an external attacker over the Internet by eavesdropping on the connectivity between it and the cloud manufacturing

controller.

- **Another possibility is that the Z movements and loading/unloading capabilities of the asset may have generated a fault**, thus causing it to move only in the Z- and Y-directions but not executing any warehousing operations. The forklifts did not offload or pick up any weights. It swayed across the warehouse, resulting in damage to the materials, warehouses, and loss of life. This could be owing to sensor faults, connectivity issues, or mechanical failure, which require immediate maintenance to prevent further operational disruptions.

  The blockchain validation mechanism can identify risks and conduct urgent interventions for both situations by ensuring that all non-compliant events are securely logged and flagged for peer review. The predictive auditing model flagged forklifts to warehouse operators to further investigate before the assets caused further disruptions. This scenario highlights the need for the incorporation of real-time monitoring, ML-based anomaly detection, and a blockchain-based auditing system for enhancing security and traceability in IIoT enabled warehouse settings

4.4.4 Cross-Scenario Analysis

(a) **Framework Validation and Security Mechanisms**

The IIoT-enabled foklifts in scenario 1, scenario 2, and scenario 3 demonstrated satisfactory compliance, unsatisfactory compliance, and the behavior of a compromised or faulty asset, respectively. The proposed ML blockchain framework consistently adhered to the guidelines and reliably generated one of three scenarios, (a), (b), or (c), demonstrating strong ML performance without encountering any runtime failures in any of the experiments. Blockchain peers can use prediction data with confidence when monitoring the compliance performance and behavior of assets. The latencies for each scenario are shown in Figure 35.

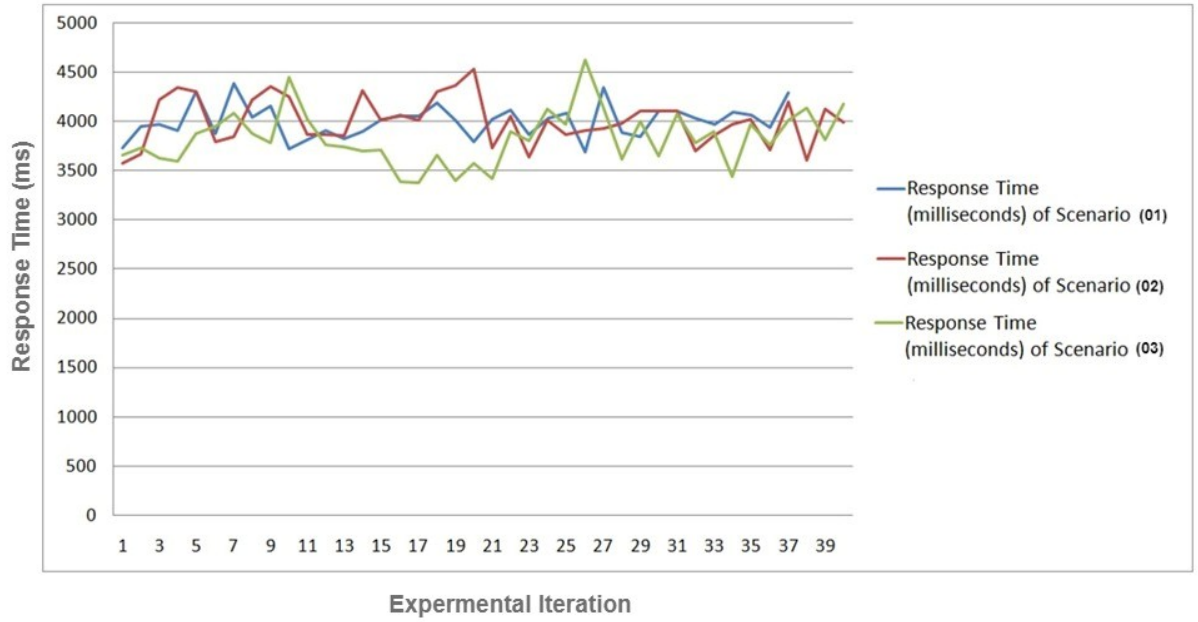


Figure 35: Response times of the three scenarios.

As shown in Figure 35, the validation latency varies significantly across behavioral scenarios, reflecting how the system adapts to context-sensitive security demands.

- In Scenario 1, which involved fully compliant forklifts, the blockchain confirmation time was minimized. This efficiency was attributed to the regular and anticipated behavior of the devices, which necessitated only lightweight validation. Smart contracts require minimal peer interaction due to low perceived risk and predictable

operational patterns, thereby facilitating rapid enforcement [158].

- In contrast, Scenario 3, which involved rogue or compromised forklifts, exhibited the longest validation latency. This increase can be attributed to heightened scrutiny, as the CORDA framework necessitates multi-party consensus across validator nodes prior to executing enforcement actions such as device isolation or transaction roll-back. This is consistent with CORDA's notary-based consensus model, which is designed to ensure transaction finality amid uncertainty [160].

In all operational scenarios, the average end-to-end latency, from data ingestion and anomaly detection to contract execution and ledger commitment, ranged between 3.5 and 4.5 seconds. This performance was achieved by implementing three architectural strategies.

- Off-chain storage of bulk IIoT data minimizes the volume and complexity of on-chain transactions [164].

- Selective smart contract invocation logs only policy-violating events, drastically reducing the transactional load on the blockchain [161].

- Decoupling of AI-based inference from blockchain consensus cycles—by executing behavior scoring through an independent Random Forest module—thus isolating computational delays from ledger write operations [162].

The findings affirm the viability of near-real-time blockchain-integrated auditing within Industrial Internet of Things (IIoT) systems under simulated industrial conditions. Nonetheless, it is crucial to acknowledge that this does not resolve the latency challenges inherent in distributed ledger technologies (DLTs). Instead, the results indicate that deliberate architectural decisions, lead to significant latency reductions within defined IIoT domains [163]

(b)**Comparative Risk Observations and Anomaly Escalation**

A progressive increase in risk severity was observed across the three scenarios, highlighting the ability of the system to adaptively classify operational anomalies based on predefined security parameters.

Table 3: Escalation of Risk Levels across Scenarios

Scenarios	NONE Risk	LOW Risk	MEDIUM Risk	HIGH Risk
Fully Compliant	78%	20%	2%	0%
Partially Compliant	15%	20%	55%	10%
Non-Compliant	0%	0%	22%	78%

From this progression, it is evident that the framework effectively categorizes routine operational variances (Scenario1) as non-threatening, while escalating genuine deviations (Scenario 2 and Scenario 3) to appropriate intervention levels. The sustained HIGH risk classification in Scenario 3 confirmed that the system was robust in distinguishing between acceptable and security-critical anomalies.

Implications for Practical Deployment

- Scenario1 required minimal intervention, validating that the framework can autonomously handle compliant assets without unnecessary alerts and reduce the operator workload.

- Scenario2's risk profile indicates the need for real-time adaptive learning, as its deviations, while non-malicious, still require oversight.

- Scenario3's sustained HIGH-risk categorization justifies the necessity of automated security intervention mechanisms such as asset isolation, blockchain-triggered alerts, and cybersecurity audits.

The ability to effectively classify and escalate risks ensures that the framework does not burden the monitoring system with excessive false positives while ensuring that critical security threats are prioritized. The determination of trust in risk logs should ultimately reside with blockchain peers, who are responsible for evaluating risk values in conjunction with other available data such as reallocations documented within the ERP system. Typically, a blockchain peer may occasionally encounter false positives, resulting in the recording of NONE and LOW risks during the process of updating smart contract states. Blockchain peers may establish criteria to adhere to prior to encountering actual

MEDIUM and HIGH risks. Given that data flows and risk logs are documented for each event when the reach forklift alters its weight, either by picking up or offloading, the sporadic MEDIUM and HIGH risks can be disregarded. For instance, a genuine MEDIUM risk might only be recognized after it has occurred consecutively, indicating a significant issue with the functioning of a particular IIoT asset. If a pattern emerges without apparent self-correction, false alarms may arise because of factors such as communication breakdown between the asset and monitoring system. When ongoing patterns of MEDIUM and HIGH risks are identified, blockchain peers may associate them with the risk logs of other IIoT assets linked to the same smart contract and seek explanations within ERP. If genuine MEDIUM and HIGH risks are identified as trends, they cannot be recorded in a smart contract because of restrictions on state changes. In such instances, blockchain peers must collaborate with the operations team to examine the risks. It is essential to document the response and mitigation procedures to clearly define the roles and responsibilities of such actions. These procedures can be integrated into smart contracts during the project-planning phase. Nevertheless, because actions are initiated following successive risk logs marked as MEDIUM and HIGH, this allows for the identification and management of risks before significant harm occurs. For example, an audible emergency alarm can be activated, prompting warehouse evacuation. In addition, the power supply to the vehicle's remote controller may be cut off to prevent the asset from continuing operation.

(d) **Performance of the Predictive Auditing Model**

The performance of the prediction model based on the decision tree in the random forest algorithm was validated by plotting four performance parameters: precision, recall, F1 score, and traceability. In Scenario (1), the curves of the performance metrics followed linear straight lines without any noticeable performance degradation. However, the performance metrics in scenarios (2) and (3) display similar performances, as shown in Figures 31 and 34, respectively. Apart from a few dips, the precision remained 100% in both the test scenarios. Recall and traceability maintained steady performance of approximately 82%. The F1 score maintains a consistent performance of 90% approximately. As Scenarios (2) and (3) were generated to simulate major differences between the predicted and actual values, only confidence in the performance of the predictive

auditing algorithm can generate confidence in the risk logs for blockchain peers to take intervention actions. To complete the analysis, the overall F1 score, precision, recall, and traceability were calculated using the weighted average method.

$F_{1_o} = \frac{\sum_{i=1}^{c}({}_i f_1^i)}{\sum_{i=1}^{c} {}_i}$

Where

- ${}_i$ is the number of test cases in scenario
- $f_1^i$is the average F1 score for scenario ,
- $c$is the total number of scenarios (in this case, 3)

By applying the weighted average formula for the overall F1 score in this study, the overall precision was calculated as 0.997630335, overall recall as 0.823801011, overall traceability as 0.821805688, and overall response time as 3952 ms. The overall accuracy of the model was 94%. The performance will improve significantly beyond the levels demonstrated in this study by expanding the training dataset to more than a million records. This will increase the resource requirements significantly but will increase the validity of risk logs in practical applications.

Table 4: Mapping scenarios to the research questions

Research Question	Scenario Contribution	Key Metrics	RQ Outcome
RQ1	All Scenarios (1–3)	Traceability Rate, Latency, Rejection Rate	Traceability was maintained at or above 82% across all behaviors. Even rogue transactions were logged and made verifiable. CORDA's design enabled fast and accountable provenance logging.
RQ2	Scenarios 2 & 3	Precision, Recall, Validation Accuracy	The predictive auditing model achieved high precision and recall across anomalous and malicious cases, with immediate smart contract enforcement. The system performed automated detection and mitigation without manual oversight.
RQ3	All Scenarios (1–3)	All Metrics	Data integrity was never compromised; all threats were detected and blocked. System uptime was uninterrupted. Even during attacks, traceability and logging continued, supporting secure and autonomous operations.

Summary

This section presents the evaluation and results of the proposed predictive auditing framework, demonstrating its effectiveness in detecting anomalies, ensuring traceability, and logging real-time risks using a blockchain. The framework showed high precision in identifying risks across three different asset behavior scenarios. Blockchain integration further strengthens data immutability and accountability, ensuring that peers can trace all the events and take action when necessary. Despite some limitations, the system shows promise for real-world applications in cloud manufacturing environments, particularly where continuous monitoring and traceability of IIoT-enabled CPS are required. In real-world implementations, the provenance blockchain solution is expected to mature with continuous learning using the machine learning algorithm to such an extent that it can predict a rogue IIoT device operating as a standalone or in a group of devices with significant confidence. The provenance will not only be based on static records but also on the history of data blocks transmitted by them. The system will perform predictive auditing based on behavioral analysis of the IIoT devices, keeping a close watch on the data blocks transmitted by them. The next chapter discusses the findings of this study and expands its use in broader industrial applications.

Chapter V

Discussions

5.1 Introduction

This chapter delves into the interpretation and implications of the proposed ML blockchain framework, focusing on its role in addressing the key challenges of enhancing security, visibility, traceability, and operational continuity within IIoT-enabled C-MFG environments. The chapter is organized on the following key aspects: Interpretation of Results; incorporation of the three main research questions to ensure that the interpretation results point to a cohesive and coherent narrative that builds the research; comparison of the results with the findings of past research; and implications of the findings for theory, practice, and policy.

5.2 Interpretation of Results

Research Question 1: How can a blockchain-based provenance tracking be optimized to support continuous visibility and traceability of IIoT-enabled Cloud Manufacturing (C-MFG)?

The first research question sought to determine how blockchain-based provenance tracking could be expanded to support continuous visibility and traceability in IIoT-enabled C-MFGs. The study findings validate that blockchain, when combined with off-chain storage and ML-based provenance data validation, significantly enhances real-time track-

ing of IIoT device interactions. Unlike the existing provenance blockchain framework, which relies on a one-time validation of device states, the proposed framework enables the continuous monitoring of all operational transitions within C-MFG environments. This is achieved through an ML-powered predictive auditing algorithm to continuously analyzes real-time IIoT data streams to detect patterns, deviations, and potential security violations. The model dynamically adapts to new behavioral data, improving its risk assessment accuracy over time. Deviations between actual operations and predicted behavior—based on smart contract rules—serve as reliable indicators of system anomalies or quality breaches. This mechanism ensures that the manufacturing processes remain fully traceable and any unauthorized modifications are immediately detectable. Furthermore, by extending provenance tracking beyond individual devices to capture interactions across multiple cloud-based manufacturing nodes, the framework enhances system-wide visibility, thereby enabling seamless traceability across distributed production networks.

Research Question 2: How can predictive auditing be integrated with blockchain technology to enhance real-time anomaly detection and proactive threat mitigation?

The second research question examined how ML-driven predictive auditing can be integrated with blockchain-based provenance tracking to improve real-time anomaly detection and proactive threat mitigation in IIoT environments. The ML-driven predictive auditing system was engineered to update the provenance database and segment it into datasets specific to each device. This segmentation facilitates the prediction of the future operational characteristics of each device. These predictions can subsequently be compared with the current state of the devices to identify any operational violations and record potential risks in the risk database. The study confirms that predictive analytics, particularly by dynamically evaluating IIoT device behavior and categorizing activities into low-, medium-, or high-risk levels, enhances the accuracy and responsiveness of security monitoring. Moreover, the integration of predictive analytics with blockchain smart contracts enables real-time security enforcement and prevents the execution of high-risk operations. Unlike traditional post-event forensic analysis, which only detects threats after an attack has occurred, this proactive approach allows for immediate threat mitigation and ensures continuous security monitoring in C-MFG environments. Evaluation metrics substantiate the effectiveness of this integration. Precision 96% to 100% confirms

the system accurately detected deviations from the predicted forklift behavior, such as anomalous lateral shifts or unexpected halts. The Recall 82.1% - 85.4% demonstrates the system was able to identify most of the anomalous events, with a high detection rate for genuine threats or irregularities. The F1 Score 90.2% - 90.4% reflects these findings by demonstrating a strong balance between precision and recall, indicating that the anomaly detection system maintained high confidence in its predictions while minimizing false positives. These results validate that the integrated framework provides robust anomaly detection while ensuring uninterrupted, secure IIoT operations.

Research Question 3: How can integrating blockchain and predictive auditing ensure data integrity, traceability, and security in IIoT-enabled C-MFGs while maintaining operational continuity?

The third research question explores how ML and blockchain technologies can be integrated to ensure data integrity, accountability, and operational continuity in dynamic Cloud Manufacturing environments. The research findings demonstrated that this integration provides a robust security architecture that outperforms traditional security models in terms of real-time responsiveness. This was achieved through a hybrid architecture that combined interval-based logging for batch processing, off-storage for capturing real-time data, and smart contract validation to ensure compliance. This approach minimizes latency while maintaining high data integrity such that transactions can be traced backward, and all accountabilities can be fixed. The incorporation of off-chain storage facilitates the continuous compliance monitoring of IIoT data devices to certain operating rules, whereas the processes carried out by the devices should be monitored by their dedicated process blockchains. Periodic on-chain hash updates ensure the integrity and immutability of records, without overburdening the blockchain with excessive data. Real-time visibility is realized through the continuous monitoring of IIoT devices, empowering dynamic, data-driven decision-making processes. The aggregation and hashing of all device states before logging onto the blockchain reduces the storage overhead while maintaining complete traceability, providing an auditable and tamper-proof record of device activity. Breaches of the operating rules were logged as risks, with full details of what had been breached.

Upon detecting possible operational and behavioral breaches in these devices, the system

triggers automated peer reviews, in which blockchain updates are executed to ensure that any detected risk is transparently logged and validated across the network. Such logs can not only invoke investigations, but also help in the reverse traceability of the operational compliance records of each asset registered in the blockchain and allocated to smart contracts.

The provenance logs in the blockchain can be analyzed to highlight the assets that operate with maximum compliance and those that default the most during operations for various smart contracts. To mitigate the performance overhead traditionally associated with blockchains, the framework incorporates dynamic consensus mechanisms adapted based on the risk levels detected in the system. Thus, suppliers providing the most trustworthy assets can be prioritized using the provenance blockchain. All these analytics are based on provenance logs, which are fully transparent to all blockchain members. By adjusting the consensus approach according to the severity of the detected anomalies, the framework minimizes unnecessary blockchain transactions, optimizes resource utilization, and maintains the system responsiveness. These features collectively ensure that the framework can handle frequent datasets in real-time while maintaining high performance and scalability, even under varying operational conditions.

In terms of operational efficiency, the framework achieved a response time of 3.952 s, significantly outperforming conventional security models. In addition, the smart contract-based security enforcement mechanism ensures that unauthorized activities are blocked at the execution level, preventing disruptions to manufacturing workflows. These findings confirm that integrating blockchain with AI-driven predictive auditing not only strengthens security, but also ensures seamless operational continuity in IIoT-enabled C-MFG environments.

5.3 Comparison with Previous Research

The findings of this study align with and extend the previous research on IIoT security. Earlier investigations [35], [41]–[43] developed frameworks that required the exchange of keys with client systems, followed by their verification through authorization records maintained in the provenance blockchain. These studies have concentrated on the pro-

cesses of identifying, authenticating, and authorizing IIoT devices. However, they do not fully mitigate several IIoT issues, as highlighted in [15] and [19]. By contrast, this study extends beyond key exchange validation by incorporating continuous validation using machine learning(ML).

The proposed system enhances security by dynamically monitoring IIoT devices in real-time, detecting anomalies, and logging risks. This ensures that the security threats identified in [15]–[19] are addressed more proactively rather than relying solely on static authentication methods. The justification of how the proposed system mitigates these threats is detailed in Section 5.2.

Previous approaches, including static anomaly detection and standalone blockchain logging, have been criticized for their nonreal-time responsiveness and limited scalability. In many cases, blockchain is employed in post-event contexts, in some studies, to ensure data integrity. While it works well for static environments, it often fails in dynamic and time-critical applications, such as manufacturing or smart logistics. The inherent latency of processing and analyzing static datasets limits the detection of faults in real time [5]; thus, they fail to respond to critical issues when needed. This study addresses these criticisms by presenting a hybrid protocol that combines continuous real-time data monitoring and immutable blockchain logging.

To the best of our knowledge, no prior research has directly compared the scope of the proposed framework because of its unique integration of real-time data logging in a blockchain and predictive auditing mechanisms in IIoT environments. Although no direct studies are available within the scope of the proposed framework for comparison, the capabilities of the proposed framework are compared with those of existing commercially available anomaly detection models, including XGBoost, LightGBM, AdaBoost, CatBoost, and GradientBoost.

5.3.1 Ruiz-Villafranca et al. (2023): AMEC-IIoT System

Ruiz-Villafranca et al. [143] conducted IIoT threat detection in a multi-access edge-computing (MEC) environment. Their approach involved testing decision trees in a random forest using five standardized commercial machine learning models: XGBoost,

AdaBoost, GradientBoost, LightGBM, and CatBoost. The dataset was prepared by collecting data from a networked environment created for experimentation. In the network, data on normal behaviors and anomalies were generated and then mixed to form the training and test datasets (80–20 split), similar to this research. However, Reference [143] used significantly larger datasets comprising several million records. This study evaluated anomaly detection performance under various network security threats, including packet manipulation, network scanning, denial-of-service attacks, and HTTP errors. Given that this research focuses on detecting anomalies in networked logistics assets, the closest comparison is with the network scanning results in [143]. The performance metrics of the five commercial models used in [143] and that of the model developed in this study are compared in Table 3.

Table 5: Comparison of Model Performance with Reference [143]

Model	Precision	Recall	F1 Score	Accuracy	Response Time
XGBoost	0.92	0.99	0.94	0.9992	5,380,689
LightGBM	0.91	0.96	0.92	0.999	160,220
AdaBoost	0.92	0.98	0.94	0.9991	19,925,673
CatBoost	0.91	0.98	0.94	0.9991	2,404,382
GradientBoost	1	0.89	0.91	0.9994	10,550,021
This Research	0.997	0.82	0.90	0.94	3.952

The most significant advantage of this research lies in the drastically lower response time (3.952 ms) compared to commercial models, which ranges from 160,220 ms to over 19 million ms. This performance difference is attributed to the significantly smaller dataset size used in this research (152,400 records per asset) compared with the multi-million record datasets in Reference [143]. This real-time performance indicated that the proposed framework is an ideal candidate for scenarios in which rapid threat detection and mitigation are critical. The model used in this study had a lower recall rate, which resulted in a lower accuracy. However, the precision was better than that of the models, and the F1 scores were comparable, given that the models used in Reference [143] were commercial models.

5.3.2 Elmrabit et al. (2020): Machine Learning for Anomaly Detection

Elmrabit et al.[144] evaluated anomaly detection in IIoT environments through machine learning using several machine learning models trained on three publicly available datasets: CICIDS-2017 (approximately 2.5 million records), UNSW-NB15 (approximately 2.8 million records), and ICS (approximately 78,391 records). For the purpose of this research, a comparison was made only with the RF results of the Random Forests. However, it may be noted that [144] achieved the best results with random forests compared with the other algorithms tested. Table 4 presents the results of a comparative performance analysis.

Table 6: Comparison of Model Performance with Reference [144]

Model	Precision	Recall	F1 Score	Accuracy
UNSW-NB15	0.844	0.991	0.912	0.877
CICIDS-2017	0.997	0.997	0.997	0.999
ICS	0.929	0.972	0.95	0.928
This Research	0.997	0.82	0.90	0.94

In this comparison, the precision was better than that of the other models (except for CICIDS-2017). The recall was lower than that of the other three models, and the F1 score was lower than that of ICS and CICIDS-2017, but comparable with UNSW-NB15. Given that the models compared with are commercially available models that have used millions of records and that the model in this study was trained with only 152400 records per asset, the performance may be considered acceptable.

Despite these differences, the main strength of this study was its ability to conduct continuous monitoring using 117 test cases, each completed within seconds. In contrast, the comparative models from [143] and [144] relied on single-snapshot testing because of their extensive training-testing times. The provenance blockchain models studied in the literature lack continuous monitoring capabilities.

Continuous and real-time detection of anomalies is a pivotal advancement in this re-

search. The comparative models in [143] and [144] suffer from excessive training and processing times, making real-time monitoring impractical. The primary reason this research achieved real-time monitoring while maintaining competitive performance was the implementation of Java-based continuous memory purging in the Java Virtual Machine (JVM). This feature is widely used in high-performance commercial Java applications.

5.4 Contributions to Existing Knowledge

This section outlines the primary contributions of this study to the existing body of knowledge organized according to three research questions (RQs). Each RQ-guided contribution addresses the key theoretical and practical gaps identified in the literature on secure and resilient IIoT-enabled cloud manufacturing.

RQ1 – Optimizing Blockchain for Continuous Traceability

This research advances the state-of-the-art by developing a blockchain-based provenance tracking framework that ensures the continuous visibility of IIoT-enabled assets throughout their operational lifecycle in cloud manufacturing settings. Unlike traditional systems that use blockchain as a passive archival medium, this study introduces an active real-time provenance layer. Every IIoT-enabled Cyber-Physical System (CPS) device is registered either on- or off-chain, recording essential metadata such as identity, deployment zone, ownership, and configuration state. A key innovation is the provenance-linked authentication mechanism, a condition-triggered routine that dynamically validates device authenticity not just at onboarding but during runtime events—e.g., relocation, reassignment, and maintenance. This promotes sustained trust in the adaptive C-MFG systems. The system architecture utilized a layered CORDA-based blockchain with a dedicated provenance layer. Smart contracts serve as autonomous validators that authorize only devices with verified provenance records. This mechanism transitions operational metadata from off-chain status to on-chain verification, activating real-time traceability. The secure encapsulation of operational parameters, such as hardware specifications, permissions, and geofencing rules, is enforced through blockchain logs. Real-time traceability is further strengthened via MQTT-enabled IIoT, transmitting device states (load weight and GPS location data) to the blockchain. Access control aligned with ISO 27001 restricts data

modification for verified administrators. During experimental evaluations under varying behavioral scenarios, the traceability index consistently ranged between 0.82 and 0.83, even in rogue or partially compliant conditions. This confirms the robustness of the proposed provenance mechanism in maintaining operational visibility. This advances the literature on blockchain for IIoT by transitioning from passive archival solutions to active and adaptive traceability mechanisms [4], [10], [15].

These findings confirm the architectural feasibility of achieving continuous traceability with minimal data loss under idealized conditions. However, it is crucial to note that these results were derived from a controlled simulation environment with uninterrupted connectivity, deterministic event sequencing, and noise-free sensor streams.

Realistic Challenges in IIoT Data Quality: When transitioning from simulation to physical deployment, several well-documented challenges must be considered:

(a). **Sensor Drift and Faults:** Over time, sensors may exhibit calibration drift or hardware failures, leading to inaccuracies in reporting critical parameters such as position or load.

(b). **Environmental Noise:** Electromagnetic interference, mechanical vibration, or physical obstructions can degrade wireless signal integrity.

(c). **Network Instability:** In complex warehouse layouts or high-density node configurations, edge devices may experience packet loss or delayed up-links due to bandwidth contention or transient disconnections.

(d). **Clock Synchronization Issues:** Distributed time-stamping is essential for chronological traceability; however, drift among local clocks may result in misaligned event logs.

These constraints directly affect the framework's ability to maintain continuous, high-fidelity visibility and must be anticipated when generalizing findings beyond the simulated context.

RQ2 – ML–Blockchain Integration for Anomaly Detection

The proposed hybrid framework integrates a Random Forest-based predictive auditing engine with a blockchain to facilitate real-time detection of behavioral anomalies across IIoT-enabled CPS systems. The auditing module is trained on location and weight data to identify abnormal patterns such as unauthorized zone entry and excessive load irregular downtime, which may indicate malicious activity or misconfiguration. These predictive insights are dynamically coupled with blockchain-based smart contracts that automatically enforce conditional responses. Upon detecting an anomaly, the system isolates the device, alerts administrators, and logs the event immutably into the blockchain's risk ledger. This mechanism ensures not only real-time mitigation, but also forensic accountability. The audit engine supports both stream and batch data processing, tolerates asynchronous device behavior, and distinguishes among multiple threat types from internal sabotage to external cyberattacks. Empirical evaluations yielded a detection latency of 3.9 seconds, accuracy of 94%, and F1-score of 0.90, validating the responsiveness and precision of the module [11], [28], [29]. Every confirmed risk instance was recorded on a chain with timestamped metadata. This enhances transparency and traceability across multiple cloud zones, thereby enabling scalable and unified threat management. This contribution fills a gap in the literature, where blockchain is often treated as a passive storage layer rather than as a real-time threat intelligence enabler [11], [28], [29].

RQ3 – Secure, Traceable, and Continuous Operations

This study demonstrated that fusing blockchain with ML-driven predictive auditing yields a unified security framework that ensures data integrity, traceability, and operational resilience in IIoT-enabled cloud manufacturing. Blockchain facilitates immutable provenance and accountability, whereas the ML engine ensures proactive detection of anomalous behaviors based on the device performance history. Unlike static validation systems, this framework supports real-time behavioral validation. Device location and weight lifted is continuously assessed, and smart contracts enforce responses based on historical behavior, thereby enhancing system agility. If an anomaly is detected, preventive actions such as isolation or risk flagging are executed immediately, reducing the likelihood of system-wide failure. To validate this architecture, a multi zone cloud simulation was conducted using IIoT-enabled forklifts exhibiting three behavioral patterns: fully compliant, partially compliant, and rogue. Across these scenarios, the system achieved complete operational

uptime, detected all major anomalies, and maintained workflow continuity, even under threat conditions. In contrast to periodic audits, the proposed system provides continuous event-driven monitoring of the IIoT devices. For example, location breaches, overloads, or unauthorized asset redeployments are logged and isolated in real time without disrupting ongoing operations. This ensures quality assurance, cyber-physical risk management and workflow resilience. Furthermore, off-chain Provenance Database (ProvDB) was incorporated to facilitate advanced functions such as compliance audits, anomaly pattern analysis, and predictive maintenance. This makes the architecture extensible for future IIoT innovations in smart manufacturing [35], [42], [46]. This confirms the viability of real-time resilience, accountability, and trust in decentralized C-MFG systems, which are often lacking in traditional periodic validation models [30], [33], [34], [35], [42], [46].

Consolidated Evaluation: Addressed Concerns in IIoT-Enabled C-MFG As summarized in Figure 37, the proposed framework systematically addresses the seven critical challenges prevalent in IIoT-enabled cloud manufacturing.

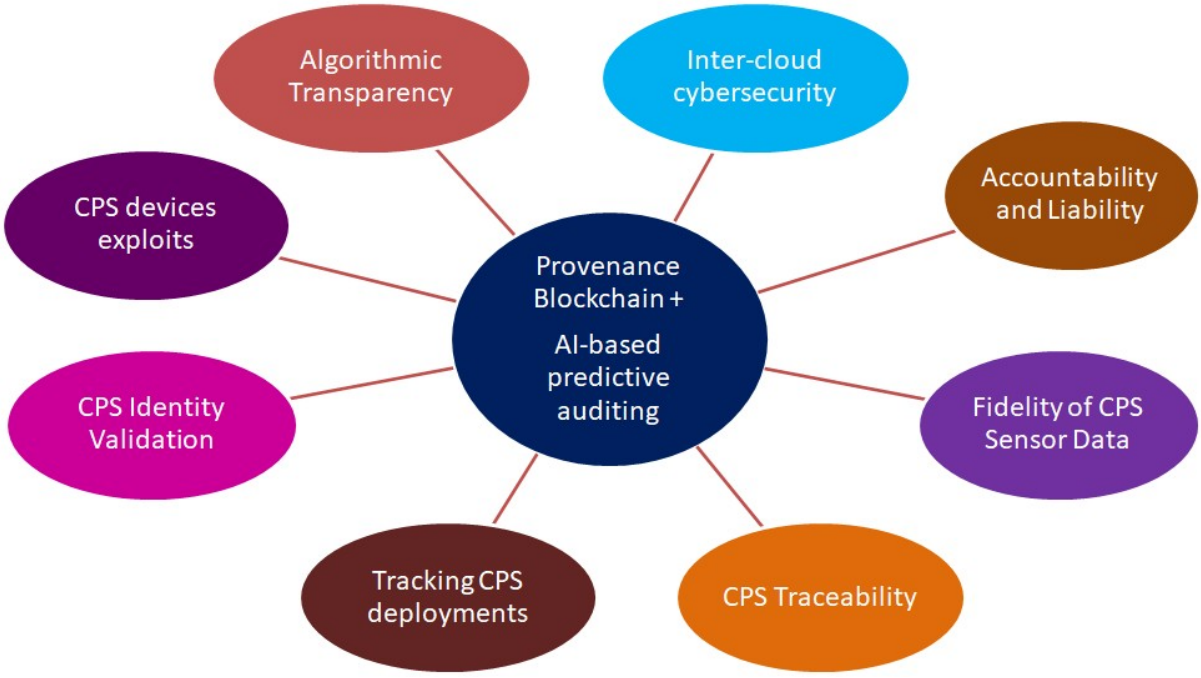


Figure 36: IIoT in cloud manufacturing concerns addressed by this study

(a). Validating the identity of CPS devices: The system registers CPS identities as on- or off-chain, with dynamic transitions enforced by smart contracts.

(b). Handling rapid deployment and scalability: Millions of CPS devices can be pre-registered and managed via smart contracts contingent on IT and bandwidth capabilities.

(c). Ensuring traceability during operational state changes: Predictive auditing continuously monitors device states and flagging and logging risks for blockchain validation.

(d). Sensor data fidelity: Although complete fidelity requires engineering knowledge, the system flags inconsistencies and assists operators through ML-based predictions.

(e). Accountability and ownership: Ownership data are captured at registration; smart contract parties and blockchain peers collectively enforce accountability.

(f). Multi-cloud cybersecurity: Security is ensured through a unified MQTT broker and a replicated ML infrastructure across cloud zones.

(g). Algorithmic transparency, which ensures the accountability of the algorithms employed to manage CPS device operations, poses a significant challenge. Advanced systems with a comprehensive understanding of their operational behaviors and performance metrics are required to monitor and control the performance and behavior of these algorithms effectively. Such systems can identify deviations in ongoing patterns and classify them as risk at various levels.

These concerns are comprehensively resolved through the integration of provenance tracking, predictive auditing, smart contracts, and blockchain-backed governance mechanisms, thus establishing a secure and resilient IIoT-enabled manufacturing infrastructure [2], [5], [10], [11], [28], [30], [35].

5.5 Implications of the Findings

The findings of this study have substantial theoretical, practical, and policy implications that have the potential to influence the future of the IIoT security frameworks. These findings provide deeper insights into the real-time performance of blockchain in IIoT systems, and highlight innovative ways to integrate blockchain with predictive auditing [147]. In this section, we elaborate on the theoretical, practical, and policy implications

of the contributions of this study.

5.5.1 Theoretical Implications

This study possesses significant theoretical importance as it advances the current understanding of blockchain technology, predictive auditing, and the Industrial Internet of Things (IIoT), particularly within the context of cloud manufacturing environments. By integrating these three areas, this study extends the theoretical framework of data provenance, security, and audit practices in the IIoT systems. This study offers a novel perspective on how blockchain and predictive auditing can be combined to create an effective continuous auditing system, which has been underexplored in the existing literature. The study:

(a). Expand the scope of the provenance blockchain's role in Industrial Internet of Things (IIoT) security to encompass not only the authentication and authorization of IIoT devices, but also the assurance of comprehensive operational compliance. Compliance should be regulated by rules established through smart contracts within a blockchain that is specifically dedicated to provenance.

(b). The theory of predictive auditing is advanced by examining the integration of machine-learning models with blockchain technology to detect and address operational anomalies in real time. This approach emphasizes the early identification of non-compliant or potentially compromised Industrial Internet of Things (IIoT) devices, thereby preventing any potential harm they may cause.

(c). This study advances the theoretical framework of IIoT system optimization by addressing issues of scalability and latency and by proposing practical solutions to mitigate these challenges, thereby enhancing the efficiency of blockchain-based systems in dynamic, real-time IIoT environments. Consequently, risk monitoring and control should extend beyond the mere introduction, reassignment, or decommissioning of devices. Instead, they should continuously monitor IIoT operations in real-time by utilizing predictive auditing capabilities. The proposed solution facilitates the monitoring and management of every variable associated with the IIoT devices linked to smart contracts.

These contributions are expected to shape future theoretical developments in the field by broadening the scope of blockchain's application in auditing and security beyond traditional approaches, fostering a more holistic understanding of how emerging technologies can optimize the operational and security processes in IIoT systems.

5.5.2 Practical Implications

From a practical standpoint, the integration of blockchain technology with predictive auditing in IIoT systems offers several benefits to industry stakeholders including manufacturers, auditors, and regulators. This study provides a pragmatic solution for some of the most pressing challenges faced by manufacturers and auditors in IIoT environments, particularly regarding real-time data validation, integrity, and automation. The practical implications of this study are as follows:

(a). Manufacturers will benefit from an enhanced ability to monitor and verify IIoT data in real time, ensuring the authenticity and reliability of critical data collected from sensors and devices.

Blockchains collect insights from supply chains using IIoT sensor transmitters for auditing and audit analysis [54], [59], [62]. As blockchain stores smart contracts, events related to the completion of contractual terms can be captured by field IIoT sensor transmitters. Field IIoT sensors with transmitters must be attached to the process data points identified in the assets assigned by the parties contributing to the smart contracts of the blockchains. At the engineering level, IIoT sensors and transmitters are digital logic electronic boards with integrated physics sensors that are capable of capturing and organizing the progress data of the parameters of interest in the process events being executed. The progress data can then be transmitted to big databases on cloud computing from where they can be analyzed and transformed into state change information fed to the smart contract tracker of the block chains [39], [44]. This leads to better decision making, optimized operations, improved compliance with industry standards, and greater transparency and traceability of manufacturing processes [6]–[9], [53], [54]. Security and privacy controls can result in trust and confidence between manufacturers and customers in cloud manufacturing [16]–[19] [28]–[30] [31], [32]. Blockchain peers should have

access to the logs of security events and enterprise risk databases to provide insight into the security exposure and risks related to the current state of events to all interested parties plugged into the blockchain. In this way, stakeholders in smart contracts can be informed in real time about the risks and exposures in industrial systems that serve their smart contracts.

(b). Auditors experience a transformation in their auditing processes as they can utilize a proactive approach to anomaly detection and audit execution. Instead of traditional manual audits, auditors can continuously monitor IIoT systems, enabling the early identification of issues and more accurate audits with less reliance on human intervention.

Predictive auditing (also called Auditing 4.0) is a fully automatic auditing system that uses Industry 4.0 [33]-[45]. It is an automatic mechanism for rapid data collection and time-stamped serial analysis using an automatic auditing mechanism for building transparency in manufacturing logistics and supply chains [62]. In traditional supply chains, transparency is limited to electronic data interchanges between collaborating parties. In digitized manufacturing and supply chains supported by blockchains, transparency is ensured through multihop tracking and tracing of signatures/fingerprints/raw data left behind by these processes. The framework enables multi-hop accountability established through blockchain monitors, registrars, and approvals. Traditional auditing in the industry is conducted manually by selecting samples from static data following a sampling process [37]–[39]. A small representation of the overall dataset was drawn to analyze the process and compliance gaps. The traditional auditing method was driven by expert auditors having "eyes for detailing" in the sample dataset and in the observations made about the environment in which the data were generated by the processes and stored. The auditing was completely driven by the auditor's evidence-based interpretation. Furthermore, auditors were mostly dependent on the declarations made by the audited organizations, which comprised sampled data pulled out from the running systems. Most auditors are detached from the transactional systems. The outcomes of the auditing process were driven by human expertise and experience. Hence, different auditors may reach conclusions based on individual analyses and interpretations

of the sample dataset and their observations [37]–[39]. The traditional auditing method recommends corrective and preventive actions, and related observations for improvement. However, audits were not preventive in nature, and auditors could not predict the required changes to prevent future processes and compliance gaps. The audits were based only on historical data, and the changes recommended by the auditors were related only to current states. Predictive auditing is about future predictions of anomalies that have not yet occurred but may be in the making process within the system [37]–[40].

(c). Regulators will gain a more efficient tool to monitor compliance with regulatory standards. The proposed blockchain framework creates a transparent and immutable audit trail, allowing regulators to track the performance and behavior of IIoT systems in real time, thus ensuring that manufacturers comply with the relevant laws and regulations.

When client organizations are tied to a blockchain, regulators do not require data samples. regulators' login to the blockchain as a blockchain peer with view-only access can manifest records stored in the blockchain. Because all nodes participating in a network segment had the same copies, the records were validated automatically. Event listeners and loggers for auditing can extract data directly from the business logic followed by smart contracts in simplified data formats such as XML.CSV and JSON, which are stored in a blockchain audit database [41]. The blockchain audit database is an added component for predictive auditing, which can be made a permanent feature within the blockchain design. Audit issues can be generated by a combination of automated analysis, predictions, and brainstorming by experts [42]. If a permanent feature is created, blockchain audit logs can be traceable following audit trails stored in defined categories such as industry, city, and state [43]. Audit trails can be protected by tracing the provenance information of blockchain audit records and maintaining multiple identical copies of audit trails that are accessible in encrypted and digitally approved read-only formats [43], [52]–[54]. Any future evaluation of audit trails can be authenticated through its provenance data, such that the sources of anomalies can be detected and the exact accountabilities can be fixed. This capability can be used to detect and fix anomalies before they

cause damage to the audited systems. The integration of blockchain with predictive auditing can be used to detect and fix engineering-level systemic errors by comparing engineering knowledge with the predictive analytical outcomes.

5.5.3 Policy Implications

The policy implications of this study are relevant to both national and international regulatory bodies, overseeing the adoption and implementation of IIoT systems in industries, such as cloud manufacturing. As the integration of IIoT and blockchain technologies increases, policymakers must establish comprehensive frameworks that address the security, data integrity, and transparency of IIoT systems. This study informs policy decisions in several ways.

(a). Guidelines for Blockchain Adoption in Securiing IIoT in C-MFG: This study provides valuable insights into how blockchain can be utilized to reduce or eliminate the risks arising from cyber–physical system security challenges if they are monitored and controlled continuously in real time. Disasters in industrial systems may require a build-up period before they occur. For example, an explosion in a boiler can occur after prolonged pressure building beyond its operating boundaries. Policymakers can leverage this information to design regulations that encourage the adoption of blockchain for real-time monitoring to detect buildup sequences during their buildup periods. Blockchain peers may be able to take timely protective and preventive actions for industrial applications.

(b). Establishing Industry Standards for Continuous Auditing: As study proposes a framework for continuous auditing through the integration of blockchain and predictive auditing, policymakers can use these findings for real-time visualization of events occurring in critical manufacturing systems [145]. This ensures that manufacturers and auditors across different industries adhere to standardized methods for real-time data validation, security, and anomaly detection. In a larger cybersecurity framework, the proposed provenance blockchain solution can play a prominent role in activity logging, monitoring, and control in the security informatics and event management (SIEM) framework [145], as the activities of IIoT devices can be monitored and compared with their pre-established operating boundaries in real time.

Blockchain peers can be empowered by the continuous flow of state monitoring and breach knowledge to ensure that exploited vulnerable IIoT devices can be quickly identified and administered before major damage occurs.

(c). Supporting the Development of Compliance Frameworks for Cloud Manufacturing: The study advocates an integrated approach to IIoT system security that combines blockchain and predictive auditing for improved compliance monitoring. The proposed framework aligns with regulatory requirements such as those outlined in the proposed AI Act in the European Union, particularly in the areas of data governance, record-keeping, and transparency [146]. The Act mandates detailed record-keeping and documentation of AI systems, which can be facilitated by blockchain technology. By integrating this capability with blockchain technology, the system can ensure that the audit trail is immutable, transparent, and decentralized, making it extremely difficult for malicious actors to tamper with data [147]. The framework was designed such that breaches in movements and weights against the rules defined in the provenance blockchain would log risks visible directly to blockchain peers monitoring smart contract execution. The proposed solution can be applied to many industrial, warehouse, and logistics operation scenarios. For example, highly critical systems, such as boilers, high-voltage transformers, and high-pressure high-velocity oil and gas pipelines, can also be monitored for breaches of state changes against the defined provenance blockchain. Policymakers can use this study to establish more robust regulatory frameworks that monitor and enforce compliance with security standards using the provenance states defined in the smart contracts of provenance blockchains. It is hereby perceived that the proposed solution can be generalized for multiple provenance scenarios provided the variables and their bounds are defined carefully.

Summary

In this chapter, the research findings are analyzed and discussed in relation to the research questions, objectives, and performance of the proposed framework. The concerns raised in the literature were cited and addressed as much as possible by using the system designed and evaluated in this study. Despite the absence of direct studies that compare the performance of the framework, this chapter illustrates how its capabilities surpass those of commercial anomaly detection models, offering faster response times, greater accuracy, and improved adaptability to evolving threats. In addition, the results were consolidated, interpreted, and discussed in relation to the relevant literature. This analysis underscores the potential of the proposed framework to set a new standard for security and accountability in IIoT environments, presenting a clear advancement over the existing methodologies. The next chapter presents the conclusions of this study. In addition to the conclusions, the next chapter also provides recommendations on how this system can be improved in future research studies.

Chapter VI

Conclusion and Future work

6.1 Summary of the Study

This dissertation addresses a pressing challenge in modern industrial systems: ensuring security, trustworthiness, and resilience in Industrial Internet of Things (IIoT) enabled cloud manufacturing environments. To address this challenge, this study proposes a hybrid ML–blockchain framework that combines predictive behavior auditing powered by a Random Forest classifier with smart contract enforcement through the CORDA permissioned blockchain. The framework was rigorously evaluated using a multi-scenario simulation featuring fully compliant, partially compliant, and rogue IIoT-enabled forklift behaviors across three operational zones and two cloud-hosted virtual machines. Key performance indicators, including traceability, detection accuracy and latency were measured to determine system effectiveness under various operational and threat conditions.

6.2 Addressing the Research Questions

RQ1: How can a blockchain-based provenance tracking be optimized to support continuous visibility and traceability of IIoT-enabled Cloud Manufacturing (C-MFG)?

This study demonstrated that the integration of CORDA's permissioned blockchain and smart contracts established tamper-proof, transparent, and immutable audit trails,

achieving traceability scores above 82% across all test cases. This extends traditional notions of traceability from static data provenance to dynamic, real-time behavioral validation, ensuring accountability and operational integrity. By embedding devices into smart contracts based on trust rankings and enabling continuous monitoring, the system significantly reduces the likelihood of unauthorized access or manipulation. Furthermore, the immutable audit ledger supports reverse traceability, allowing forensic identification of the origin and trajectory of anomalies. This meaningfully contributes to the discourse on scalable blockchain solutions in IIoT environments.

RQ2: How can predictive auditing be integrated with blockchain technology to enhance real-time anomaly detection and proactive threat mitigation?

The predictive auditing module achieved precision levels between 96% and 100%and recall rates up to 85.4%. This confirmed the viability of supervised machine learning for the real-time classification of IIoT behaviors. The model was further enhanced using device-specific training datasets that were dynamically updated from the provenance database. This allowed for fine-grained anomaly detection, whereby deviations in behavior could be detected, logged, and escalated for review, enabling both preemptive intervention and historical auditability.

RQ3: How can integrating blockchain and predictive auditing ensure data integrity, traceability, and security, while maintaining operational continuity in IIoT-enabled C-MFGs?

The system successfully identified, blocked, and responded to rogue behavior within a latency threshold of 4.5 seconds, ensuring that the operations could continue without degradation. The smart ledger component prevents state changes during elevated risk conditions, thereby ensuring that threats are neither propagated nor ignored. This proactive risk-management mechanism, where participants are required to log and escalate detected threats, marked a significant advancement in autonomous incident response. The framework demonstrated that on-chain intelligence combined with ML-driven foresight can deliver rapid, accurate, and self-regulating threat mitigation.

6.3 Key Contributions

6.3.1 Theoretical Contributions

(a). A decentralized trust model that operationalizes auditability as a core system design feature rather than as a post-hoc compliance measure.

(b). Advanced literature on hybrid cyber-physical architectures shows that ML and blockchain technologies can function symbiotically to enhance trust, resilience, and transparency in IIoT systems.

(c). Reframed traceability from a passive record-keeping function to an active, continuous process of behavioral verification and risk propagation.

6.3.2 Methodological Contributions

(a). A partitioned training data methodology was introduced in which each device's operational behavior was independently modeled for greater classification accuracy.

(b). Establish a risk-tiered consensus mechanism, where smart contracts dynamically restrict or escalate state changes based on real-time risk assessments logged via smart ledgers.

6.3.3 Practical Contributions

(a). Demonstrated that CORDA-based smart contracts can support real-time auditability..

(b). It enabled a second line of defense through predictive auditing, allowing manufacturers, auditors, and regulators to move from periodic to continuous compliance monitoring.

(c). Enhanced device allocation trust via long-term provenance logs, allowing manufacturers to rank and assign IIoT devices based on historical compliance.

6.4 Limitations

Although this study has achieved its objectives, some limitations must be acknowledged.

6.4.1 Dataset and Testing Environment

The framework is evaluated using simulated scenarios constrained by hypothetical warehouse settings. Although these scenarios are crafted to mirror real-world operations, they may not fully encapsulate the complexity inherent in diverse manufacturing environments. Future studies should incorporate larger and more varied datasets to enhance generalizability.

6.4.2 Discussion on Comparative Baseline and Claim Boundaries.

Although this study demonstrates promising outcomes in terms of device visibility, traceability, and anomaly detection, it is important to acknowledge a key limitation: the absence of a direct comparative baseline against existing ML–blockchain frameworks or industrial monitoring solutions. Although individual components such as predictive auditing and distributed logging were evaluated within the proposed architecture, a head-to-head comparison with alternative systems was beyond the scope of this dissertation because of time and resource constraints. While the findings indicate the framework's potential, claims related to superior performance, scalability, or architectural novelty must be interpreted with caution. Without a standardized benchmark or reference system, it is difficult to isolate the performance gains attributable solely to the hybrid integration approach. Future work should aim to incorporate comparative testing against monolithic ML solutions, traditional access control mechanisms, or blockchain logging systems without real-time predictive intelligence. In light of this, the study positions itself not as a definitive solution outperforming all others but rather as a conceptual and experimental proof-of-concept for an integrated, adaptive, and traceable IIoT security framework. This architecture is presented as a plausible and extensible alternative, meriting further validation through broader empirical benchmarking.

6.4.3 Provenance Privacy and Regulatory Compliance.

Although the framework facilitates traceability and auditability, it lacks an integrated privacy engine. It is presumed that device identity mapping and user consent management can be managed externally by deploying organizations. This approach may not fulfill all regulatory requirements, such as the General Data Protection Regulation (GDPR) mandates concerning the right to erasure, data minimization, or purpose limitations.

6.5 Future Research Directions

Although the proposed framework has shown satisfactory performance in key areas, such as traceability, security, and efficiency, there are a number of avenues to explore with regard to future research and development to enhance its capabilities and extend its usability to a wider industrial context. The following domains are important for extending theframework.

Optimization for Large-Scale Deployments: Despite the hybrid logging mechanism and dynamic consensus model ensuring efficiency, more optimizations must be implemented to support extremely large-scale IIoT deployments. Further studies may include various approaches for compressing blockchain log activities, optimizing consensus algorithms, and developing new distributed ledger technologies that yield low latency, low computational demands, and high scalability. Furthermore, future studies can design an architecture with parallel computing using multiple transmitters and receivers, multiple asset files for training machine learning, and multiple blockchain peers entering data on multiple assets for multiple smart contracts. This architecture is a parallel processing and multitasking framework, as is expected in real industrial networks. One may imagine a single blockchain comprising several smart contracts, smart ledgers, and industrial devices registered in each smart contract. Machine learning will be common, but will be trained by the data files of several industrial devices and logging risks after making predictions after each learning event and applying the rules engine. When several industrial assets would transmit their states to the listener files, who in turn would commit the data into the ProvDB database and the training data files of the respective assets, some kind of queuing develops during machine learning, which was not tested in this study.

Edge Computing Integration: Exploring the integration of edge computing to provide real-time data processing. Edge computing refers to processing that may be closer to the data source, relying on less cloud infrastructure, and reducing latency. This integration may provide even more value in real time, which is important for environments, such as autonomous vehicles and industrial automation facilities.

Handling False Positives and Critical Errors: Our approach rely on a deterministic enforcement pipeline based on the probabilistic model outputs. This raises the following concerns in edge cases:

- A false high-risk score could lead to unwarranted action against a compliant device.
- A false negative can fail to prevent a critical event.

This anticipates a more adaptive and error-tolerant system, which is particularly critical in high-stakes industrial settings.

Scalability Considerations: Although the simulation effectively demonstrates performance within a limited three-zone warehouse scenario, real-world industrial systems would encompass hundreds or thousands of concurrent IIoT endpoints. In such environments, both the processing load of the ML engine and latency of the blockchain may emerge as bottlenecks. Future iterations of this framework should investigate model optimization techniques (e.g., pruning and quantization) to decrease ML inference time, blockchain sharding or pruning strategies to mitigate ledger bloat, and edge–cloud partitioning, where anomaly detection occurs at the edge, with only high-risk events being escalated to the blockchain layer. These enhancements can enable the system to scale horizontally without compromising its fundamental guarantees of security, traceability, and autonomous enforcement.

Blockchain Latency: While the architecture incorporates strategies to mitigate latency, such as off-chain storage, event summarization, and selective logging, blockchain latency continues to pose a challenge contingent upon context and scale. Future research should include stress testing under conditions of high transaction volumes, a comparative analysis with alternative distributed ledger technology (DLT) platforms such as Hyperledger and IOTA, and the application of formal queuing models for latency prediction.

Consequently, rather than resolving latency issues, the current architecture provides a domain-specific mitigation strategy that is effective under the current test conditions but requires further validation in more demanding or distributed environments.

Final Statement

In conclusion, this dissertation contributes a robust, scalable, and intelligent framework for achieving secure, traceable, and resilient IIoT-enabled cloud manufacturing. By seamlessly fusing ML-driven predictive analytics with blockchain-based accountability, the framework redefines how industrial systems can detect threats, validate behavior, and enforce compliance in real time. This work lays a foundational step toward self-governing cyber-physical systems, where security is not reactive but embedded, auditable, and adaptive by design, offering a compelling pathway for the trustworthy digital transformation of modern manufacturing.

Reference

[1] Kartal Akdil, Alp Ustundag, and Emre Cevikcan. "Maturity and readiness model for industry 4.0 strategy". In: *Industry 4.0: Managing the digital transformation* (2018), pp. 61–94.

[2] Beyzanur Ervural and Bilal Ervural. "Overview of cyber security in the industry 4.0 era". In: *Industry 4.0: managing the digital transformation* (2018), pp. 267–284.

[3] Emre Cevikcan Alp Ustundag. "A conceptual framework for Industry 4.0". In: *Industry 4.0: managing the digital transformation* (2018), pp. 3–23.

[4] Martin Wollschlaeger, Thilo Sauter, and Juergen Jasperneite. "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0". In: *IEEE industrial electronics magazine* 11.1 (2017), pp. 17–27.

[5] Quanxin Zhao. "Presents the technology, protocols, and new innovations in Industrial Internet of Things (IIoT)". In: *Internet of Things for Industry 4.0: Design, Challenges and Solutions* (2020), pp. 39–56.

[6] Einollah Jafarnejad Ghomi, Amir Masoud Rahmani, and Nooruldeen Nasih Qader. "Cloud manufacturing: challenges, recent advances, open research issues, and future trends". In: *The International Journal of Advanced Manufacturing Technology* 102 (2019), pp. 3613–3639.

[7] Ming K Lim, Weiqing Xiong, and Chao Wang. "Cloud manufacturing architecture: a critical analysis of its development, characteristics and future agenda to support its adoption". In: *Industrial Management & Data Systems* 121.10 (2021), pp. 2143–2180.

[8] Iain Barclay, Alun Preece, and Ian Taylor. “Defining the collective intelligence supply chain”. In: *arXiv preprint arXiv:1809.09444* (2018).

[9] Selvi Arumugam et al. “IOT Enabled Smart Logistics Using Smart Contracts”. In: Aug. 2018, pp. 1–6. DOI: 10.1109/LISS.2018.8593220.

[10] Manoshi Turjo, Mohammad Monirujjaman, and Manjit Kaur. “Smart Supply Chain Management Using the Blockchain and Smart Contract”. In: *Scientific Programming* 2021 (Sept. 2021), pp. 1–12. DOI: 10.1155/2021/6092792.

[11] Tharaka Hewa, Yining Hu, and Madhusanka Liyanage. “Survey on Blockchain-Based Smart Contracts: Technical Aspects and Future Research”. In: *IEEE Access* 9 (2021), pp. 87643–87662. DOI: 10.1109/ACCESS.2021.3068178.

[12] Mehrdokht Pournader et al. “Blockchain applications in supply chains, transport and logistics: a systematic review of the literature”. In: *International Journal of Production Research* 58.7 (2020), pp. 2063–2081. DOI: 10.1080/00207543.2019.1650976.

[13] Parastoo Veisi. “Visualizing provenance in a supply chain using ethereum blockchain”. PhD thesis. University of Saskatchewan, 2019.

[14] Saqib Ali, Guojun Wang, and Md Zakirul Bhuiyan. “Secure Data Provenance in Cloud-Centric Internet of Things via Blockchain Smart Contracts”. In: *2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. 2018, pp. 991–998. DOI: 10.1109/SmartWorld.2018.00175.

[15] Liang Xueping, Sachin Shetty, and Tosh Deepak. “ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability”. In: *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. 2017, pp. 468–477. DOI: 10.1109/CCGRID.2017.8.

[16] Aravind Ramachandran and Murat Kantarcioglu. “Smartprovenance: a distributed, blockchain based dataprovenance system”. In: *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*. 2018, pp. 35–42.

[17] Pingcheng Ruan, Gang Chen, and Tien Dinh. "Fine-grained, secure and efficient data provenance on blockchain systems". In: *Proceedings of the VLDB Endowment* 12.9 (2019), pp. 975–988.

[18] Ebelechukwu Nwafor, Andre Campbell, and David Hill. "Towards a provenance collection framework for internet of things devices". In: *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE. 2017, pp. 1–6.

[19] Lei Yang, Wanrong Zou, and Jianggan Wang. "EdgeShare: A blockchain-based edge data-sharing framework for Industrial Internet of Things". In: *Neurocomputing* 485 (2022), pp. 219–232.

[20] Shahid Latif, Zeba Idrees, and Zil Huma. "Blockchain technology for the industrial Internet of Things: A comprehensive survey on security challenges, architectures, applications, and future research directions". In: *Transactions on Emerging Telecommunications Technologies* 32.11 (2021), e4337.

[21] Sabah Suhail, Choon Seong Hong, and Zuhaib Uddin Ahmad. "Introducing secure provenance in IoT: Requirements and challenges". In: *2016 International Workshop on Secure Internet of Things (SIoT)*. IEEE. 2016, pp. 39–46.

[22] Farooq Israr Ahmed Shaikh. *Security framework for the Internet of Things leveraging network telescopes and machine learning*. University of South Florida, 2019.

[23] Ruchika Gupta, Rakesh Kumar Phanden, and Shubham Sharma. "Security in manufacturing systems in the age of industry 4.0: Pitfalls and possibilities". In: *Advances in Industrial and Production Engineering: Select Proceedings of FLAME 2020*. Springer. 2021, pp. 105–113.

[24] Mifta Ahmed Umer, Elefelious Getachew Belay, and Luis Borges Gouveia. "Leveraging Artificial Intelligence and Provenance Blockchain Framework to Mitigate Risks in Cloud Manufacturing in Industry 4.0". In: *Electronics* 13.3 (2024), p. 660.

[25] Md Haris Uddin Sharif and Mehmood Ali Mohammed. "A literature review of financial losses statistics for cyber security and future trend". In: *World Journal of Advanced Research and Reviews* 15.1 (2022), pp. 138–156.

[26] Daniel DiMase, Zachary A Collier, and Kenneth Heffner. "Systems engineering framework for cyber physical security and resilience". In: *Environment Systems and Decisions* 35 (2015), pp. 291–300.

[27] Bruno Lepri, Nuria Oliver, and Emmanuel Letouz. "Fair, transparent, and accountable algorithmic decision-making processes: The premise, the proposed solutions, and the open challenges". In: *Philosophy & Technology* 31.4 (2018), pp. 611–627.

[28] Elizabeth Reddy, Baki Cakici, and Andrea Ballestero. "Beyond mystery: Putting algorithmic accountability in context". In: *Big Data & Society* 6.1 (2019), p. 2053951719826856.

[29] Gaper kulj, Rok Vrabi, and Peter Butala. "Decentralised network architecture for cloud manufacturing". In: *International Journal of Computer Integrated Manufacturing* 30.4-5 (2017), pp. 395–408.

[30] Steven Walker-Roberts, Mohammad Hammoudeh, and Omar Aldabbas. "Threats on the horizon: Understanding security threats in the era of cyber-physical systems". In: *The Journal of Supercomputing* 76 (2020), pp. 2643–2664.

[31] Marion Pauline Gauthier and Nathalie Brender. "How do the current auditing standards fit the emergent use of blockchain?" In: *Managerial auditing journal* 36.3 (2021), pp. 365–385.

[32] Maria Cadiz Dyball and Ravi Seethamraju. "The impact of client use of blockchain technology on audit risk and audit approach—An exploratory study". In: *International Journal of Auditing* 25.2 (2021), pp. 602–615.

[33] Aneta Zemánková. "Artificial intelligence and blockchain in audit and accounting: Literature review". In: *wseas Transactions on Business and Economics* 16.1 (2019), pp. 568–581.

[34] Siripan Kuenkaikaew and Miklos A Vasarhelyi. "The predictive audit framework". In: *The International Journal of Digital Accounting Research* 13.19 (2013), pp. 37–71.

[35] Nigar Hashimzade, Gareth D Myles, and Matthew D Rablen. "Predictive analytics and the targeting of audits". In: *Journal of economic behavior & organization* 124 (2016), pp. 130–145.

[36] Daniel Broby. "The use of predictive analytics in finance". In: *The Journal of Finance and Data Science* 8 (2022), pp. 145–161.

[37] Deniz Appelbaum, Alexander Kogan, and Miklos A Vasarhelyi. "Big data and analytics in the modern audit engagement: Research needs". In: *Auditing: A Journal of Practice & Theory* 36.4 (2017), pp. 1–27.

[38] Michael Adidharma Mervelito and Baihaqi Arsyad Lintang. "Internal Auditing Paradigm Shift: From Traditional Audits to Audits in the 4.0 Industry Era". In: *International Journal of Innovative Science and Research Technology* 6.3 (2021), pp. 56–63.

[39] Connor Ortman. "Blockchain and the Future of the Audit". In: (2018).

[40] Erica Pimentel, Emilio Boulianne, and Shayan Eskandari. "Systemizing the challenges of auditing blockchain-based assets". In: *Journal of Information Systems* 35.2 (2021), pp. 61–75.

[41] Ashar Ahmad, Muhammad Saad, and Laurent Njilla. "Blocktrail: A scalable multichain solution for blockchain-based audit trails". In: *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE. 2019, pp. 1–6.

[42] Hongdan Han, Radha K Shiwakoti, and Robin Jarvis. "Accounting and auditing with blockchain technology and artificial Intelligence: A literature review". In: *International Journal of Accounting Information Systems* 48 (2023), p. 100598.

[43] Lucian Carata. "Provenance-based computing". PhD thesis. 2019.

[44] Aitizaz Ali, Hashim Ali, and Aamir Saeed. "Blockchain-powered healthcare systems: enhancing scalability and security with hybrid deep learning". In: *Sensors* 23.18 (2023), p. 7740.

[45] Charles Guandaru Kamau and Abdulkerim Yavuzaslan. "CryptoAudit: Nature, requirements and challenges of Blockchain transactions audit". In: *African Journal of Commercial Studies* 3.2 (2023), pp. 101–107.

[46] Harun Jamil, Faiza Qayyum, and Naeem Khan Iqbal. "Secure hydrogen production analysis and prediction based on blockchain service framework for intelligent power management system". In: *Smart Cities* 6.6 (2023), pp. 3192–3224.

[47] Hussam Saeed Musa, Moez Krichen, and Adem Alpaslan Altun. "Survey on blockchain-based data storage security for android mobile applications". In: *Sensors* 23.21 (2023), p. 8749.

[48] Ahmad-Reza Sadeghi, Christian Wachsmann, and Michael Waidner. "Security and privacy challenges in industrial internet of things". In: *Proceedings of the 52nd annual design automation conference*. 2015, pp. 1–6.

[49] Lachlan Urquhart and Derek McAuley. "Avoiding the internet of insecure industrial things". In: *Computer law & security review* 34.3 (2018), pp. 450–466.

[50] Sebahattin Demirkan, Irem Demirkan, and Andrew McKee. "Blockchain technology in the future of business cyber security and accounting". In: *Journal of Management Analytics* 7.2 (2020), pp. 189–208.

[51] Jin Li, Xiaofeng Chen, and Qiong Wong Huang. "Digital provenance: Enabling secure data forensics in cloud computing". In: *Future Generation Computer Systems* 37 (2014), pp. 259–266.

[52] Arnab Banerjee. "Blockchain technology: supply chain insights from ERP". In: *Advances in computers*. Vol. 111. Elsevier, 2018, pp. 69–98.

[53] Abubakar Sadiq Sani, Dong Yuan, and Wei Yeoh Bao. "Xyreum: A high-performance and scalable blockchain for iiot security and privacy". In: *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. 2019, pp. 1920–1930.

[54] Xingjuan Cai, Shaojin Geng, and Jingbo Wu Zhang. "A sharding scheme-based many-objective optimization algorithm for enhancing security in blockchain-enabled industrial internet of things". In: *IEEE Transactions on Industrial Informatics* 17.11 (2021), pp. 7650–7658.

[55] Xinzheng Feng, Jun Wu, and Yulei Li Wu. "Blockchain and digital twin empowered trustworthy self-healing for edge-AI enabled industrial Internet of things". In: *Information Sciences* 642 (2023), p. 119169.

[56] Oscar Carlsson, Daniel Vera, and Eduardo Tauber Arceredillo. "Engineering of IoT automation systems". In: *IoT Automation*. CRC Press, 2017, pp. 197–246.

[57] Sofia Terzi, Angeliki Zacharaki, and Alexandros Votis Nizamis. “Transforming the supply-chain management and industry logistics with blockchain smart contracts”. In: *Proceedings of the 23rd Pan-Hellenic conference on informatics*. 2019, pp. 9–14.

[58] Muath A Obaidat and Joseph Brown. “Perspectives of Blockchain in cybersecurity: Applications and future developments”. In: *Research anthology on convergence of Blockchain, internet of things, and security*. IGI Global, 2023, pp. 818–840.

[59] Matluba Khodjaeva, Muath Obaidat, and Douglas Salane. “Mitigating threats and vulnerabilities of RFID in IoT through outsourcing computations for public key cryptography”. In: *Security, Privacy and Trust in the IoT Environment* (2019), pp. 39–60.

[60] Jan Pennekamp, Lennart Bader, and Roman Niemietz Matzutt. “Private multi-hop accountability for supply chains”. In: *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2020, pp. 1–7.

[61] Isuru Suriarachchi, Sachith Withana, and Beth Plale. “Big provenance stream processing for data intensive computations”. In: *2018 IEEE 14th International Conference on e-Science (e-Science)*. IEEE. 2018, pp. 245–255.

[62] Mufajjul Ali. “Provenance-based data traceability model and policy enforcement framework for cloud services”. PhD thesis. University of Southampton, 2016.

[63] Fengqun Wang et al. “Blockchain-based lightweight message authentication for edge-assisted cross-domain industrial internet of things”. In: *IEEE transactions on dependable and secure computing* (2023).

[64] Jie Cui, Nan Liu, and Qingyang Zhang. “Efficient and anonymous cross-domain authentication for IIoT based on blockchain”. In: *IEEE Transactions on Network Science and Engineering* 10.2 (2022), pp. 899–910.

[65] Marten Sigwart, Michael Borkowski, and Marco Peise. “A secure and extensible blockchain-based data provenance framework for the Internet of Things”. In: *Personal and Ubiquitous Computing* (2020), pp. 1–15.

[66] Amrita Jyoti and R. K. Chauhan. “A Blockchain and Smart Contract-based Data Provenance Collection and Storing in Cloud Environment”. In: *Wireless Networks* 28.4 (2022), pp. 1541–1562.

[67] Malik Sidra, Dedeoglu Volkan, and Kanhere Salil S. "PrivChain: Provenance and Privacy Preservation in Blockchain Enabled Supply Chains". In: *2022 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2022, pp. 157–166.

[68] Yao Yu, Shumei Liu, and Phee Lep Yeoh. "LayerChain: A Hierarchical Edge-cloud Blockchain for Large-scale Low-delay Industrial Internet of Things Applications". In: *IEEE Transactions on Industrial Informatics* 17.7 (2020), pp. 5077–5086.

[69] Samir M. Umran, Songfeng Lu, and Zaid Ameen Abduljabbar. "Secure Data of Industrial Internet of Things in a Cement Factory Based on a Blockchain Technology". In: *Applied Sciences* 11.14 (2021), p. 6376.

[70] Thomas Pasquier, Jatinder Singh, and Julia Powles. "Data Provenance to Audit Compliance with Privacy Policy in the Internet of Things". In: *Personal and Ubiquitous Computing* 22 (2018), pp. 333–344.

[71] Junqin Huang, Linghe Kong, and Guihai Chen. "Towards Secure Industrial IoT: Blockchain System with Credit-based Consensus Mechanism". In: *IEEE Transactions on Industrial Informatics* 15.6 (2019), pp. 3680–3689.

[72] Ningjie Gao, Ru Huo, and Shuo Wang. "Sharding-Hashgraph: A High-Performance Blockchain-based Framework for Industrial Internet of Things with Hashgraph Mechanism". In: *IEEE Internet of Things Journal* 9.18 (2021), pp. 17070–17079.

[73] Yan Cao, Feng Jia, and Gunasekaran Manogaran. "Efficient Traceability Systems of Steel Products Using Blockchain-based Industrial Internet of Things". In: *IEEE Transactions on Industrial Informatics* 16.9 (2019), pp. 6004–6012.

[74] Mengting Liu, F. Richard Yu, and Yinglei Teng. "Performance Optimization for Blockchain-enabled Industrial Internet of Things (IIoT) Systems: A Deep Reinforcement Learning Approach". In: *IEEE Transactions on Industrial Informatics* 15.6 (2019), pp. 3559–3570.

[75] Adebola Folorunso, Temitope Adewumi, and Adeola Adewa. "Impact of AI on Cybersecurity and Security Compliance". In: *Global Journal of Engineering and Technology Advances* 21.01 (2024), pp. 167–184.

[76] Mujaheed Abdullahi, Yahia Baashar, and Hitham Alhussian. "Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods: A Systematic Literature Review". In: *Electronics* 11.2 (2022), p. 198.

[77] Lloyd Chilongo and Abubakkar Sithik Km. "Impact of Artificial Intelligence on Cybersecurity: A Case of Internet of Things". In: *i-Manager's Journal on Digital Forensics & Cyber Security (JDF)* 2.1 (2024).

[78] Sotirios Messinis, Nikos Temenos, and Nicholas E. Protonotarios. "Enhancing Internet of Medical Things Security with Artificial Intelligence: A Comprehensive Review". In: *Computers in Biology and Medicine* 170 (2024), p. 108036.

[79] Shakila Zaman, Khaled Alhazmi, and Mohammed A. Aseeri. "Security Threats and Artificial Intelligence based Countermeasures for Internet of Things Networks: A Comprehensive Survey". In: *IEEE Access* 9 (2021), pp. 94668–94690.

[80] Hui Wu, Haiting Han, and Xiao Wang. "Research on Artificial Intelligence Enhancing Internet of Things Security: A Survey". In: *IEEE Access* 8 (2020), pp. 153826–153848.

[81] Zhanyang Xu, Wentao Liu, and Jingwang Huang. "Artificial Intelligence for Securing IoT Services in Edge Computing: A Survey". In: *Security and Communication Networks* 2020.1 (2020), p. 8872586.

[82] Miloud Bagaa, Tarik Taleb, and Jorge Bernal Bernabe. "A Machine Learning Security Framework for IoT Systems". In: *IEEE Access* 8 (2020), pp. 114066–114077.

[83] Rashmita Khilar, K. Mariyappan, and Mary Subaja Christo. "Artificial Intelligence-Based Security Protocols to Resist Attacks in Internet of Things". In: *Wireless Communications and Mobile Computing* 2022.1 (2022), p. 1440538.

[84] Murat Kuzlu, Corinne Fair, and Ozgur Guler. "Role of Artificial Intelligence in the Internet of Things (IoT) Cybersecurity". In: *Discover Internet of Things* 1.1 (2021), p. 7.

[85] Celestine Iwendi, Saif Ur Rehman, and Abdul Rehman Javed. "Sustainable Security for the Internet of Things Using Artificial Intelligence Architectures". In: *ACM Transactions on Internet Technology (TOIT)* 21.3 (2021), pp. 1–22.

[86] Liang Xiao, Xiaoyue Wan, and Xiaozhen Lu. "IoT Security Techniques Based on Machine Learning: How Do IoT Devices Use AI to Enhance Security?" In: *IEEE Signal Processing Magazine* 35.5 (2018), pp. 41–49.

[87] Mohammed Ali Al-Garadi, Amr Mohamed, and Abdulla Khalid Al-Ali. "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security". In: *IEEE Communications Surveys & Tutorials* 22.3 (2020), pp. 1646–1685.

[88] Fatima Hussain, Rasheed Hussain, and Syed Ali Hassan. "Machine Learning in IoT Security: Current Solutions and Future Challenges". In: *IEEE Communications Surveys & Tutorials* 22.3 (2020), pp. 1686–1721.

[89] Geetanjali Rathee, Adel Khelifi, and Razi Iqbal. "Artificial Intelligence-(AI-) Enabled Internet of Things (IoT) for Secure Big Data Processing in Multihoming Networks". In: *Wireless Communications and Mobile Computing* 2021.1 (2021), p. 5754322.

[90] Shatha Alharbi, Afraa Attiah, and Daniyal Alghazzawi. "Integrating Blockchain with Artificial Intelligence to Secure IoT Networks: Future Trends". In: *Sustainability* 14.23 (2022), p. 16002.

[91] Ankit Attkan and Virender Ranga. "Cyber-Physical Security for IoT Networks: A Comprehensive Review on Traditional, Blockchain and Artificial Intelligence Based Key-Security". In: *Complex & Intelligent Systems* 8.4 (2022), pp. 3559–3591.

[92] Jiahong Cai, Wei Liang, and Xiong Li. "GTxChain: A Secure IoT Smart Blockchain Architecture Based on Graph Neural Network". In: *IEEE Internet of Things Journal* 10.24 (2023), pp. 21502–21514.

[93] Shitharth Selvarajan, Gautam Srivastava, and Alaa O. Khadidos. "An Artificial Intelligence Lightweight Blockchain Security Model for Security and Privacy in IIoT Systems". In: *Journal of Cloud Computing* 12.1 (2023), p. 38.

[94] Sara Motiee, Kirstie Hawkey, and Konstantin Beznosov. "Do Windows Users Follow the Principle of Least Privilege? Investigating User Account Control Practices". In: *Proceedings of the Sixth Symposium on Usable Privacy and Security*. 2010, pp. 1–13.

[95] Lei Meng, Daochao Huang, and Jiahang Xianwei Zhou. "A Continuous Authentication Protocol without Trust Authority for Zero Trust Architecture". In: *China Communications* 19.8 (2022), pp. 198–213.

[96] M. Misbahuddin, B. S. Bindhumadhava, and B. Dheeptha. "Design of a Risk based Authentication System Using Machine Learning Techniques". In: *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, Aug. 2017, pp. 1–6.

[97] Aintzane Mosteiro-Sanchez, Marc Barcelo, and Jasone Astorga. "Securing IIoT Using Defence-in-Depth: Towards an End-to-End Secure Industry 4.0". In: *Journal of Manufacturing Systems* 57 (2020), pp. 367–378.

[98] Chaimaa Boudagdigue, Abderrahim Benslimane, and Abdellatif Kobbane. "Trust-based Certificate Management for Industrial IoT Networks". In: *IEEE Internet of Things Journal* 10.14 (2023), pp. 12867–12885.

[99] Mohammad Mehedi Hassan, Shamsul Huda, and Shaila Sharmeen. "An Adaptive Trust Boundary Protection for IIoT Networks Using Deep-Learning Feature-Extraction-based Semisupervised Model". In: *IEEE Transactions on Industrial Informatics* 17.4 (2020), pp. 2860–2870.

[100] Imanol Mugarza, Jose Luis Flores, and Jose Luis Montero. "Security Issues and Software Updates Management in the Industrial Internet of Things (IIoT) Era". In: *Sensors* 20.24 (2020), p. 7160.

[101] Ethereum. *Intro to Ethereum*. `https://ethereum.org/en/developers/docs/intro-to-ethereum/`. [Accessed: 02 May 2023]. 2023.

[102] Ethereum. *Introduction to Smart Contracts*. `https://ethereum.org/en/developers/docs/smart-contracts/`. [Accessed: 02 May 2023]. 2023.

[103] Ethereum. *Intro to DApps*. `https://ethereum.org/en/developers/docs/dapps/`. [Accessed: 02 May 2023]. 2023.

[104] Hyperledger. *Fabric Docs: Introduction*. `https://hyperledger-fabric.readthedocs.io/en/release-2.5/blockchain.html`. [Accessed: 02 May 2023]. 2023.

[105] Corda. *Key Concepts of the Corda Community Edition*. `https://docs.r3.com/en/platform/corda/4.10/community/about-corda/corda-key-concepts.html`. [Accessed: 02 May 2023]. 2023.

[106] Docker. *Docker Overview*. https://docs.docker.com/get-started/overview/. [Accessed: 02 May 2023]. 2023.

[107] Hyperledger. *Install the Fabric and Fabric Samples*. https://hyperledger-fabric.readthedocs.io/en/release-2.5/install.html. [Accessed: 02 May 2023]. 2023.

[108] Hyperledger. *Fabric Gateway*. https://hyperledger.github.io/fabric-gateway/. [Accessed: 02 May 2023]. 2023.

[109] Hyperledger. *Hyperledger Fabric Gateway SDK for Java*. https://hyperledger.github.io/fabric-gateway-java/. [Accessed: 02 May 2023]. 2023.

[110] Hyperledger. *Fabric Chaincode Java*. https://hyperledger.github.io/fabric-chaincode-java/. [Accessed: 03 May 2023]. 2023.

[111] Corda. *Building Your First Basic CorDapp*. https://docs.r3.com/en/platform/corda/4.10/community/get-started/tutorials/build-basic-cordapp/basic-cordapp-intro.html. [Accessed: 03 May 2023]. 2023.

[112] Corda. *Write States*. https://docs.r3.com/en/platform/corda/4.10/community/get-started/tutorials/build-basic-cordapp/basic-cordapp-state.html. [Accessed: 03 May 2023]. 2023.

[113] Corda. *Corda Community Edition Key Concepts*. https://docs.r3.com/en/platform/CORDA/4.10/community/about-CORDA/CORDA-key-concepts.html. [Accessed: 02 May 2023]. 2023.

[114] Corda. *Building the First Basic CorDapp*. https://docs.r3.com/en/platform/CORDA/4.10/community/get-started/tutorials/build-basic-CORDApp/basic-CORDApp-intro.html. [Accessed: 03 May 2023]. 2023.

[115] Leo Breiman. "Random forests". In: *Machine Learning* 45 (2001), pp. 5–32.

[116] Mark Saunders, Philip Lewis, and Adrian Thornhill. *Research Methods for Business Students*. Essex: Prentice Hall: Financial Times, 2003.

[117] Emma Bell, Bill Harley, and Alan Bryman. *Business Research Methods*. Oxford University Press, 2022.

[118] Alan R. Hevner, Salvatore T. March, and Jinsoo Park. "Design science in information systems research". In: *MIS Quarterly* 28.1 (2004), pp. 75–105.

[119] Ken Peffers, Tuure Tuunanen, and Marcus A. Rothenberger. “A design science research methodology for information systems research”. In: *Journal of Management Information Systems* 24.3 (2007), pp. 45–77.

[120] Thomas C. Reeves, Jan Herrington, and Ron Oliver. “Design research: A socially responsible approach to instructional technology research in higher education”. In: *Journal of Computing in Higher Education* 16 (2005), pp. 96–115.

[121] Diksha Rangwani, Dipanwita Sadhukhan, and Ray Sangram. “A robust provable-secure privacy-preserving authentication protocol for Industrial Internet of Things”. In: *Peer-to-Peer Networking and Applications* 14.3 (2021), pp. 1548–1571.

[122] Kuljeet Kaur, Sahil Garg, and Gagangeet Singh Aujla. “Edge computing in the industrial internet of things environment: Software-defined-networks-based edge-cloud interplay”. In: *IEEE Communications Magazine* 56.2 (2018), pp. 44–51.

[123] Fan Liang, Wei Yu, and Xing Liu. “Toward computing resource reservation scheduling in Industrial Internet of Things”. In: *IEEE Internet of Things Journal* 8.10 (2020), pp. 8210–8222.

[124] Yuri Santo, Roger Immich, and L. Dalmazo Bruno. “Fault detection on the edge and adaptive communication for state of alert in industrial internet of things”. In: *Sensors* 23.7 (2023), p. 3544.

[125] P. Senthilkumar and K. Rajesh. “Design of a model based engineering deep learning scheduler in cloud computing environment using Industrial Internet of Things (IIoT)”. In: *Journal of Ambient Intelligence and Humanized Computing* (2021), pp. 1–9.

[126] Yu Liu, Zhibo Pang, and Karlsson Magnus. “Anomaly detection based on machine learning in IoT-based vertical plant wall for indoor climate control”. In: *Building and Environment* 183 (2020), p. 107212.

[127] Hanan Hindy, Ethan Bayne, and Miroslav Bures. “Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset)”. In: *International Networking Conference*. Springer International Publishing, 2020, pp. 73–84.

[128] Jordan Lam and Robert Abbas. “Machine learning based anomaly detection for 5G networks”. In: *arXiv preprint arXiv:2003.03474* (2020).

[129] Emilio Sansano. “Machine learning-based techniques for indoor localization and human activity recognition through wearable devices”. PhD thesis. Universitat Jaume I, 2020.

[130] Bradley Potteiger, Goncalo Martins, and Xenofon Koutsoukos. “Software and attack centric integrated threat modeling for quantitative risk assessment”. In: *Proceedings of the Symposium and Bootcamp on the Science of Security*. 2016, pp. 99–108.

[131] Mohamed Badawy, Nada H. Sherief, and Ayman A. Abdel-Hamid. “Legacy ICS cybersecurity assessment using hybrid threat modeling—An oil and gas sector case study”. In: *Applied Sciences* 14.18 (2024), p. 8398.

[132] Abel Yeboah-Ofori and Shareeful Islam. “Cyber security threat modeling for supply chain organizational environments”. In: *Future Internet* 11.3 (2019), p. 63.

[133] Rita Azzi, Rima Kilany Chamoun, and Maria Sokhn. “The power of a blockchain-based supply chain”. In: *Computers & Industrial Engineering* 135 (2019), pp. 582–592.

[134] Ari Sivula, Ahm Shamsuzzoha, and Petri Helo. “Requirements for blockchain technology in supply chain management: An exploratory case study”. In: *Published online* (2021).

[135] Bayu Adhi Tama, Bruno Joachim Kweka, and Park Youngho. “A critical review of blockchain and its current applications”. In: *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*. IEEE, 2017, pp. 109–113.

[136] Cristina Alcaraz, Javier Lopez, and Zhou Jianying. “Secure SCADA framework for the protection of energy control systems”. In: *Concurrency and Computation: Practice and Experience* 23.12 (2011), pp. 1431–1442.

[137] Stamatis Karnouskos and Armando Walter Colombo. “Architecting the next generation of service-based SCADA/DCS system of systems”. In: *IECOn 2011-37th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2011, pp. 359–364.

[138] Hendrik Blockeel and Joaquin Vanschoren. "Experiment databases: Towards an improved experimental methodology in machine learning". In: *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer Berlin Heidelberg, 2007, pp. 6–17.

[139] Gaia Franceschini and Sandro Macchietto. "Validation of a model for biodiesel production through model-based experiment design". In: *Industrial & Engineering Chemistry Research* 46.1 (2007), pp. 220–232.

[140] M. Prasad and T. Srikanth. "Clustering Accuracy Improvement Using Modified Min-Max Normalization". In: *Published online* (2024).

[141] Sergio Ruiz-Villafranca et al. "A MEC-IIoT intelligent threat detector based on machine learning boosted tree algorithms". In: *Computer Networks* 233 (2023), p. 109868.

[142] N. Elmrabit, F. Zhou, and Li H. "Evaluation of Machine Learning Algorithms for Anomaly Detection". In: *2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. Dublin, Ireland: IEEE, June 2020, pp. 1–7.

[143] Gustavo González-Granadillo, Susana González-Zarzosa, and Rodrigo Diaz. "Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures". In: *Sensors* 21.14 (2021), p. 4759.

[144] Simona Ramos and Joshua Ellul. "Blockchain for Artificial Intelligence (AI): Enhancing compliance with the EU AI Act through distributed ledger technology. A cybersecurity perspective". In: *International Cybersecurity Law Review* 5.1 (2024), pp. 1–20.

[145] Shi-Cho Cha and Kuo-Hui Yeh. "An ISO/IEC 15408-2 compliant security auditing system with blockchain technology". In: *2018 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2018, pp. 1–2.

[146] Design-Based Research Collective. "Design-based research: An emerging paradigm for educational inquiry". In: *Educational Researcher* 32.1 (2003), pp. 5–8.

[147] Don Ihde. "The Designer Fallacy and Technological Imagination". In: *Philosophy and Design: From Engineering to Architecture*. Ed. by Peter E. Vermaas et al. Berlin: Springer, 2008, pp. 51–60.

[148] Philip Brey. "Technological Design as an Evolutionary Process". In: *Philosophy and Design: From Engineering to Architecture*. Ed. by Peter E. Vermaas et al. Berlin: Springer, 2008, pp. 61–76.

[149] Patrick Feng and Andrew Feenberg. "Thinking about Design: Critical Theory of Technology and the Design Process". In: *Philosophy and Design: From Engineering to Architecture*. Ed. by Peter E. Vermaas et al. Berlin: Springer, 2008, pp. 105–118.

[150] Kristo Miettinen. "Design: Structure, Process, and Function: A Systems Methodology Perspective". In: *Philosophy and Design: From Engineering to Architecture*. Ed. by Peter E. Vermaas et al. Berlin: Springer, 2008, pp. 217–232.

[151] Jeffrey L. Funk. "Thinking about the Future of Complex Technological Systems: Which Technologies Should Shape Their Designs?" In: *Complex Systems Design & Management Asia*. Ed. by Michel-Alexandre Cardin et al. Heidelberg: Springer, 2015, pp. 133–140.

[152] Stavros Salonikias, Antonios Gouglidis, and Ioannis Mavridis. "Access control in the industrial internet of things". In: *Security and privacy trends in the industrial internet of things* (2019), pp. 95–114.

[153] Muhammad Syafrudin, Ganjar Alfian, and Norma Latif Fitriyani. "Performance analysis of IoT-based sensor, big data processing, and machine learning model for real-time monitoring system in automotive manufacturing". In: *Sensors* 18.9 (2018), p. 2946.

[154] Arunan Sivanathan. "IoT behavioral monitoring via network traffic analysis". In: *arXiv preprint arXiv:2001.10632* (2020).

[155] Kaleido. "Ethereum vs Fabric vs Corda: Blockchain Protocols Compared". In: (2023). https://www.kaleido.io/blockchain-blog/enterprise-blockchain-protocols-a-technical-analysis-of-ethereum-vs-fabric-vs-corda.

[156] Cais Manai. "Contract Upgrades Constraints in V3". In: *Medium* (2018). https://medium.com/c upgrades-constraints-in-v3-d5b14d5fb258.

[157] K. Christidis and M. Devetsikiotis. "Blockchains and Smart Contracts for the Internet of Things". In: *IEEE Access* 4 (2016), pp. 2292–2303. DOI: 10.1109/ACCESS.2016.2566339. URL: https://doi.org/10.1109/ACCESS.2016.2566339.

[158] Y. Lu, X. Xu, and L. Wang. "Smart manufacturing process and system automation—A critical review of the standards and enabling technologies". In: *Journal of Manufacturing Systems* 56 (2020), pp. 364–380. DOI: 10.1016/j.jmsy.2020.07.015. URL: https://doi.org/10.1016/j.jmsy.2020.07.015.

[159] R. G. Brown et al. *Corda: An Introduction*. Tech. rep. R3, 2018. URL: https://docs.r3.com.

[160] H. Xie J.and Tang and Y. Huang. "A survey of blockchain technology applied to smart cities: Research issues and challenges". In: *IEEE* (2021). Publication details like volume, pages, and DOI are missing. Please provide if available.

[161] G. Alfian. "Performance Analysis of IoT-Based Sensor, Big Data Processing, and Machine Learning Model for Real-Time Monitoring System in Automotive Manufacturing". In: *Sensors* 18.9 (2018), p. 2946. DOI: 10.3390/s18092946. URL: https://doi.org/10.3390/s18092946.

[162] N. Abbas et al. "Mobile Edge Computing: A Survey". In: *IEEE Internet of Things Journal* 5.1 (2018), pp. 450–465. DOI: 10.1109/JIOT.2017.2750180. URL: https://doi.org/10.1109/JIOT.2017.2750180.

[163] Y. Zhang, Y. Qian, and H. Sharif. "Security and Privacy for Smart Manufacturing Systems: Challenges and Opportunities". In: *IEEE Transactions on Industrial Informatics* 18.2 (2022), pp. 1328–1338. DOI: 10.1109/TII.2021.3071679. URL: https://doi.org/10.1109/TII.2021.3071679.

[164] International Organization for Standardization. *ISO 3691: Industrial trucks – Safety requirements and verification*. International Organization for Standardization. 2012.

[165] International Electrotechnical Commission. *IEC 61508: Functional safety of electrical/electronic/programmable electronic safety-related systems*. International Electrotechnical Commission. 2010.

[166] H. S. Kang et al. "Smart manufacturing: Past research, present findings, and future directions". In: *IJPEM-Green Technology* 3.1 (2016), pp. 111–128.

[167] J. Lee, B. Bagheri, and H. A. Kao. "A cyber-physical systems architecture for Industry 4.0-based manufacturing systems". In: *Manufacturing Letters* 3 (2015), pp. 18–23.

[168] H. F Atlam, R. J. A. Alenezi, and G. B. Wills Walters. "Developing an adaptive security model for IIoT systems". In: *Future Generation Computer Systems* 86 (2018), pp. 729–737.

[169] P. P. Jayaraman et al. "Internet of Things platform for smart farming: Experiences and lessons learnt". In: *Sensors* 16.11 (2017), p. 1884.

[170] M. Ghafari, M. Hajivaliei, and A. M. Esfahani. "Detection and classification of cyber-physical attacks in smart manufacturing systems". In: *Journal of Manufacturing Systems* 58 (2021), pp. 178–189.

[171] M. Lezzi, M. Lazoi, and A. Corallo. "Cybersecurity for Industry 4.0 in the current literature: A reference framework". In: *Computers in Industry* 103 (2018), pp. 97–110.

[172] Jason Brownlee. *Random Forest for Time Series Forecasting*. Machine Learning Mastery. 2017. URL: `https://machinelearningmastery.com/random-forest-for-time-series-forecasting/`.

[173] RandomTrees. *Mastering Model Retraining in MLOps*. 2024. URL: `https://randomtrees.com/blog/mastering-model-retraining-in-mlops/`.