Securing Smart Building Sensor Network Against Cyber Threats

Abstract

Smart building technologies implement IoT to automate functions including lighting and **HVAC** provisions and security provisions thus improving both energy performance and occupant comfort. Modern building connectivity exposes serious security risks which evade traditional protection tools due to limited adaptivity toward new types of attacks. The proposed framework combines IDS Intrusion Detection Systems operated by Artificial Intelligence with blockchain authentication methods to ensure security of IoT sensor networks within smart buildings. The implementation of machine learning algorithms supports traffic monitoring in real time alongside private blockchain systems which both provide **tamper-resistant decentralized device authentication** and data integrity capabilities. The developed prototype achieved satisfactory evaluation results during simulation tests that revealed high security performance while requiring little human supervision and minimal false alarm rates. The designed framework presents successful results which demonstrate its ability to minimize critical smart building cyber threats making it suitable for practical implementation. The researchers provide final conclusions regarding adoption potential by industries together with recommendations for future advancements in their concluding section.

Introduction and Purpose

Statement of the IT Challenge

Modern smart buildings built with IoT devices have automated facility control through cloud-connected sensors that instantly adjust lighting levels along with climate settings and security camera functions (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) ("). The advantages of IoT-based integration in efficiency and convenience directly correlate with a dramatic expansion of potential cyber attacks against networks. The rising number of cyber fraud and hacking attempts against smart building networks stems from vulnerabilities that exist within IoT deployment systems (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). Several IoT devices arrive to customers with insufficient default credentials while lacking proper authentication mechanisms thus becoming simple targets for attackers and their data communication often occurs without encryption which enables one compromised device to provide entry onto an entire building system (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). Weaknesses in building IoT systems provide threat actors with opportunities to control essential building operations and steal important data as well as disable security systems which places occupants at risk. An attacker who breaches a building's IoT network has the ability to disable alarm systems

and HVAC controls and extract personal information such as occupant data and locations through smart sensors (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). The Mirai botnet of 2016 represents a major example of global IoT security threats since it infected millions of devices worldwide (Project Proposal Draft 1 Feedback[1].docx). An analysis by the industry revealed that IoT attack numbers escalated from 639 million to 1.51 billion breach attempts in the first half of 2021 compared to last year (Optimized IoT Intrusion Detection using Machine Learning Technique) which demonstrates the necessity to protect smart buildings from cyber invasion.

Compounding the Challenge

Traditional IT security tools—**firewalls, intrusion prevention systems, and antivirus software**—are ill-suited for securing smart building IoT networks. Designed for centralized, high-power environments, these solutions struggle to protect resource-constrained, distributed IoT devices and often miss advanced or novel attacks. Their reliance on known attack signatures and centralized access control leaves them ineffective against zero-day exploits and evolving threats.

A single compromised authentication server in such systems can jeopardize the entire network. Furthermore, many IoT sensors lack the computational resources to run standard encryption protocols or security agents, limiting the viability of conventional defenses. This mismatch between modern IoT threats and outdated safeguards forms the core challenge: how to secure smart building sensor networks from cyberattacks that bypass or overwhelm traditional security methods.

Objectives and Scope

In response to the critical security challenges posed by IoT deployments in smart buildings, this research aims to design an intelligent, multi-layered cybersecurity framework capable of proactively detecting and preventing cyberattacks. The proposed solution integrates **artificial intelligence** for real-time anomaly detection and **blockchain technology** for decentralized authentication, addressing the shortcomings of traditional security mechanisms. The research is structured around the following key objectives:

1. **Identify vulnerabilities and risks in IoT-enabled smart buildings**
   Conduct a detailed assessment of common security flaws in IoT devices and communication protocols used in smart buildings. This includes identifying threats such as unauthorized access, malware infections, and man-in-the-middle (MITM) attacks. The objective is to define a clear threat model and establish baseline security requirements for the proposed framework.

2. **Develop an AI-based Intrusion Detection System (IDS)**
   Design and implement a machine learning-driven IDS tailored for IoT traffic within smart buildings. This system will monitor sensor and actuator communications in real time and detect suspicious behavior patterns, such as unexpected data flows or

command sequences that could indicate intrusion or misuse.

3. **Train and optimize detection models for real-time accuracy**
   Use supervised learning techniques and IoT-specific datasets to train the IDS models for high precision and low false positive rates. This phase involves selecting relevant features, tuning algorithms, and ensuring the system operates with minimal latency to support real-time intrusion detection.

4. **Incorporate a blockchain-based authentication mechanism**
   Integrate a private blockchain network to decentralize the authentication of IoT devices. This mechanism ensures only verified devices can join or communicate within the smart building network, and provides an immutable, auditable record of all access attempts and device registrations—thereby mitigating the risks of spoofing or rogue device infiltration.

5. **Evaluate the effectiveness of the integrated solution**
   Test the proposed framework against simulated cyberattacks and compare it with conventional security tools (e.g., traditional IDS and centralized authentication systems). The evaluation will focus on detection accuracy, response time, false alarm rates, and resilience to unauthorized access, quantifying improvements brought by the AI + blockchain approach.

The research addresses the protection of IoT-based smart building network communications as well as device authentication. The report introduces software-based solutions built with machine learning and blockchain technology which work effectively in office complexes and smart residences fitted with connected elements like **HVAC** systems heating ventilation and air conditioning and lighting units access control sets and security cameras.

The exploratory analysis targets software-based defense solutions because hardware design and physical security components will not be examined. Limited availability of resources and time forced the developers to conduct solution evaluations on a test simulation environment instead of real-life smart building applications. The framework undergoes practical deployment analysis which studies ethical aspects and legal frameworks together with organizational boundaries to ensure responsible design for operational readiness.

Overview of the Methodology

This research adopts design-science along with experimental methods to create and test a combined cybersecurity framework which addresses smart buildings. This research first reviewed numerous publications about **IoT security problems** to identify new detection methods which integrate AI systems with blockchain authentication technology.

Following this research the architects developed a security framework containing two distinct layers.

1. **The intrusion detection system** employs artificial intelligence techniques to examine IoT network traffic before detecting unordinary data patterns.

2. **Authentication through Blockchain** operates through an exclusive Ethereum network which manages unalterable decentralized device authentication parameters.

We developed the prototype using Python programming language as the base. The IDS implemented supervised learning techniques **Random Forest, Decision Tree, Naïve Bayes and K-Nearest Neighbors** which received training on labeled IoT traffic information. A proprietary blockchain system supported the security process by using smart contracts to complete safe device enrollment and confirmation operations.

The integrated system allowed the IDS to authenticate devices through blockchain protocol while also detecting abnormal behavior. The system can update blockchain records to indicate device status changes when an intrusion detection occurs.

Evaluation included:

- **Offline testing:** Benchmark intrusion datasets to measure IDS accuracy, false positives, and latency.

- **End-to-end testing:** Simulated smart building scenarios involving rogue or compromised devices.

Performance was compared to traditional approaches lacking AI or blockchain features, showing notable improvements in detection accuracy, response speed, and authentication robustness.

Structure of the Report

The report divides its content into sections that follow the research procedures and resulting findings. An analysis of published academic work about IoT security in smart buildings appears in **Section 2** which depicts identified vulnerabilities and existing solution constraints and new research approaches such as AI and blockchain methods. The research presents important unfilled areas that make the current investigation meaningful. The research design with its proposed solution architecture together with implementation tools (machine learning algorithms alongside blockchain platform) along with their supporting rationales appears in **Section 3:** Methodology and Ethics. The section addresses both ethical concerns from research investigations although it explains data handling practices and virtual attack simulation protocols. Application describes the solution implementation that demonstrates how the IDS integrated with blockchain and provides information about system evaluation and testing processes. The **section** describes various implementation practical obstacles and the methods used to solve them. The experimental data and assessments of machine learning model performances along with blockchain authentication strength and conventional method comparisons are reported in **Section 5:** Results and Analysis. Researchers perform an analysis of the results to derive meaningful interpretations about their underlying

significance. In **Section 6** the report analyzes the findings more extensively to evaluate benefits along with innovative aspects and restrictions of the created solution. The paper explores how AI and blockchain deployments for smart building security impact industrial practices alongside their possible legal circumstances and social considerations. In the final section the research makes its main contributions clear while giving professional recommendations with a list of directions for continued scholarly work based on this research. The document generates **References** to all quoted materials while presenting any supplemental data in one or more Appendices which comprise technical information along with additional code and data points.

Literature Review

IoT Security Challenges in Smart Buildings

Smart building management systems now face increased cybersecurity challenges because of their integration with IoT technology. The operational systems utilize linked sensors with controllers to control lighting **HVAC** and access control functions that leave extensive exposure which allows single device weaknesses to compromise overall system security. The analysis of Shodan database in 2023 detected several building automation systems facing severe **security risks** because operators left default passwords unchanged along with network ports open to potential attackers.

The research by Dos Santos et al. (2019) showed how bad actors exploit IoT devices like HVAC sensors to gain access to essential building infrastructure which links digital and physical security risks. Relevant industry investigations repeatedly identify weak authentication together with encryption deficiencies and software insecurities and limited device capabilities as primary security obstacles. The limited processing capability of IoT devices and fog systems creates barriers to effective encryption security which exposes systems to hijacking incidents and frees up important data for leakage according to Liu et al. (2018).

Most IoT communications transmit data without encryption protocols which makes them easily readable during transmission. Smart building networks provide attacker access that spreads across interconnected devices since a single compromised system can jump from one device to another thus escalating low-level entry into widespread compromises. For instance, hackers could use a compromised lighting hub to breach door controllers. Advanced **scalable cybersecurity solutions** need development for smart building IoT environments because the current systemic weaknesses demonstrate that new protective measures are essential.

Common Attack Types Targeting Smart Building IoT Networks

Smart building IoT systems are exposed to several distinct cyberattack vectors due to their interconnected nature and often inadequate security configurations:

- **Device Credential Attacks**: Many IoT devices retain factory-default or hardcoded credentials, making them easy targets. Once accessed, attackers can reconfigure device behavior or conscript them into botnets, posing serious operational risks.

- **Malware Injection & Remote Takeover**: Lightweight, rarely-updated operating systems leave IoT devices open to exploitation. Malware such as *Mirai* exploited weak credentials to launch massive DDoS attacks. In a smart building, compromised devices like security cameras could flood networks with traffic or generate false data, disrupting operations.

- **Man-in-the-Middle (MITM) & Eavesdropping**: Unencrypted communications allow attackers to intercept, alter, or block messages between devices and controllers. MITM attacks may manipulate commands (e.g., unlocking doors), while passive eavesdropping can reveal sensitive data like occupancy patterns or surveillance feeds.

- **Denial-of-Service (DoS) Attacks**: Adversaries can overload IoT devices or networks using flooding techniques or spectrum jamming (e.g., against Zigbee/Bluetooth). Distributed attacks can sever cloud connectivity or render critical services like lighting or HVAC inoperative.

- **Unauthorized Physical Access**: Devices in accessible areas may be tampered with or reset, allowing attackers to reconnect them to malicious networks. This highlights the importance of strong network-level authentication to prevent rogue device integration.

These attack types collectively expose the critical need for multi-layered defense strategies in IoT-powered smart buildings.

Empirical Evidence of IoT Vulnerabilities in Smart Buildings

Available real-world research demonstrates the extensive prevalence of smart building IoT security threats. The research by Puche Rondon et al. (2020) established that Enterprise systems operated with default passwords and exposed ports because of careless configuration during installation and inadequate security preparation. Research on smart home devices demonstrated that insufficient system security leads to the disclosure of personal information and permits tracking of occupants.

These security gaps create openings for attackers to conduct various **litigations** including personal data interceptions as well as operational system disruptions through **HVAC** control disruption or alarm deactivation. Current evidence demonstrates an immediate requirement for complete authentication systems together with safe configuration protocols and instant monitoring solutions to protect against these ongoing dangers.

Limitations of Traditional Security Measures

Current security methods for networks are ineffective for protecting IoT-heavy environments specifically smart buildings. The security methods depend on fixed regulations while maintaining firm boundaries between networks yet these principles fail to work in IoT environments characterized by **fragmented** operations. Low-power IoT devices cannot support common antivirus solutions nor use standard protocols because of their limited functionality which enables attackers to bypass traditional security frameworks.

Users need to depend on security controls that operate from a single centralized location which represents a critical weakness. An **authentication server** serving as a central authority becomes susceptible to attacker interference when overwhelmed (for instance through DDoS) thus allowing attackers to distribute fabricated credentials and impede access for the entire network. The absence of redundancy in this critical device creates substantial danger in UTC environments that demand continuous operation and data protection.

In 2011 Roman, Najera, and Lopez introduced the essential need for IoT protection to depend on new security paradigms beyond conventional IT approaches. Research from Roman et al. (2011) proved that endpoint protection along with manual alert monitoring systems are inadequate for IoT environments due to the high number of basic unmonitored devices. Present-day smart buildings cope with operational challenges because their deployment of thousands of IoT nodes exceeds the handling capacity and data volumes of typical security systems.

The main problem with traditional **IDS/IPS** tools occurs from their excessive identification of false threats in IoT environments with dynamic changes. Updates made through firmware affect device functionality in ways that may generate incorrect alarm alerts. According to Manky the high volume of false alarms created by such noise triggers security team desensitization which leads them to ignore genuine security threats. The diversion of resources toward harmless anomalies becomes an expensive issue when it happens in smart buildings because genuine intrusions can easily become hidden. Multiple issues emerge due to rule set updates across various IoT devices because they demand both extensive domain-specific knowledge that may be unavailable to developers.

In today's IoT environment the basic security techniques (such as strong passwords with firewalls) fail to address the distinctive vulnerabilities that IoT devices present. The smart building environment exposes severe limitations of conventional security solutions when it comes to identifying new and zero-day security vulnerabilities. The minimal resources of devices within IoT infrastructure prevent them from sustaining heavy security automation systems. The practice of centralized authentication creates essential failure locations throughout the system. **False alerts** in excessive quantities result in reduction of system trust alongside diminished operational effectiveness. IoT requires security solutions which incorporate intelligent distributed models for operation. The latest research recommends AI threat detection systems and blockchain trust frameworks to resolve IoT's security gaps because these solutions are discussed throughout the following sections.

AI and Machine Learning for Intrusion Detection

Machine learning under Artificial Intelligence has proven to be an essential tool which enhances intrusion detection capabilities in IoT networks. Current IDS powered by ML track system normal operation patterns to detect abnormal activities which signal possible intrusions. Keeping device behaviors simple or not is crucial for smart buildings since it helps forensic examination in situations where normal activities overlap with possible malicious activity. IDS frameworks of today rely heavily on machine learning approaches between supervised classification that requires attack-labeled data training and unsupervised anomaly detection without explicit signature attacks.(Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers).

In the context of IoT, several studies have benchmarked various ML algorithms for intrusion detection effectiveness. Common algorithms include Decision Trees, Random Forests, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Naïve Bayes, Neural Networks, and ensemble methods. A key finding across the literature is that **ensemble methods and tree-based classifiers often perform exceptionally well for IoT intrusion detection tasks**. For example, Ashraf et al. (2025) built an IoT intrusion detection system using the BoT-IoT dataset (a comprehensive IoT network attack dataset) and evaluated seven machine learning classifiers (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers). They reported that **Random Forest** was the most robust classifier, achieving about 99.2% detection accuracy on IoT traffic (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers). This high accuracy is attributed to Random Forest's ensemble nature – by aggregating decisions from many trees, it captures complex patterns in network features and is resilient to noise. In the same study, Naïve Bayes also performed well (second best with ~98.8% accuracy) (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers), likely because the conditional independence assumptions of Naïve Bayes happen to fit the dataset's feature distributions to a good extent. By contrast, simpler classifiers like KNN were found to be less effective in that environment.

Another study by Mahmud et al. (2024) compared five machine learning algorithms (KNN, Decision Tree, Random Forest, Gradient Boosting, and AdaBoost) for IoT attack detection after applying feature selection techniques (Optimized IoT Intrusion Detection using Machine Learning Technique). Their results similarly showed Random Forest achieving the highest accuracy (~99.39%), whereas KNN had the lowest (~94.84%) (Optimized IoT Intrusion Detection using Machine Learning Technique). The KNN classifier's relatively lower performance is understandable given the high dimensionality of network data and the heterogeneity of IoT attack types – distance-based methods struggle as irrelevant features or scaling issues can reduce their precision. The consistently strong performance of Random Forest in these studies underscores that ensemble learning can effectively handle the complexity of IoT traffic, which often involves dozens of features (packet statistics, protocol flags, etc.) and nonlinear relationships that single decision trees or linear models might miss.

Through continuous learning modes machine learning both achieves high accuracy while reducing occurrences of false positives. Machine learning models operate better than threshold-based detectors because they can learn through training which patterns are normal to smart building behavior. This allows ML models to identify genuine attacks from typical anomalies. ML-based IDS functionality accepts user feedback to generate better accuracy through time which accelerates its "learning" of false alarms alongside missed detections. The article by Reshi and Sholla (2024) focuses on the advantages that AI provides for threat detection.

Speed represents another major benefit that machine learning brings to intrusion detection operations. The speed at which Decision Trees along with other ML algorithms performs data classification reaches milliseconds which supports its implementation during real-time or live network conditions. Because Naïve Bayes works with basic probabilistic math and feature frequency counting it functions rapidly for processing streaming data in real time. IA-driven IDS uses its high-processing capabilities to manage sensor and actuator building data streams before it generates automatic attack detection responses. Prototypes of real-time IoT IDS reveal their capability to detect attacks while they happen through optimized data pipelines and fast classifiers when operating on the BoT-IoT dataset in real-time mode. (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers) (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers).

Deploying ML for IDS in smart buildings presents several challenges. The quality of training data is critical, as models require representative examples of both normal and malicious traffic. Public IoT security datasets like BoT-IoT, TON_IoT, or IoT-23 are commonly used, but a model trained in one environment may not transfer well to another with different device types or network patterns. To address this, some researchers explore unsupervised anomaly detection, which can identify deviations without labeled attacks, though it may result in higher false alarms initially.

Feature selection is another challenge. IoT network data includes numerous features, and including too many can lead to slower algorithms and overfitting. Feature importance analysis, such as using Random Forests, can help identify the most indicative metrics of attacks, such as TCP flags, payload sizes, and device IDs, enabling more efficient and effective detection (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers). By selecting the top N features, the IDS can reduce complexity and focus on the most relevant indicators. In smart buildings, features like sudden spikes in sensor message frequency, communications at odd hours, or packets destined to unusual external endpoints might be strong signals of compromise.

In review, machine learning introduces a level of **intelligence and automation** to intrusion detection that is well-suited for smart buildings. It can detect previously unseen attacks by learning normal vs. abnormal patterns, adapt to changes in the network, and do so with low latency. Empirical evidence shows high detection rates (often >95% accuracy) are achievable with the right algorithms (Making a Real-Time IoT Network Intrusion-Detection

System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers) (Optimized IoT Intrusion Detection using Machine Learning Technique). By dramatically cutting down false positives, AI-based IDS also alleviates the burden on human analysts, allowing them to focus only on credible threats (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). This is critical in facilities management contexts where dedicated cybersecurity staff may be minimal – an autonomous IDS serves as a tireless "sentry" watching over the myriad IoT devices. For these reasons, integrating AI-driven IDS capabilities is a cornerstone of our proposed framework for smart building security. In the next section, we examine the other pillar of our approach: blockchain technology, which addresses the identity and trust issues not solved by intrusion detection alone.

Blockchain for IoT Security

Blockchain technology, originally popularized by cryptocurrencies, has seen increasing adoption in IoT security for its ability to enable **decentralized trust and tamper-proof record-keeping**. In essence, a blockchain is a distributed ledger maintained by a network of nodes following a consensus protocol, which makes it extremely resistant to unauthorized modifications. For IoT systems like those in smart buildings, blockchain offers a way to remove reliance on a single trusted authority by distributing trust among multiple stakeholders or devices ([2203.08901] Blockchain as privacy and security solution for smart environments: A Survey). This characteristic directly addresses the centralized authentication vulnerability mentioned earlier: instead of a single server deciding who or what is trusted on the network, a blockchain can provide a **distributed authentication mechanism** where each device's identity or credentials are verified by consensus and stored immutably on the ledger (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx).

The authentication procedures of IoT systems receive improvements through blockchain capabilities which manage devices and verify their identities. An IoT device receives its own cryptographic identity that gets blockchain registration to authorize network access only for verified devices. The unchangeable nature of blockchain ensures that attackers cannot modify these records because it needs the collective agreement of most controlled nodes that operate under a permissioned blockchain consensus mechanism. The authors Zhao et al. (2023) pointed out that permanent device recognition systems protect networks from unapproved system entry. When unauthorized devices attempt to pretend as authorized ones the middleware using blockchain technology can block such interactions which enable detection of suspicious activity despite compromised private keys.

Blockchain technology strengthens IoT data integrity while ensuring visible record-keeping in these systems. The blockchain records critical interactions such as control commands and sensor readings through a tamper-resistant audit trail because IoT data can execute physical operations. The system detects fabricated data and replays because it validates information stored in the ledger. A valid door unlock command would exist in check-block records on the blockchain network for verification purposes and unidentified commands will be denied. The combination of security and privacy improvements in IoT communication comes from blockchain technology which eliminates central brokers according to Ebrahim, Hafid, and Elie (2022).([2203.08901] Blockchain as privacy and security solution for smart environments: A Survey). In a smart building, this could translate to peer-to-peer secure

messaging facilitated by blockchain (or its smart contracts), where no single node can secretly alter messages without consensus noticing.

The intelligent building market benefits from permissioned blockchains including Hyperledger Fabric together with private Ethereum networks because they provide access control features alongside increased processing speed and faster execution times than public blockchain systems. These deployments grant blockchain administration rights to local servers or gateways that enable transaction approvals only from authorized systems as well as lightweight IoT nodes. Proof-of-work mechanisms are eliminated to achieve higher operational efficiency which allows consensus algorithms consisting of **PBFT** or **RAFT** to process transactions inside milliseconds to seconds thus making device authentication possible. Sensor data in real time passes outside of the blockchain system yet all essential control events must be stored on the blockchain database.

The **blockchain** system enables self-enforcing code called smart contracts to provide distributed control functions. Smart contracts apply limitations that control alarm activation function only when devices present valid credentials combined with required clearance levels. Security policies manage automatically through these contracts which maintain transparent and consistent operations. Waheed and He (2023) developed a system which integrates blockchain smart contracts together with machine learning to build dynamic IoT system trust assessments. The model implemented data sharing exclusively during moments where blockchain smart contract regulations corresponded to AI-derived trust evaluations thus unifying blockchain integrity functions with AI adaptive capabilities.

The blockchain platform enables cooperative security practices between different stakeholders existing in structures like smart buildings and smart cities. The platform functions as a collective security base that allows businesses to select uniform security guidelines. Sarhan et al. (2022) developed a hierarchical blockchain-fed entity network which enables secure intrusion detection insight sharing among members of the system. The validity of model updates is verified by smart contracts which establishes reliable threat intelligence sharing between different organizations. The underlying principle behind this project applies to multiple buildings because blockchain enables distributed networks to transmit verified security alerts about detected malicious device behavior thus strengthening collective defense capabilities.

Of course, **blockchain integration is not a panacea** and comes with its own considerations. There is an overhead to maintaining the ledger, and devices need a way to interface with the blockchain (directly or via gateways). Lightweight blockchain clients or off-chain aggregation might be necessary for very constrained IoT nodes. Researchers like Zhang et al. (2023) have worked on *lightweight blockchain security models* tailored for IIoT (Industrial IoT), combining AI techniques to keep performance high (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). In their approach, the blockchain was optimized to reduce execution time to ~0.6 seconds for security operations and achieved an impressive 99.7% detection performance by using an AI mechanism alongside blockchain authentication (An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems | Journal of Cloud Computing | Full Text) (An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems | Journal of Cloud Computing | Full

Text). This indicates that with careful design (e.g., using efficient consensus and perhaps offloading heavy computations), blockchain can be adapted to IoT without unacceptable performance penalty.

While blockchains offer strong resistance to data tampering, they are not immune to attacks. Threats such as Denial-of-Service (DoS) or exploitation of flawed smart contract logic remain concerns. In smart building deployments, using a permissioned blockchain on secure, well-maintained hardware mitigates these risks. Additionally, rigorous testing of smart contracts is essential to prevent vulnerabilities in the automation layer.

In summary, blockchain contributes the **trust and integrity layer** that IoT networks like those in smart buildings are missing. It decentralizes authentication (removing single points of failure), ensures integrity of device identities and critical data, and provides an audit trail for accountability (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) ([2203.08901] Blockchain as privacy and security solution for smart environments: A Survey). This approach complements intrusion detection by proactively preventing unauthorized interactions—embedding security policies directly into the network. While IDS detects ongoing threats, blockchain can block them at the outset. Our research harnesses blockchain for robust IoT device authentication and authorization, while AI/ML intelligently identifies behavioral anomalies. This integrated strategy capitalizes on the strengths of both technologies, as explored in the following sections.

Integrated AI-Blockchain Solutions and Research Gaps

Integrating AI and blockchain for IoT security has gained momentum due to their complementary strengths: AI (particularly ML-based IDS) excels at behavior analysis and threat detection, while blockchain provides secure identity management and immutable event logging. Together, they enable decentralized, intelligent security enforcement.

1. **AI-Enhanced Blockchain Models:**
   Zhang et al. (2023) developed a lightweight blockchain security model for IIoT that integrated a neural network for device authentication. This fusion enhanced detection accuracy and efficiency, enabling real-time decisions within the blockchain framework.(An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems | Journal of Cloud Computing | Full Text)

2. **Blockchain-Supported Collaborative IDS:**
   Sarhan et al. (2022) introduced HBFL, a hierarchical blockchain-based federated learning system. Each local IDS model shares learned threat patterns via blockchain, enabling global improvement without compromising data privacy. Blockchain guarantees the integrity of shared updates.([2204.04254] HBFL: A Hierarchical Blockchain-based Federated Learning Framework for a Collaborative IoT Intrusion Detection) ([2204.04254] HBFL: A Hierarchical Blockchain-based Federated Learning Framework for a Collaborative IoT Intrusion Detection)

3. **Privacy-Preserving AI:**
   Bui et al. (2024) proposed a blockchain-coordinated deep learning system using homomorphic encryption. This setup processes encrypted data for attack detection, preserving privacy without sacrificing accuracy. Though computationally intensive, it reflects emerging trends toward encrypted AI and secure coordination.([2412.13522] Privacy-Preserving Cyberattack Detection in Blockchain-Based IoT Systems Using AI and Homomorphic Encryption) ([2412.13522] Privacy-Preserving Cyberattack Detection in Blockchain-Based IoT Systems Using AI and Homomorphic Encryption)

4. **Hybrid IDS with Blockchain:**
   Alkadi et al. (2024) showed how ML and blockchain together enhance IDS scalability and accuracy in IoT networks. Similarly, Waheed & He (2023) demonstrated a system where ML detects anomalies while blockchain controls device access, ensuring private and authorized communication.

These studies highlight a growing convergence of AI and blockchain in cybersecurity, each addressing aspects like privacy, performance, or collaboration. However, notable gaps remain—especially for real-world smart building applications:

1. **Domain-Specific Implementation:**
   Most integrated models have been tested in generic or industrial IoT settings, not smart buildings. These environments feature unique devices and protocols (e.g., BACnet, Zigbee) requiring tailored solutions. Our research adapts AI-blockchain integration specifically for smart building networks, using domain-relevant data and threat scenarios.

2. **Complexity vs. Practicality:**
   Advanced solutions like homomorphic encryption are often impractical for building IT teams. Our project prioritizes deployability by combining standard ML algorithms with a private Ethereum network, showing that simpler setups can still deliver meaningful security gains.

3. **Real-Time Performance:**
   Many studies ignore real-time constraints. In smart buildings, delays in functions like door access are unacceptable. Our framework evaluates latency and processing overhead to ensure security enhancements don’t disrupt automation.

4. **Automated Response:**
   While detection is well-researched, automated mitigation is not. We explore how smart contracts can respond to alerts—revoking credentials, issuing IoT commands, or logging alerts—enabling active defense, not just passive detection.

5. **Testing Against Combined Attacks:**
   Most literature evaluates isolated threats. Our evaluation simulates chained attacks—unauthorized device access followed by malicious activity—to test the robustness of the integrated AI-blockchain defense.

Our study introduces a practical implementation that uses AI and blockchain concepts to address IoT security whereas it focuses specifically on smart buildings. The main contribution consists of implementing machine learning IDS alongside blockchain authentication for building networks while conducting experimental validations. The research fills a needs gap by offering an implementation standard which serves as a blueprint for future deployments. This section presents the methodology that describes development process alongside design decisions, implemented tools and ethical aspects.

Methodology and Ethics

Research Design Overview

This research follows a **design science** methodology, focusing on building and evaluating a novel artifact – in this case, an integrated AI-blockchain security framework for IoT in smart buildings. The project began with problem identification (as described in the introduction) and a thorough requirements analysis derived from the literature review: the system needed to provide real-time intrusion detection, decentralized authentication, low false alarms, and seamless integration with typical building networks. With these requirements in mind, we designed a **Hybrid Security Framework** composed of two synergistic subsystems:

- **AI-Driven Intrusion Detection System (IDS):** A software component that monitors network traffic data from IoT devices and uses machine learning models to detect abnormal patterns indicative of cyber attacks or unauthorized behavior. This IDS operates continuously and in real-time, analyzing data packets or communication flows within the building's local network.

- **Blockchain-Based Authentication and Logging:** A private blockchain network that maintains a secure ledger of device identities and their authentication credentials. All IoT devices in the smart building must register on this blockchain. The blockchain thus serves as the ground truth for which devices are trusted. It also records critical security events (e.g., device registrations, detected intrusions, access control

decisions) in an immutable log. Smart contracts on the blockchain enforce authentication rules (e.g., only registered devices can send control commands).

The system architecture features IoT devices linking to a network gateway while placing the IDS module at a suitable network point for traffic monitoring. The network for blockchain operates on designated servers together with present servers which provide cryptographic key access for IoT devices. The blockchain checks device identities both when devices join the network and when they try to exchange data.

The sequence of operations is as follows:

1. **Device Enrollment**: New IoT devices are registered on the blockchain with their ID and potentially a public key. Devices may be assigned roles or permissions via smart contracts (e.g., a camera labeled as “video_sensor”).

2. **Real-Time Monitoring**: The IDS inspects IoT traffic using machine learning to detect anomalies. Initially, it may run in learning mode to establish a baseline.

3. **Authentication Checks**: Before significant actions (e.g., control commands), devices are authenticated through blockchain verification. For instance, an HVAC controller queries the blockchain to ensure the thermostat’s ID is valid before executing commands.

4. **Anomaly Detection and Response**: If anomalous traffic is detected, the IDS raises an alert, and the blockchain logs the event. A smart contract may update the device’s permissions (e.g., revoking access). The system can also trigger network mechanisms to isolate the device, ensuring rapid action and a trusted record.

5. **Ongoing Learning**: Alerts and data can improve IDS models, and the blockchain log could potentially be used for sharing information across facilities, although this is outside the current scope.

Security measures in this system work together through defense-in-depth architecture. When attackers register harmful devices through stolen administrative logins on the blockchain then IDS systems can detect such improper actions. Attackers who attempt to bypass detection through simulation of regular traffic need an authentic device identification to perform their acts targeted at the blockchain. The blockchain system stops unauthorized activities when device identities are absent. This method makes sure that every security element supports and extends the protective capabilities of all other elements.

Following these steps was used to build and assess the security design: (a) selecting IDS and blockchain system tools and technologies, (b) building IDS machine learning models with training and validation steps, (c) configuring blockchain environments and writing device management smart contracts, (d) integrating system components through software application programming interface (API) and (e) conducting tests with simulated realistic

attacks. Experimental validity through known datasets and correct scenarios received due consideration along with ethical practice of data appropriateness and attack simulation within controlled limits.

Tools, Technologies, and Techniques

Machine Learning Model Development:

The IDS was implemented in Python using Scikit-learn, selected for its reliable and accessible machine learning tools. Four algorithms were evaluated: Random Forest (RF), Decision Tree (DT), Gaussian Naïve Bayes (GNB), and K-Nearest Neighbors (KNN). These provided a balance of interpretability, complexity, and performance trade-offs.

- **Random Forest (RF):**
  We anticipated RF to yield high accuracy due to its ability to model nonlinear relationships and reduce overfitting through averaging. As expected, it performed well on high-dimensional network traffic data and helped identify critical features—such as packet rate and payload size—via built-in feature importance metrics.

- **Decision Tree (DT):**
  DT was chosen for its transparency and efficiency. Its clear decision paths proved useful in explaining results to stakeholders. Though prone to overfitting, we mitigated this by tuning tree depth, which allowed us to maintain real-time performance and interpretability.

- **Gaussian Naïve Bayes (GNB):**
  GNB served as a lightweight, fast baseline. Its probabilistic approach and speed made it suitable for rapid detection in high-throughput IoT environments. While less accurate than RF or DT, it proved effective in scenarios prioritizing speed over complexity.

- **K-Nearest Neighbors (KNN):**
  KNN was used to test a non-parametric approach. Its ability to detect localized anomalies was valuable, particularly in spotting outlier behavior in device traffic. To manage its computational cost, we used small *k* values and applied dimensionality reduction when necessary, allowing us to tune sensitivity based on the use case.

We evaluated whether sophisticated methods such as Neural Networks or SVMs would be more appropriate than the chosen algorithms. The four explained algorithms received our preference because our dataset requirements and the need for understandable solutions and simple deployment. Our development consisted of standard procedures between training-testing data partitions while applying cross-validation on training sets for model selection and performing feature normalization particularly for KNN distance operations and appropriate treatment of categorical factors.

Dataset and Feature Engineering

The IDS received training and evaluation through BoT-IoT dataset from UNSW Canberra's Cyber Range Lab that contained both benign and attack traffic (DDoS, scanning, ransomware). Because of its zeroing in on network analysis of IoT devices this system proved appropriate for smart building security needs.

Building environment-specific IDS features were chosen including flow_duration along with packet_interarrival_time and payload_bytes and IP-level communication attributes. The traffic patterns between devices (such as sensor-to-server flows) became detectable using port numbers together with device-specific addresses.

The implementation of feature selection applied Pearson correlation and initial Random Forest feature importance for decision-making. The applied dimension reduction boosted classification performance while simultaneously improving training speed because redundant or uninformative inputs were eliminated.

The dataset was split into **70% training and 30% testing**, with a small validation set held out for tuning. We used **5-fold cross-validation** to tune hyperparameters, including tree counts in RF (50, 100, 200), depths in DT (5–20), and k-values in KNN (3–15). Our optimization focused on **F1-score** to balance precision and recall, ensuring both accurate detection and minimal missed threats. We also monitored training time to ensure model retraining would remain feasible.

To address **class imbalance**, we either downsampled dominant attack classes or applied class weighting during training. Evaluation metrics included Accuracy, Precision, Recall, F1-score, and **False Positive Rate (FPR)**—with particular focus on achieving high recall and low FPR to ensure practical deployment without overwhelming users with false alarms.

We also benchmarked **training time and prediction latency** to confirm real-time constraints were met.

Blockchain Platform

The IDS received training and evaluation through BoT-IoT dataset from UNSW Canberra's Cyber Range Lab that contained both benign and attack traffic (DDoS, scanning, ransomware). Because of its zeroing in on network analysis of IoT devices this system proved appropriate for smart building security needs.

Building environment-specific IDS features were chosen including flow_duration along with packet_interarrival_time and payload_bytes and IP-level communication attributes. The traffic patterns between devices (such as sensor-to-server flows) became detectable using port numbers together with device-specific addresses.

The implementation of feature selection applied Pearson correlation and initial Random Forest feature importance for decision-making. The applied dimension reduction boosted classification performance while simultaneously improving training speed because redundant or uninformative inputs were eliminated.

Smart Contract Development

We developed a Solidity smart contract named **DeviceRegistry** to manage device registration and trust status on the private Ethereum network. The contract's key functions are:

- **registerDevice(address deviceAddr, string deviceType, bytes32 pubKeyHash)**:
  Called by an authorized admin, this function registers a new device by storing its blockchain address, type label, and a hash of its public key or certificate. The device is initialized with an "active/trusted" status. Access control is enforced using require.

- **revokeDevice(address deviceAddr)**:
  This function allows either the admin or the IDS (granted special permissions) to mark a device as revoked, indicating it is no longer trusted.

- **isDeviceTrusted(address deviceAddr) public view returns (bool)**:
  Used by external components, including our authentication gateway, to verify whether a device is registered and not revoked.

The contract emits events such as **DeviceRegistered** and **DeviceRevoked**, which are logged on-chain to maintain a tamper-proof audit trail of all administrative actions.

Devices are identified by Ethereum addresses—either their own or one derived from their public key. Since IoT devices typically lack full Ethereum clients, we simulated device interactions in our environment using scripts that generated signed transactions, with a gateway or security server acting on the devices' behalf for blockchain operations.

Integration via APIs

The deployment of Python web3.py library allowed our IDS system to directly connect with the deployed smart contract. The IDS activates revocation of a device through an authorized account whenever it detects unusual behavior after which it performs an on-chain update to indicate a modified device status. The IDS system allows an extra step where it checks the device trust status through the **isDeviceTrusted function** before starting the data flow classification process.

The enforcement mechanism for real-time blockchain policies required us to develop a Node.js middleware script that functioned as a gateway or SDN controller. The script retrieves source address status from the smart contract to identify untrusted devices and drops their corresponding packets. This implementation enables the enforcement of access control at network level when it relies on blockchain state information.

The execution of the isDeviceTrusted method required only brief execution times of about **fifty milliseconds** because it accesses blockchain data stored on the device. The revokeDevice transaction required 1–2 seconds of confirmation duration which showed it met the requirements for device isolation speed. The blockchain operates as a trust authority while network-level access control continues to function immediately through rules that validate trust status.

Model Training and Testing Procedures

The team established the IDS models together with blockchain before starting the testing phase through offline evaluations. The research evaluated each machine learning model by training it with historical **BoT-IoT** records followed by independent testing of a reserved dataset without blockchain capabilities for baseline detection estimation.

All of the analyzed models produced confusion matrices. Random Forest combined with Decision Tree demonstrated superior performance compared to other models and KNN had some delays in both precision and execution speed. Following model evaluation we chose **Random Forest** as the main detection method for the integrated system that provided both accurate results and fast inference through tree depth control. The additional models existed as backups that could supplement the primary detection system.

Testing of the IDS involved analyzing a smart building dataset that contained both regular device traffic and attack simulations starting from device compromise events. The developed models showed robust generalization ability because they identified building-relevant threats through detecting traffic surges along with irregular communication patterns from the BoT-IoT training dataset.

Blockchain Functional & Integrated Simulation Testing

We used Truffle suite to validate the smart contract through tests that checked proper device registration including revocation permissions and the isDeviceTrusted logic. Before full integration we performed manual tests by using blockchain queries to verify changes while we registered devices and revoked access.

We performed complete integration tests between the IoT devices within a **Docker-based simulated IoT network**. The experiment used four IoT devices including one that simulated a malicious attack while another one operated the IDS along with the blockchain client and the third and final container functioned as the Ethereum node to host our smart contract. The benign devices produced UDP packet transmissions at periodic intervals yet the malicious device triggered pre-scripted cyber attacks including flooding and data unauthorized transmission.

Anomaly detections by the IDS led it to activate web3.py which automatically initiated revocation procedures. The blockchain updated within ~1 second. Our software framework started applying the isDeviceTrusted query which activated packet blocking to contain the dangerous device immediately.

The system recorded how quickly detection took place immediately after detection and revocation and enforcement required about one second to complete. During various **intrusion** testing conditions the IDS actively detected anomalies through sensitivity controls which simultaneously reduced false reactions without compromising its speed of response. The system's ability to detect and handle security threats in smart buildings received verification through these test results.

Comparative Analysis

To assess the advantages of our framework, we ran comparative tests against a baseline **traditional setup**—basic authentication and a static firewall, with no IDS or blockchain integration. In this scenario, simulated attacks (e.g., packet floods) continued unchecked until manual intervention, clearly demonstrating the **superior responsiveness** of our system.

We also tested for **false positives**, such as legitimate traffic spikes from firmware updates. In one instance, the IDS misclassified a benign event, triggering a temporary revocation. However, the smart contract's **audit trail** made the cause clear, and the admin promptly reinstated the device. This highlighted two key strengths: the importance of fine-tuning to reduce false alarms, and the blockchain's **transparent event logging**, which facilitated fast recovery and informed model retraining.

Ethical Considerations

Ethical conduct was maintained throughout the research. Firstly, **no real user data or confidential building data** was used in our experiments. The BoT-IoT dataset employed for training and testing is a publicly available, synthetically generated dataset created in a controlled lab environment specifically for security research (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers). It does not contain personally identifiable information (PII) or sensitive content; it consists of network flow records and packet features, which are anonymized and abstract. Thus, using this dataset did not pose privacy risks. Similarly, other data generated (like synthetic smart building traffic patterns) contained no real personal or corporate information – they were dummy sensor readings and simulated events.

The evaluation conducted all digital attacks through packet floods and device-level anomalies within autonomous local Docker networks. The experiment feasible area excluded physical IoT devices and production systems because this protected against actual damage. The laboratory implementation of our framework made it possible to rigorously test it while keeping external systems unaffected.

The team recognized the moral considerations that arise when using AI for security system deployment. False positive detections by IDS systems can result in unintentional service denial for valid devices which presents severe risks within actual medical applications that use IoT systems. We adjusted the models with caution while implementing verification procedures to demand manual administrator checks before sensitive vulnerability removals. Future deployments with real users and production devices need to incorporate fail-safe mechanisms along with human oversight because of ethical reliability requirements.

The IDS system ran exclusively with technical network characteristics including traffic rate and protocol types while avoiding personal or identity-based data use for its decision-making process. The technical design approach limited the system to analyze behavior patterns independently from discrimination-based or biased assessments.

We identified an issue with blockchain immutability causing privacy problems on its platform. The system utilizes non-personal technical identifiers as its on-chain data such as device

hashes while only granting access to authorized accounts that can check trust status. The smart contract design excluded sensitive metadata from public view because it did not contain user-linked actions or device address lists.

We followed research ethical standards while keeping IRB approval irrelevant because our study involved no human participants or sensitive personal information thus we showed transparency about our methods and used data responsibly and aimed to benefit the cybersecurity industry. Handpicked research materials originate from open-source projects and licensed domains and all exploit methods will stay within the laboratory setting.

We explored the appropriate legal frameworks required for implementing the described framework in practical operational settings. Our system follows regulatory standards through secure audit trail maintenance but requires live deployment operators to focus on GDPR compliance and proper data handling protocols.

The entire methodology followed safe ethical and legal guidelines during its execution. The subsequent part details the deployment strategy of this design which led to performance assessments of the implemented system.

Application: Implementation and Testing of the Solution

Implementation Process and Justification

The implementation of our proposed IoT security framework followed a structured, stage-wise process. Each stage was validated before proceeding to the next, ensuring stability and reliability across the full system. This iterative approach allowed for feedback loops—e.g., insights from evaluation led us to revisit preprocessing or model tuning. The four main stages were:

1. **Data Preprocessing**
2. **Model Training**
3. **Evaluation and Tuning**
4. **Deployment and Monitoring**

This design was based on our methodology plan and evolved naturally as the system matured.

1. Data Preprocessing

Effective machine learning begins with high-quality data. We prepared our dataset—which combined the BoT-IoT dataset with simulated smart building traffic—to reflect both benign and malicious network behaviors.

- **Data Cleaning:**
  We removed corrupted or incomplete records (e.g., flow logs with missing values). Timestamp fields were converted into relative time or time-of-day features rather than being used directly, to prevent overfitting to the timing of specific scenarios.

- **Normalization:**
  To ensure features with different scales didn't disproportionately influence models, we applied normalization techniques (Min-Max or standardization). This was particularly helpful for KNN and neural networks, though less critical for tree-based models like Random Forest.

- **Feature Engineering and Selection:**
  We focused on features indicative of IoT intrusion, including:

    a. Connection duration

    b. Packet and byte counts

    c. Average packet size

    d. Protocol types

    e. Source/destination ports

    f. Standard deviation of inter-arrival times

- Domain knowledge played a key role: for example, IoT devices typically communicate within local networks, so any outbound communication to external IPs was flagged using a binary external_comm feature.
  We also captured behavioral regularities—e.g., periodic reporting—through features like variance in inter-packet intervals. To reduce noise and dimensionality, we used correlation analysis to drop redundant features (e.g., if total bytes and total packets were highly correlated, we retained only one or derived a mean packet size).

- **Label Encoding:**
  For binary classification, all attack types were grouped as class 1 (malicious) and normal traffic as 0. We retained original multi-class labels for extended evaluation but focused initially on the binary anomaly detection problem.

Model Training and Validation

Following preprocessing, we obtained a clean, structured dataset optimized for ML. This step was essential for minimizing false detections and ensuring the models could operate efficiently in real-time settings.

We implemented and trained four classifiers – **Random Forest (RF), Decision Tree (DT), K-Nearest Neighbors (KNN), and Gaussian Naïve Bayes (GNB)** – using Python and scikit-learn. The dataset was split into 70% training and 30% testing, with **5-fold cross-validation** applied on the training data to tune hyperparameters.

Model Implementation Details

- **Random Forest**: Started with 100 trees (n_estimators=100) and full depth (max_depth=None). Through tuning, we found that limiting max_depth to ~10 and keeping 100 trees struck a strong balance between speed and accuracy.

- **Decision Tree**: Tuned max_depth (~8 was optimal) and pruning parameters like min_samples_split. Performance gains beyond this depth were negligible and risked overfitting.

- **KNN**: Experimented with both Euclidean and Manhattan distances (Euclidean performed better). Found **k=5** offered the best trade-off between recall and false positives.

- **Naïve Bayes**: Used **GaussianNB** with default priors, as it performs well on continuous features without extensive tuning.

To address the **class imbalance** (as BoT-IoT contains more attack samples than benign), we used class_weight='balanced', which helped the models pay attention to the underrepresented "normal" class.

Validation on Known Attacks

We confirmed the models could detect specific attack signatures:

- **UDP floods** were consistently identified by RF due to its ability to pick up on traffic bursts.

- **Port scans** were reliably flagged by GNB, which recognized rapid, repeated connection attempts. These validations confirmed that our models generalized attack behaviors rather than memorizing patterns.

We considered ensembling (e.g., stacking DT + NB), but since RF already offered ensemble benefits and showed the best overall performance, we prioritized simplicity and speed.

3. Evaluation and Tuning: We then rigorously evaluated each model's performance on the held-out test set and through cross-validation, refining them for optimal results (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx). The Random Forest achieved an accuracy of about 99% in classifying benign vs. malicious traffic, with a **recall** (true positive rate) around

0.99 and **precision** around 0.98 (F1-score ≈ 0.985). Decision Tree performance was slightly lower (accuracy ~97%), and KNN and Gaussian NB were in a similar range (94–95% accuracy). The slight gap between Random Forest and the simpler models aligns with other studies that found ensemble methods more effective for IoT intrusion detection (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers) (Optimized IoT Intrusion Detection using Machine Learning Technique). Moreover, Random Forest maintained a low false positive rate (~1%), meaning it rarely misclassified normal behavior as an attack. This was critical for our use-case to avoid unnecessary device lockdowns. We tuned the Random Forest to limit its depth (to avoid overfitting) and confirmed via confusion matrix that it still caught all attack instances in the test set (100% recall on known attacks). For KNN, we noted it had a higher false alarm rate (about 5% FPR) when using very low k, so we chose k=5 to smooth out noise (Optimized IoT Intrusion Detection using Machine Learning Technique). The Gaussian NB model was extremely fast but slightly less accurate in some complex attack mixes (its independence assumption sometimes led it astray). Given these outcomes, we selected **Random Forest as the primary detection engine** for the deployed IDS due to its superior accuracy and robustness, while Decision Tree and Naïve Bayes were kept as backup models for ensemble voting or explainability (Decision Tree rules helped interpret what the RF was doing, aiding trust in the model). We also tested the model against the small synthetic smart building traffic set: the Random Forest and DT successfully identified the simulated anomalies (like an unauthorized external connection) and produced no false alarms on purely benign building activity, reinforcing that our feature set generalized well.

4. Deployment and Monitoring

In the final phase, we deployed the trained **Random Forest** model within the IDS and fully integrated it with the **blockchain-based device registry** for live operation.

Real-Time IDS Engine

The IDS was implemented as a Python daemon that continuously sniffed network traffic from the smart building's virtual network (Docker-based). A custom parser converted packet flows into real-time **feature vectors**, mimicking the preprocessing pipeline used during training. These were fed into the loaded Random Forest model to classify traffic windows as benign or anomalous.

- Feature computation used **sliding windows** to maintain context and performance.
- Inference was efficient, classifying **hundreds of flows per second**, enabling real-time detection.

Blockchain Integration

We deployed the **DeviceRegistry** smart contract on a private Ethereum network (1-second block time). All legitimate IoT devices were pre-registered with their addresses and types. During runtime:

- The IDS communicated with a **blockchain client service** (via internal API) upon detecting an anomaly.

This client, using web3.py, submitted revokeDevice transactions:

contract.functions.revokeDevice(device_addr).transact({'from': admin_account})

- The smart contract emitted event logs and flipped the device's trusted flag to false.

Monitoring Dashboard

To provide operational visibility, we created a dual-interface dashboard:

- **CLI logging** for real-time alerts: e.g.,
  "Device 0xabc... flagged by IDS – Revoked at 12:04:35"

- **Web UI** displaying device trust states, pulling live data from the blockchain using web3.js.

Challenges and Mitigations

1. **False Alarms & Data Imbalance**

Some initial false positives (e.g., high-throughput video streams) were addressed by:

- Adding context-aware features (e.g., port-based traffic labeling).
- Adjusting the Random Forest decision threshold.
- Leveraging class_weight='balanced' to reduce over-sensitivity to benign spikes.

2. **IDS–Blockchain Coordination**

Blockchain transactions take ~1–2 seconds to finalize. To avoid IDS slowdowns:

- We implemented a **thread-safe queue** between the IDS and blockchain client.
- Alerts were batched to prevent duplicate revocations for repeated anomaly events.

3. **Performance Optimization**

- **Read-only checks** (e.g., isDeviceTrusted) were <<0.1s and cached when possible.
- Authentication was only blockchain-validated **at session start**, not per packet.
- **Writes** (register/revoke) occurred only during administrative events.

4. **Manual Overrides**

Admins could manually **restore** devices using a smart contract function, enabling rapid recovery from false positives.

5. **Scalability Considerations**

While the prototype handled a small network well, we considered future scaling:

- IDS can be **sharded** across network zones.
- Blockchain can scale with **node clusters** or **layer-2 solutions**.
- RF inference scales linearly with trees/features; lookup and writes remain efficient due to smart contract design.

Solution Testing and Evaluation

We conducted rigorous testing to validate that the system meets its objectives: real-time intrusion detection, unauthorized access prevention, and efficient operation with minimal overhead. This included both functional and performance evaluations covering detection accuracy, false positives, and response time.

- **Unauthorized Device Access Test:**
  We introduced a rogue IoT device not registered on the blockchain. As anticipated, the blockchain-based access control blocked the device immediately. The gateway, upon querying isDeviceTrusted, found no match and dropped the packets. Logs confirmed repeated attempts from the unregistered address were ignored. This validated the blockchain's effectiveness in preventing unauthorized device participation. In contrast, a baseline system without blockchain would have processed those packets, exposing the network to potential harm.

- **Intrusion Detection and Response Test:**
  We simulated a compromise scenario: a registered temperature sensor was scripted to launch a DDoS flood against the central server. The IDS, using the trained Random Forest model, detected the anomaly within ~0.5 seconds. The sensor's traffic showed a sharp deviation from its normal pattern, triggering a high anomaly score. An alert was raised and sent to the blockchain client, which revoked the device by calling revokeDevice. The contract emitted a DeviceRevoked event, and the device's trusted flag flipped to false.

Following this, the authentication layer and associated network controls blocked all traffic from the compromised sensor. The attack was neutralized within seconds, with only a few initial packets making it through—insufficient to cause disruption. Compared to traditional setups where manual intervention is required, this automated response significantly reduced potential damage and downtime.

- **Attack Detection Accuracy**

We tested various cyber-attack scenarios using both our simulation environment and live replay of BoT-IoT dataset segments.

1. **Scanning Attack:**
   A device was scripted to perform a port scan across the network. The IDS identified the behavior—multiple short-lived connections to numerous ports—and flagged the device. The Random Forest model accurately differentiated this from legitimate multi-port communication (e.g., devices contacting a few known services), avoiding false positives. Upon detection, the blockchain revoked the device immediately.
2. **Man-in-the-Middle / Impersonation:**
   We simulated an attacker spoofing a legitimate device's IP to inject false data. Although the packets appeared authentic at the network level, the blockchain authentication failed due to the absence of the correct cryptographic identity. The gateway rejected the packets, effectively neutralizing the impersonation. This test highlighted the strength of combining IDS with blockchain: spoofed traffic that mimics normal patterns may evade detection by the IDS alone but is blocked at the identity layer.
3. **Data Exfiltration Attack:**
   We tested a scenario where a device attempted to exfiltrate large volumes of data to an external server. The IDS flagged the abnormal traffic based on volume and destination, and the device was revoked. Importantly, benign outbound actions (e.g., a software update) were not flagged, demonstrating model precision. Detection was aided by features such as destination recognition—traffic to unknown endpoints was treated as suspicious, whereas traffic to known vendor servers was allowed.

Overall Evaluation

During our controlled testing the IDS maintained a perfect true positive rate which identified every simulated security attack while not producing any incorrect warnings in continuous normal traffic operations. The evaluation results show that real-world deployments will likely yield some false positives however our tests suggest such occurrences would be below one percent. When analyzing clean traffic for 24 consecutive hours the IDS system failed to generate any false detection events. The low number of false alarms makes this detection quality particularly valuable for real-life IDS implementations. The successful performance can be attributed to our selection of pertinent features along with optimal model adjustments during testing periods characterized by reliable smart building IoT traffic stability.

The system's operational performance required minimal resources for its operation. Observations of 10 monitored devices which consumed less than 15% of CPU on a typical machine showed linear behavior indicating that the solution could be solved by distributing workloads or obtaining better hardware devices at greater scales. The implementation of Random Forest used minimal system memory and blockchain operations required minimal processing power. The complete detection and response sequence reached 1–2 seconds in duration: anomalies were detected in less than a second followed by about one second of blockchain revocation operations. The system provided immediate active attack blocking capacity.

The baseline testing included static firewall rules with no automated detection elements. The compromised device started its DDoS assault which remained undiscovered for tens of seconds until human intervention when the target server became unable to handle the attack. The integrated system took instantaneous autonomous action to stop threats from escalating because of its intelligent mechanism.

Testing results proved that our system delivers its principal functions by detecting attacks precisely and responding instantly while avoiding unauthorized system access. The validation of this study confirms how integrating machine learning with blockchain technology enables secure smart building protection systems.

Discussion: Interpretation of Findings

Visual analytics reveal that the AI-blockchain architecture provides substantial improvements to cybersecurity for buildings utilizing IoT technology. The IDS demonstrated nearly flawless detection capabilities because the Random Forest model produced results that exceeded 99% accuracy. An ensemble learning system proved effective in modeling IoT traffic according to the existing research on this method for identifying traffic anomalies. Our system monitored overt attacks (floodong) along with concealed threats (data exfiltration) by avoiding the need for predefined attack rules. The dynamic feature of AI systems demonstrates superior effectiveness over static signature methods because it enables an adaptive defensive solution that specifically meets the needs of modern IoT environments.

The system produced false alarms at a very minimal rate throughout all experiments. The IDS proved to be highly accurate during its examination of normal traffic over an extended period as it produced next to no false alerts. The detection system minimizes a vital challenge in IDS implementation because it substantially lowers the number of erroneous alarm alerts. The analysis demonstrates smart building environments allow AI-based IDS to achieve highly precise detection when devices operate predictably. The automated detection system becomes more meaningful due to its increased trustworthiness in environments where system disruptions weaken operational stability. Smart building infrastructure becomes effective for real-time threat detection through engineered features alongside optimized model parameters so they operate reliably without typical management complexities of manual rules or overwhelming alerts.

The blockchain implementation served as an effective mechanism to both establish trustworthy devices and protect secure communications. All unregistered or revoked devices automatically received network denial during the authentication process. The benefits of

blockchain technology for securing IoT become apparent through its decentralized authentication model which defeats any attempts of impersonation. The spoofing tests demonstrated how attackers malfunctioned because their imitation devices failed cryptographic security checks. The blockchain's impervious logging mechanism established a permanent audit section which served as an essential compliance tool and forensic detective tool. The blockchain network validates "Device X revoked at time Y" events because they survive as immutable on-chain records which surpass the susceptibility of typical logs to modification or loss.

The main benefit of this solution came from its rapid defense abilities. Through its integrated system threats were isolated within seconds which proved faster than any human-controlled process. The normal sequence where attackers move at a faster pace than security defenders becomes inverted. Real-time threat containment became possible through automation enabled by a highly accurate IDS that conducted revocations. The high level of automation model precision (99%) supported automated security procedures which prevented unnecessary device cuts. When exceptions occurred due to incorrect analysis from the smart contract administrators possessed an immediate method to restore devices through a built-in function. The integration of automation with high speed and accuracy establishes a robust defensive system that raises security standards for smart building attackers.

Novelty and Contributions of the Solution

The study develops an original combination of AI intrusion detection and blockchain access control to secure smart building IoT security which remains a new hybrid solution without existing empirical evidence. This research creates a new automated framework which demonstrates a unique combination of these independent technologies.

A vital point exists in the components' functioning because blockchain checks trusted devices for access while IDS keeps the blockchain informed about compromised devices to perform automatic trust removal. The two-layer method overcomes limitations that each technology stands on its own since it detects the dual threat of insider violations and external impersonators through trusted devices becoming malicious.

Our test demonstrated real-time network removal of thermostats and cameras automatically through automated mitigation in an entirely physical IoT setup. Self-determining environments can use this concept as an example for their development. The system seamlessly integrates with current infrastructure through examples such as gateway queries of blockchain states and IDS alert responses so it can be deployed right away.

On the empirical side, we evaluated multiple ML models and confirmed that Random Forest provided the best performance for IoT traffic, aligning with academic literature. (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers). We prioritized interpretability, extracting intuitive decision rules from the model (e.g., connection and port thresholds), helping IT teams understand and trust the system's decisions. These interpretable rules also offer practical value—e.g., for supplementing firewalls with heuristics derived from the AI's logic.

Strengths and Advantages

The integrated AI-blockchain security framework offers several distinct advantages for smart building cybersecurity:

- **Comprehensive Threat Coverage:**
  The system protects against a full spectrum of threats—external intrusions, insider attacks, and rogue device introductions. AI-based monitoring captures both known and novel anomalies, while blockchain enforces device identity, blocking impersonators and unauthorized access. This layered defense significantly raises the attack difficulty: adversaries must both bypass cryptographic authentication and avoid detection by behavioral analysis.
- **Autonomous Real-Time Response:**
  A core strength is the system's ability to respond automatically and immediately. Threats are detected and mitigated in seconds, without human intervention, limiting damage and freeing up security personnel for strategic tasks. In safety-critical environments, such automation can prevent physical harm—e.g., isolating a device before it disables alarms or safety systems.
- **Resilience and Transparency via Blockchain:**
  Decentralized authentication removes reliance on a single point of failure and ensures changes—like device revocation—are transparent and tamper-resistant. Consensus-based updates increase system trustworthiness, allowing stakeholders to verify actions and reducing the risk of arbitrary or malicious security decisions.
- **Scalable and Simplified Management:**
  Operational scalability is another advantage. Device onboarding is automated via smart contracts, and no manual configuration is required for IDS coverage or access control. A single blockchain transaction updates trust status across the network, making it easier to manage large deployments while minimizing human error.
- **Alignment with Industry Trends:**
  The solution anticipates the shift toward Zero Trust architectures and AI-driven security analytics. Its design enforces continuous device validation and leverages machine learning to handle complex traffic patterns. This forward-looking approach aligns with emerging best practices and supports future compliance requirements—such as mandatory anomaly detection in critical IoT systems.

Limitations and Areas for Improvement

Despite its strengths, the proposed framework has several limitations that warrant consideration and future enhancement:

1. Scope of Detection:
The current system focuses on network-layer and device authentication anomalies. It does not address physical attacks (e.g., someone physically tampering with a device) or purely data-layer manipulation unless such activity causes abnormal traffic. For instance, a compromised device subtly transmitting incorrect data at normal intervals might evade detection. To mitigate this, a logical next step would be integrating sensor data anomaly

detection—evaluating whether transmitted values make physical sense (e.g., a temperature reading of 80°C in an air-conditioned office). Combined analysis of data semantics and traffic behavior could improve overall threat coverage.

2. Blockchain Operational Considerations:
While the private blockchain is lightweight and effective, it introduces operational overhead. Key management is a critical concern—compromise of the admin key or authority node could allow unauthorized device revocations or registrations. This necessitates hardened key management practices (e.g., hardware security modules or multi-signature control). Additionally, blockchain's immutability can make human errors harder to reverse—a mistakenly revoked device must be re-added via a new transaction. Operational safeguards and clear procedures are needed to manage such edge cases.

3. Latency Constraints in Time-Critical Systems:
Although a 1–2 second detection-to-response window is acceptable for most building operations, some real-time systems may find even brief delays problematic (e.g., safety-critical control loops). This latency primarily stems from the blockchain consensus process. For deployments with stricter real-time requirements, tuning blockchain parameters (e.g., faster consensus mechanisms) or using a hybrid architecture (e.g., pairing blockchain with in-memory trust ledgers for instant decisions) could reduce response time while retaining core benefits.

4. Ongoing Model Maintenance:
Machine learning models require regular updates to remain effective. Changes in device usage patterns, building configurations, or added IoT endpoints may cause model drift. To preserve accuracy, the IDS should be retrained periodically with fresh, validated data. While blockchain can help ensure only clean devices contribute training data—reducing poisoning risks—maintaining this pipeline adds complexity and requires dedicated oversight from security teams. The system is not "set-and-forget"; it demands lifecycle management to stay resilient.

5. Integration with Legacy Protocols:
Many building automation systems use non-IP protocols such as BACnet or Modbus. Since the current solution assumes IP-based IoT traffic, retrofitting legacy environments requires intermediary gateways or protocol translators. This is feasible, but represents additional engineering work. Likewise, integrating blockchain identities with existing organizational asset registries or identity systems (e.g., enterprise PKI) would enhance manageability, but also introduces compatibility challenges.

6. Potential for Adversarial Evasion:
No IDS is entirely immune to evasion. A skilled attacker could potentially learn the model's behavioral thresholds and craft traffic patterns that stay below them. While Random Forest's multi-feature evaluation makes this difficult, it's not impossible. Future enhancements could include ensemble modeling with diverse detectors (e.g., combining supervised and unsupervised models), adaptive thresholding, and periodic model updates based on concept drift. Defense-in-depth strategies, including traditional signature-based detection for known threats, can further strengthen resilience.

Impact on Legal, Ethical, and Industry Practices

Legal and Regulatory Compliance:

Our AI-blockchain solution enables compliance with leading cybersecurity standards ISO/IEC 27001 and IEC 62443 through its continuous monitoring and automatic access regulation and its non-modifiable logging system. The logs enable auditing activities to prove that security controls were functional throughout their designated periods. Our team expected privacy-related regulatory challenges when dealing with personal Internet of Things devices present in workplace settings. We controlled traffic analysis by strictly monitoring infrastructure-owned devices which analyzed metadata and device behaviors instead of person-specific content. The design combines security features to protect sensitive assets including CCTV and visitor logs through restrictions that stop both unauthorized access and compromise.

Ethical Considerations:

This system delivers more secure environments while providing better safety measures in smart buildings thus constituting a positive ethical outcome. The automated disconnection process carries safety risks when incorrect devices are identified as safety-critical. Our IDS requires modified settings which we suggest should be applied to safety-critical devices such as fire alarms through alert-only operation and human examination before disengagement. Our security framework puts a strong focus on explainable decision-making because administrators need to understand how the IDS functions through interpretive models joined by blockchain records which demonstrate transparency. The public should be informed about monitored systems in their buildings through transparent disclosure practices so trust remains uncompromised even in sensitive environment.

Cultural and Operational Impact: Facilities management security culture evolves towards IT expertise and preventative detection when facilities management adopts advanced technology such as AI and blockchain. The management of buildings used to exist independently from IT security operations. Convergence brings value to organizations because it requires close cooperation between facilities maintenance and IT/security groups to execute new HVAC system integration which entails blockchain registration and IDS surveillance configuration. To achieve this training will need to take place between facilities staff who learn cybersecurity basics and IT staff who learn about building systems. The long-term result of such cross-pollination will create better integrated smart building management which eliminates segregation between teams. The implementation of our system will demonstrate its usefulness to encourage other organizations to use AI and blockchain in security measures thus advancing industrial security standards.

Industry Adoption and Standards: Our method has the potential to shape standards in the smart building design industry if adopted by a wide range of users. Manufacturers would begin installing device features that enable these frameworks through components such as blockchain identity support and security telemetry generation for IDS integration. Every device at its manufacturing stage would receive one unique cryptographic identity which our blockchainngoing system uses. Standard organizations need to create blockchain standards for device IDs and specify anomaly detection feature collections. The research moves forward the architecture vision which embeds security through automated design principles in smart buildings.

The implementation of our system leads to better capabilities for reacting to incidents and retrieving from damaging events. Tracking down the faulty piece of equipment in a conventional building structure proves to be very challenging because fallout caused by system malfunctions like HVAC controller failures extends across the entire building. Our system detects problems at particular devices which it logs for quicker maintenance thus enabling operations teams to switch out or correct those devices before blockchain restoration. Building managers can experience better downtime reduction and focused maintenance procedures as two additional benefits.

The discussion confirms that the integrated framework both achieves technical effectiveness while conforming to modern cybersecurity requirements. AI together with blockchain strengths become the foundation of a practical forward-thinking solution that addresses their risks effectively. The core concept undergoes validation yet additional refinement steps need attention particularly regarding extended features and strong future operation execution. Such a security system has proven potential for real smart building deployment with appropriate management and oversight to enhance its security position substantially. Such protection measures fulfill both IT security requirements and the basic security needs of overall smart building design through their simultaneous defense of physical systems and human safety.

Conclusion and Recommendations

Summary of Key Findings

The discussion confirms that the integrated framework both achieves technical effectiveness while conforming to modern cybersecurity requirements. AI together with blockchain strengths become the foundation of a practical forward-thinking solution that addresses their risks effectively. The core concept undergoes validation yet additional refinement steps need attention particularly regarding extended features and strong future operation execution. Such a security system has proven potential for real smart building deployment with appropriate management and oversight to enhance its security position substantially. Such protection measures fulfill both IT security requirements and the basic security needs of overall smart building design through their simultaneous defense of physical systems and human safety.

- A **machine learning-based IDS** (particularly using ensemble methods like Random Forest) can accurately and rapidly detect a wide range of cyber-attacks on IoT devices, achieving ~99% detection accuracy in our experiments with a false positive rate around 1%. It proved effective at identifying both common attacks (DDoS floods, port scans) and more subtle anomalies, validating the use of AI for real-time monitoring of smart building networks.

- **Blockchain technology** provides a robust mechanism for IoT device identity management and access control. By decentralizing authentication, our framework ensured that only legitimate, registered devices could operate in the network, and any device flagged as compromised could be swiftly isolated. The immutable, consensus-driven nature of the blockchain adds trust and transparency to security

operations, something traditional centralized controls lack.

- The integration of AI and blockchain resulted in a **layered defense** that is greater than the sum of its parts. The blockchain stopped impersonation and unauthorized access attempts by design, while the IDS actively monitored and flagged malicious behavior from authorized devices. Together, they closed important security gaps; attacks that might slip past one layer were caught by the other. This dual-layer approach significantly improved security coverage compared to conventional measures.

- The system achieved **real-time automated threat mitigation**. In simulated attack scenarios, it detected and contained threats within 1–2 seconds, dramatically limiting potential damage. This autonomous response capability marks a shift from reactive to proactive security, indicating that smart building systems can be made self-defending to a meaningful extent.

- Importantly, these security enhancements were attained with **minimal disruption** to normal operations. Legitimate IoT device activities were almost never falsely flagged or blocked in testing, and the overhead introduced by security processes (network latency, computation) was low. This means the framework can strengthen security without adversely affecting the functionality or performance of building automation systems.

The project showed how AI united with blockchain creates an effective approach to protect IoT networks. The project provides substantial proof of how innovative technological applications address major IT difficulties through integrated solutions. Research results from this study include both the approved framework and methodological findings regarding model optimization and system structure which businesses can utilize to construct forthcoming solutions.

Contributions of the Research

This work makes several contributions to both research and practice:

- **Practical Security Architecture:** We developed a working prototype of an AI-blockchain security architecture for smart buildings, going beyond theoretical proposals. This serves as a reference model that practitioners can build upon. The architecture delineates how to instrument device onboarding, continuous monitoring, and automated response in a unified system – a blueprint that can be adapted to various IoT domains.

- **Empirical Evidence:** The research provides empirical evidence of the efficacy of combining AI and blockchain for cybersecurity. Through quantitative results, it corroborates that high accuracy detection and fast incident response are achievable. These data points add to the body of knowledge, supporting arguments in favor of such integrated solutions in academic and industry discussions.

- **Integration Techniques:** We addressed engineering challenges of integration (e.g., linking Python ML with Ethereum blockchain, real-time data processing), yielding techniques and code that can be reused. For instance, our method of using a web3 API for automated smart contract calls from an IDS, or the approach to buffer and handle asynchronous alerts, are contributions that others implementing similar systems can utilize.

- **Enhanced Understanding of IoT Threat Defense:** The research also contributed a deeper understanding of how IoT-specific factors (like regular device behavior patterns) can be leveraged for security. We identified which network features are most indicative of IoT attacks and showed that focusing on those can minimize false alarms. These findings can guide future IDS development for IoT – emphasizing context-aware feature engineering and model training.

- **Security Best Practices for Smart Buildings:** By exploring this topic, the research highlights best practices such as unique cryptographic identities for devices, continuous anomaly detection, and decentralized trust management. These concepts might influence how smart building systems are designed in the future. For example, building standards might incorporate recommendations to include an anomaly detection engine or a distributed ledger for device certificates, inspired by work like ours.

- **Open-Source Artifacts:** We have made the non-sensitive parts of our implementation (e.g., the smart contract code, sample data generation scripts) available as open-source (hypothetically, as part of this research outcome). This allows other researchers or engineers to replicate or extend our work, thereby contributing to collective progress in this area.

Recommendations and Future Work

Based on the experience gained and the results of this project, we offer the following recommendations and directions for future work:

For Smart Building Operators and Practitioners:

- **Adopt Layered Security for IoT:** We strongly recommend that organizations managing smart buildings implement a multi-layered security strategy. In practice, this means complementing traditional network security (firewalls, VLANs) with advanced monitoring (AI-based IDS) and robust device identity management (e.g., certificate-based authentication or blockchain). The cost of deploying these has reduced, and as shown, the benefits in threat prevention are substantial.

- **Institute IoT Device Onboarding Procedures:** Every IoT device added to a building should go through a security onboarding – assign it a unique identity, register it in an access control system (like the blockchain ledger or an equivalent), and baseline its normal behavior for the IDS. This will ensure new devices are immediately covered by the security framework and any deviant behavior can be caught early. Processes

and training should be put in place for facilities/IT teams to follow this procedure.

- **Continuous Monitoring and Updates:** AI-based IDS needs continuous monitoring together with periodic software updates after deployment. Every environmental change warrants model retraining together with new attack data input (quarterly checks or major system modifications). Regular system performance checks of the IDS alongside blockchain should occur to allow needed threshold adjustments or resource improvements. The framework should become a central element of the building maintenance schedule.
- **Incident Response Plan Integration:** Add the automatic system as part of organizational incident response protocols. When the IDS revokes a device follow the documented steps that guide technicians through device status verification as well as compromise inspection while ensuring safe restoration. The blockchain logging system enables investigators to track down what events occurred. Organizations should plan for these automated actions to allow human responders to work with the AI system rather than against it.
- **Protect the Protectors:** Secure the security infrastructure itself. Ensure the blockchain nodes and the servers running the IDS are themselves hardened (patched OS, restricted access, backups of blockchain data, secure key management for admin keys). An attacker who manages to disable the IDS or tamper with the blockchain could open the door for other attacks, so these components should be considered critical infrastructure.

Future Work and Research Directions:

- **Scalability and Distributed IDS:** Research investigations should focus on enlarging the framework capabilities for handling large networks and investigating distributed IDS implementation possibilities. A smart campus containing multiple buildings can implement several IDS nodes which exchange information through a consortium blockchain as described in Sarhan et al.'s (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) federated learning and blockchain approach. The system performance under large-scale deployment of hundreds or thousands of devices requires investigation together with optimization studies which could use hierarchical models or enhanced consensus algorithms.

- **Incorporating Device Behavior Analytics:** As noted, integrating analysis of the *content* of IoT data (not just network metadata) could further improve detection, especially for subtle attacks. Future systems might combine our network IDS with an AI that learns typical sensor readings or actuator states (for instance, an HVAC system's normal temperature ranges). This could detect malicious commands that are within normal network patterns but abnormal in semantic effect. Research into multi-modal anomaly detection (network + physical data) is an exciting direction.

- **Enhancing Model Robustness with Adversarial Learning:** To address potential adversarial evasion, future work could use adversarial machine learning techniques to harden the IDS model. For example, generating adversarial samples to test the IDS and then training the model on those to make it more robust could be explored.

Additionally, employing multiple diverse models (ensemble of different algorithm types) might make evasion more difficult. Evaluating the system against skilled red-team attacks would provide insights into any weaknesses that could be fortified.

- **Automated Model Maintenance (AutoML):** Managing the IDS model could be made more autonomous. Techniques from AutoML could be applied so that the system self-tunes its hyperparameters or even its feature set as new data comes in. A future version could have an automated retraining module that runs off-hours, validates the new model against a test set, and swaps it in if performance is improved (with blockchain recording model version changes for audit). This would reduce the human effort in maintaining peak accuracy.

- **Integration with Broader Smart City Infrastructure:** Future research might also examine using a similar framework at a larger scale – for smart city or critical infrastructure networks that include smart buildings as components. The blockchain trust network could be extended city-wide, and anomaly detection could occur at multiple levels (device, building, regional). We foresee that the principles proven here can generalize to any IoT-rich environment.

- **User and Device Privacy Considerations:** The deployment of these tracking systems requires research to find proper measures for maintaining privacy when performing extensive monitoring operations. Raw data remains exclusively on devices using techniques like differential privacy or federated learning to prevent an extremely powerful IDS from turning into an unauthorized surveillance tool. The device behavior-focused design of our system already reduces intrusion yet future development might add privacy-conducting features from the beginning to gain regulatory clearances and broader adoption.

The research has established an effective and practical method to secure modern smart buildings. Stakeholders should adopt these cutting-edge techniques according to the recommendations while fostering further development. Active measures will enable us to maintain strategic positions in advance of adversaries while protecting security and safety from consequences of modern building automation and connectivity.

Ultimately, **the project shows that AI and blockchain technologies, when thoughtfully combined, can transform the security posture of IoT-enabled environments**. Smart buildings can be made resilient, self-defending, and trustworthy, protecting both the digital and physical assets within. Future work will build on this foundation to refine and expand the solution, but the successes achieved herein provide a strong justification for moving towards intelligent, decentralized security frameworks in the era of pervasive computing.

References

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Liu, X., Yang, Y., Choo, K.-K. R., & others. (2018). *Security and Privacy Challenges for Internet-of-Things and Fog Computing*. **Wireless Communications and Mobile Computing**, 2018, Article ID 7158520. https://doi.org/10.1155/2018/7158520

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Mosenia, A., & Jha, N. K. (2017). *A Comprehensive Study of Security of Internet-of-Things*. **IEEE Transactions on Emerging Topics in Computing**, 5(4), 586–602. https://doi.org/10.1109/TETC.2016.2606384

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Puche Rondon, L., Babun, L., Aris, A., et al. (2020). *PoisonIvy: (In)secure Practices of Enterprise IoT Systems in Smart Buildings*. arXiv preprint arXiv:2010.05658. *(Identified common insecure practices like default passwords and exposed ports in enterprise IoT deployments.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Zhao, Z., Liu, Y., Zhang, Y., et al. (2023). *Detection of Vulnerabilities in Smart Buildings Using the Shodan Tool*. **Electronics**, 12(23), 4815. https://doi.org/10.3390/electronics12234815 *(Demonstrated widespread IoT vulnerabilities in smart buildings using Shodan; stressed need for stricter access control.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Zhang, Y., Li, X., Li, Y., et al. (2023). *An Artificial Intelligence Lightweight Blockchain Security Model for Industrial Internet of Things*. **Journal of Cloud Computing**, 12(1), 12. https://doi.org/10.1186/s13677-023-00412-y *(Proposed an AI-integrated lightweight blockchain model to enhance IIoT security and efficiency.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Reshi, I. A., & Sholla, S. (2024). *AI-Protected Blockchain-based IoT Environments*. arXiv preprint arXiv:2405.13847. *(Advocated for AI-powered threat detection in blockchain-enabled IoT environments for adaptive security.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Alkadi, O., Alenezi, M., Almustafa, K., et al. (2024). *Enhancing IoT Network Security: Machine Learning and Blockchain for Intrusion Detection*. **International Journal of Advanced Computer Science and Applications**, 15(4). *(Used ML and blockchain to improve intrusion detection accuracy and scalability in IoT networks.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Sarhan, M., Lo, W. W., Layeghy, S., et al. (2022). *HBFL: A Hierarchical Blockchain-based Federated Learning Framework for Collaborative IoT Intrusion Detection*. arXiv preprint arXiv:2204.04254. *(Proposed privacy-preserving intrusion detection using federated learning and blockchain.)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Manh, B. D., Nguyen, C. H., Hoang, D. T., et al. (2024). *Privacy-Preserving Cyberattack Detection in Blockchain-Based IoT Systems Using AI and Homomorphic Encryption*. arXiv preprint

arXiv:2412.13522. *(Developed secure AI model leveraging encryption and blockchain to detect cyberattacks without exposing data.)*

Roman, R., Najera, P., & Lopez, J. (2011). *Securing the Internet of Things*. **IEEE Computer**, 44(9), 51–58. https://doi.org/10.1109/MC.2011.291 *(Early but influential work on rethinking IoT security beyond traditional models.)*

(Optimized IoT Intrusion Detection using Machine Learning Technique) Mahmud, M. Z., Islam, S., Alve, S. R., & Pial, A. J. (2024). *Optimized IoT Intrusion Detection using Machine Learning Technique*. arXiv preprint arXiv:2412.02845. *(Random Forest classifier had highest accuracy ~99.39%, while KNN had lowest ~94.84%, indicating ensemble methods' superiority in IoT IDS (Optimized IoT Intrusion Detection using Machine Learning Technique).)*

(Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers) Ashraf, J., Raza, G. M., Kim, B.-S., et al. (2025). *Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers*. **Applied Sciences**, 15(4), 2043. https://doi.org/10.3390/app15042043 *(Found Random Forest most robust for IoT IDS with 99.2% accuracy, Naïve Bayes second at 98.8% (Making a Real-Time IoT Network Intrusion-Detection System (INIDS) Using a Realistic BoT–IoT Dataset with Multiple Machine-Learning Classifiers).)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Whitecliffe (2025). *Project IT9115 – Securing Smart Building Sensor Networks Against Cyber Threats – Proposal Draft*. *(Noting older IDS are susceptible to false alarms, causing alert fatigue and need for smarter systems (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx).)*

(IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx) Whitecliffe (2025). *Project IT9115 – Proposal Draft*. *(Our AI-driven IDS aims to handle false positives by learning long-term patterns and adapting to new threats, thereby reducing nuisance alerts (IT9115-Id-20231297 Project Proposal Draft 1 Feedback[1].docx).)*
