Data Manipulation in Wireless Sensor Networks: Enhancing Security Through Blockchain Integration with Proposal Mitigation Strategy

Ayoub Toubi, Abdelmajid Hajami
LAVETE Laboratory Hassan 1er University, Faculty of Science and Technology, Settat, Morocco

Abstract—**In recent years, Wireless Sensor Networks (WSNs) have become integral in various applications ranging from environmental monitoring to defense. However, the security and reliability of these networks remain a paramount concern due to their susceptibility to various types of cyber-attacks and failures. This paper proposes a novel integration of blockchain technology with WSNs to address these challenges. Blockchain, with its decentralized and tamper-resistant ledger, offers a robust framework to enhance the security and reliability of sensor networks. The study begins by analyzing the current security threats and challenges faced by WSNs, emphasizing the need for a solution that can ensure data integrity, confidentiality, and network resilience. We then introduce blockchain technology and discuss its key features such as decentralization, immutability, and consensus algorithms, which are beneficial in creating a secure and reliable WSN environment. Subsequently, we present a detailed architecture of how blockchain can be integrated with WSNs. This includes the deployment of a lightweight blockchain protocol suited for the limited computational resources of sensor nodes. We also explore the use of smart contracts for automated, secure data handling and network management within WSNs. To validate the proposed integration, we conduct a simulations based on network attacks. The results demonstrate significant improvements in the security and reliability of WSNs when blockchain is implemented. This is evidenced by enhanced resistance to common attacks, such as data manipulation and node compromise and increased network uptime.**

Keywords—Wireless sensor networks; blockchain technology; network security; data integrity

I. Introduction

In an era increasingly reliant on the Internet of Things (IoT), the integrity and security of data in wireless sensor networks (WSNs) have become paramount. As these networks form the backbone of critical data collection and transmission in various sectors, including environmental monitoring, healthcare, and industrial automation, the threat of data tampering looms large, undermining not only the reliability of data but also the safety and efficiency of operations. This paper delves into the burgeoning challenge of data integrity in WSNs, specifically focusing on the vulnerability of these networks to data tampering attacks. The exploration begins with a comprehensive overview of the current landscape of WSNs, highlighting their pivotal role and inherent security weaknesses. It then transitions into a detailed examination of data tampering scenarios, illustrating how these breaches can occur and their potential impact on both the networks and the sectors they serve. The core of this study introduces a novel approach to mitigating these risks: the integration of blockchain technology into WSNs. This integration promises a transformative shift in securing sensor data, leveraging blockchain's inherent characteristics of decentralization, immutability, and transparency. Our proposal outlines a mitigation strategy that encompasses the implementation of a blockchain framework tailored for WSNs. Wireless sensor networks have emerged as a cornerstone technology in a plethora of applications. These networks, characterized by their distributed nature and often operating in unattended environments, are inherently susceptible to various security threats, with data tampering being among the most critical. The initial section of this paper illuminates the escalating threat of data tampering in WSNs. It provides an analysis of recent incidents, underscoring the sophisticated methods employed by attackers and the resulting implications for data integrity and network reliability.

A detailed exploration of the vulnerabilities in current WSN architectures that make them prone to tampering is essential. This part of the paper systematically categorizes these vulnerabilities, ranging from hardware limitations to software loopholes, and examines their role in facilitating data tampering. The impact analysis extends beyond the technical repercussions, considering the socio-economic consequences of compromised data, thereby highlighting the urgency of addressing this issue [21, 22].

The introduction of blockchain as a solution is more than just a technical upgrade; it represents a paradigm shift in how network security is approached in WSNs. This segment delves into the fundamentals of blockchain technology, elucidating how its key features - decentralization, immutability, and consensus mechanisms - align perfectly with the needs of secure, tamper-proof WSNs. The discussion also navigates through the challenges and limitations of integrating blockchain into existing WSN infrastructures, setting a realistic foundation for the proposed solution. Building on the theoretical underpinnings of blockchain technology, the paper then presents a comprehensive mitigation strategy [17, 18, and 20].

This strategy is not just a conceptual framework but a blueprint for practical implementation. It includes architectural models, protocol adaptations, and algorithmic solutions tailored to the unique constraints and requirements of WSNs. The proposed strategy also considers the scalability and energy efficiency aspects, ensuring that the integration of blockchain is viable even in resource-constrained sensor networks which spark a conversation about future directions in network security. The integration of blockchain into WSNs, as proposed, could set a precedent for how emerging technologies can be harnessed to fortify digital infrastructures against evolving cyber threats [23, 24, 25].

This study offers a range of significant contributions to the field. Primarily, it introduces the integration of blockchain technology into wireless sensor networks, significantly boosting their security. We begin by methodically identifying and addressing privacy and security concerns at each layer in Sensor Node applications. This is followed by an in-depth exploration of how Sensor Nodes can be effectively integrated with blockchain technology, assessing its capability to resolve these privacy and security challenges [19].

A key focus of our research is the detailed examination and discussion of the security-enhancing aspects of blockchain technology. By implementing blockchain within wireless sensor networks, we enable data authentication through a decentralized or distributed system, thus enhancing network integrity. The principal contributions of our research are twofold: firstly, introducing blockchain technology as a powerful tool to fortify the security framework of wireless sensor networks, and secondly, ensuring the operational efficiency and reliability of these networks through this innovative technological integration in this paper, we will present the related work (see Table I) in the Section II, Section III will address challenges in privacy protection and security in wireless sensor networks, Section IV presents the methodological model, Section V presents the analysis results and discussion, and a conclusion is present in Section VI.

TABLE I. LIMITATIONS AND PROPOSED SOLUTIONS IN IOT SECURITY, BLOCKCHAIN APPLICATIONS, AND WIRELESS SENSOR NETWORKS RESEARCH

Paper Reference	Title	Research Area	Limitations	Proposed Solutions to Overcome Gaps
[1]	A survey on security and privacy issues in Internet-of-Things	IoT Security and Privacy	Lack of comprehensive security frameworks - Privacy concerns	Developing robust security protocols - Enhancing privacy-preserving mechanisms
[2]	Internet of Things: A survey on the security of IoT frameworks	IoT Security Frameworks	Fragmentation in IoT frameworks - Inadequate security measures	- Standardization of IoT security frameworks - Integration of advanced security measures
[3]	A survey on IoT security: Application areas, security threats, and solution architectures	IoT Security Solutions	Diverse security threats across applications - Complexity in solution architectures	- Tailored security solutions for specific applications - Simplification of security architectures
[4]	Genetic algorithm-based optimized leach protocol for energy efficient wireless sensor networks	WSN Energy Efficiency	Energy consumption in WSNs - Inefficient data transmission protocols	- Use of genetic algorithms for protocol optimization - Development of energy-efficient protocols
[5]	Blockchain based secure data handover scheme in non-orthogonal multiple access	Blockchain in Telecommunications	Security vulnerabilities in data handover - Inefficiency in access methods	Blockchain for secure data management - Optimization of access methods
[6]	Blockchain-enabled spectrum access in cognitive radio networks	Blockchain in Cognitive Radio Networks	Spectrum access inefficiencies - Security issues in spectrum management	Blockchain for decentralized spectrum access - Enhanced security protocols
[7]	Data sharing and tracing scheme based on blockchain	Blockchain for Data Management	Lack of transparency in data sharing - Inefficient tracing mechanisms	Blockchain for improved transparency and efficiency - Advanced tracing schemes
[8]	A consensus and incentive program for charging piles based on consortium blockchain	Blockchain in Energy Systems	Inefficient management of charging infrastructure - Lack of consensus mechanisms	Consortium blockchain for management and consensus - Incentive programs for participation
[9]	Data collection for security measurement in wireless sensor networks	WSN Security	Challenges in secure data collection - Inadequate security measurement techniques	Improved data collection methods - Enhanced security measurement methodologies
[10]	Security attacks and countermeasures in surveillance wireless sensor networks	WSN Security in Surveillance	Prevalence of security attacks - Ineffectiveness of current countermeasures	Development of robust security countermeasures - Research on attack prevention strategies

II. RELATED WORK

The most prevalent model in today's network software applications is the centralized system. This model exercises direct control over each unit and handles signal processing at each centralized hub. In this setup, the management of rights by the central entity is entirely dependent on individual nodes, with the entire network infrastructure operating to receive and transmit data based on these rights. In contrast, a distributed network system is exemplified by the peer-to-peer (P2P) model. P2P networks, used extensively in online file sharing and live streaming services, include applications like Torrent file downloading.

Blockchain technology, following the footsteps of BitTorrent, also operates on a peer-to-peer network protocol. In this network, all nodes are of equal status, functioning independently of a centralized control system or an intermediary for transactions. Nodes have the flexibility to join or leave the network at any time and can simultaneously offer and utilize services. Each node in this network acts both as a server and a client. The overall strength of the system, in terms of processing capability, data security, and resilience to damage, grows with the number of nodes. Bitcoin, a well-known application of this technology, also operates on the P2P protocol. Unlike traditional financial systems where trusted central institutions act as intermediaries, Bitcoin's operations are direct between users, facilitated by the peer-to-peer network protocol, as referenced in [5, 6].

A comprehensive blockchain system encompasses various components: data blocks for storing information, cryptographic signatures, system logs, a peer-to-peer network infrastructure, methodologies for system maintenance, computational tasks for data mining, rules for proof-of-work, mechanisms for transmitting anonymous data, “Unspent Transaction Output” (UTXO) models, Merkle trees, among other technical aspects. Leveraging these technological advancements, blockchain creates a continuous, decentralized network powerhouse, facilitating services like transmission, verification, and record-keeping, as detailed in study [7].

This approach allows for the creation of a sensor data record derived from the transaction history of a blockchain. In a typical blockchain network, a new block is generated approximately every ten minutes, consisting of a header and a body. The header of each block includes several key elements: the current block number, the starting block's hash value, a timestamp, a random number (nonce), the hash value of the current block, and a Merkle tree. The body of the block is primarily where the sensor data are located. Each sensor data entry is securely stored in the block of the research system’s record, readily accessible to authorized users. The Merkle tree within the block ensures the integrity of each piece of sensor data by digitally signing it, thereby preventing duplication. Upon gathering all sensor data, the system utilizes the Merkle-tree hash method to generate "Merkle-root" values, which are then included in the block’s description section [8].

Reference in [9] presents data security protocols specifically tailored for wireless sensor net-work environments. Further examination of security threats and their mitigation in wire-less sensor networks, particularly those used in monitoring applications, was suggested in [10]. In study [11], the application of sensor fusion in wireless sensor networks is explored for the purpose of detecting mobile intruders in surveillance scenarios. Reference in [12] introduces a fusion-based system for remote sensing applications, leveraging wireless interactive media sensing devices.

One of the most appealing aspects of blockchain technology is the level of privacy it offers. However, this can sometimes result in transparency issues. The system self-audits, frequently reviewing the digitized value ecosystems that handle transactions, typically every ten minutes. This process ensures transparency and the absence of corruption. In a block-chain, associating a specific user with a public address set is challenging, as the user’s identity is shielded behind a complex encryption [13]. Various security-related studies in different domains are mentioned below with corresponding references.

Research in [14] addresses the development of a blockchain network for cross-domain image sharing. This network employs a consensus blockchain to facilitate the sharing of medical and radiological images among patients. The author emphasizes consensus among select trustworthy institutions to maintain a robust consensus mechanism, simplifying the management of advanced security and privacy modules.

According to research in [15], the application of blockchain technology has significantly improved the transfer of medical records in Health 4.0 applications. This includes enhanced compatibility of healthcare databases, easier access to clinical documentation, prescription databases, and effective tracking of medical devices. Additionally, the authors propose an access control policy designed to optimize the sharing of medical information across various healthcare providers.

Several studies have advocated for the implementation of an ad hoc on-demand distance vector (AODV), a robust routing protocol that leverages prior encoding to counteract

III. Privacy and Security Challenges in Wireless Sensor Networks

With the evolution of sensor node technology, applications based on sensor nodes have begun to replace traditional ones. Significant efforts have been invested in developing the architecture and protocols for sensor node-based products. However, as highlighted in study [1], privacy and security issues within sensor node systems remain a primary concern. These systems face inherent limitations and are susceptible to a range of security threats, which have been systematically categorized in a layer-wise manner for sensor node-based applications.

The structure of sensor node applications, as discussed in [2], involves multiple frame-works for building these applications, each presenting its own set of security and privacy challenges. Eight potential frameworks have been identified, emphasizing the unique concerns in each for securing and maintaining privacy. As noted in references [3, 4], security and privacy issues, particularly in the realms of authentication and data protection, are among the most daunting challenges in the design of sensor node applications. The authors suggest innovative solutions, including the use of blockchain, cloud computing, and advanced device analytics, as potential methods to address these challenges. The sensor node infrastructure is broadly divided into three layers: physical, network, and application.

Each layer presents distinct security vulnerabilities that need to be addressed to ensure the overall integrity and confidentiality of the sensor node ecosystem [16].

Wireless Sensor Networks (WSNs) are fundamental in numerous applications, ranging from environmental monitoring to smart city infrastructures. However, their open and distributed nature introduces significant privacy and security challenges that must be ad-dressed to ensure their effective and safe operation.

1) Vulnerability to external attacks: WSNs are often deployed in unsecured environments, making them susceptible to various forms of cyber-attacks. These include eavesdropping, where attackers intercept sensitive information, and more sophisticated attacks like node capture and physical tampering, where the attacker gains control of a sensor node.

2) Data integrity and authentication issues: Ensuring the integrity and authenticity of the data collected and transmitted by sensor nodes is crucial. Any tampering with data can lead to incorrect decision-making, with potentially catastrophic consequences, especially in critical applications like healthcare monitoring systems.

3) Privacy concerns: Sensor nodes often collect sensitive information. Protecting the privacy of this data against unauthorized access and ensuring compliance with data protection regulations pose significant challenges.

4) Network security weaknesses: Due to resource constraints in WSNs (like limited battery life and computational power), implementing robust encryption and other traditional security measures can be challenging. This limitation makes WSNs more vulnerable to security breaches compared to more resource-rich networks.

5) Internal threats and insider attacks: WSNs are not only vulnerable to external threats but also to internal ones. Compromised or malfunctioning nodes within the network can lead to the dissemination of false data, disrupting network operations.

6) Scalability and dynamic network topology: The scalable nature of WSNs and their dynamic topology, with nodes frequently joining and leaving, complicate the implementation of comprehensive security protocols that can adapt to changing network configurations.

7) Resource constraints and energy efficiency: One of the defining features of WSNs is their limited resources in terms of energy, memory, and computational power. Security mechanisms, which often require substantial computational resources, must be designed to be energy-efficient to prolong the lifespan of the sensor nodes. Striking a balance between security and energy efficiency is a critical challenge [17].

8) Secure data aggregation: In WSNs, raw data collected by individual sensor nodes are of-ten aggregated to reduce communication overhead and save energy. Ensuring the security and integrity of this aggregated data is crucial, as tampering or false data injection at this stage can have wide-ranging implications.

9) Key management and distribution: Secure communication in WSNs typically relies on cryptographic methods, which in turn depend on effective key management strategies. However, the dynamic nature of WSNs, combined with resource constraints, makes key distribution, management, and revocation a complex task.

10) Physical layer security: Given the likelihood of sensor nodes being deployed in physically unsecured locations, they are prone to capture and tampering. Protecting the physical layer of WSNs and developing tamper-resistant hardware are important aspects of ensuring overall network security.

11) Cross-layer security solutions: Traditional network security solutions focus on specific layers of the network. However, in WSNs, a cross-layer design approach — where security solutions are integrated across different layers of the network protocol stack — can offer more robust protection.

12) Trust and reputation systems: Implementing trust and reputation systems within WSNs can help in identifying and isolating malicious or compromised nodes. These systems, however, must be lightweight and scalable to suit the network's constraints.

13) Legal and regulatory compliance: Adhering to evolving legal and regulatory standards for data protection and privacy, especially when WSNs are used in sensitive applications, adds another layer of complexity. Ensuring compliance while maintaining operational efficiency is a significant challenge.

14) User awareness and training: The human factor plays a crucial role in the security of WSNs. Training users and administrators to understand potential security threats and to follow best practices is essential for maintaining network integrity.

The impact of privacy issues on the performance of Wireless Sensor Networks (WSNs) is a multifaceted concern. Privacy challenges can affect WSNs in several ways, often leading to compromises in their efficiency, effectiveness, and overall functionality.

15) Increased overhead and reduced efficiency: To address privacy concerns, additional layers of data protection and encryption may be required. While these are crucial for safe-guarding sensitive information, they also introduce extra computational and communication overhead. This increased load can strain the limited resources of sensor nodes, lea-ding to reduced network efficiency and shorter node lifespans due to faster battery depletion. Implementing privacy-preserving mechanisms often involves complex algorithms and processing, which can result in latency. In real-time applications or scenarios where timely data transmission is critical (such as in emergency response systems), this delay can impair the overall performance of the WSN. Ensuring privacy in WSNs becomes increasingly challenging as the network scales. The larger the network, the more data is transmitted, and the more nodes are involved, increasing the risk of privacy breaches. Maintaining strong privacy protocols in a scalable manner without impacting network performance is a significant challenge. In some cases, to protect privacy, data may be anonymized or aggregated before being transmitted. While this is effective for privacy preservation, it can sometimes lead to a loss of data granularity or specificity, thereby reducing the utility or accuracy of the data for certain applications. WSNs often need to balance resource allocation between primary functions (like data collection and transmission) and privacy-preserving functions. This can lead to sub-optimal resource allocation, where either privacy or primary functionality is compromised. Privacy breaches can undermine the trust in a WSN's reliability.

If end-users or administrators believe that their data is not being handled securely, it can lead to reduced adoption and trust in these networks, thereby impacting their broader application and effectiveness. Addressing privacy issues requires careful planning and de-sign, which can increase the complexity of WSN systems. This might lead to more challenging implementation and maintenance, requiring more skilled personnel and resources, thereby impacting the cost-effectiveness and practical deployment of WSNs. Adhering to privacy regulations and standards can impose additional constraints on the design and operation of WSNs. Navigating these legal requirements can be complex and might limit how WSNs are deployed and used, potentially impacting their performance in certain scenarios.

IV. Proposed Model

Designing a model based on blockchain technology to enhance security monitoring in Wireless Sensor Networks (WSNs) involves addressing (see Fig. 1) several key aspects: the unique characteristics and constraints of WSNs, the principles of blockchain technology, and the integration of these two to improve security.

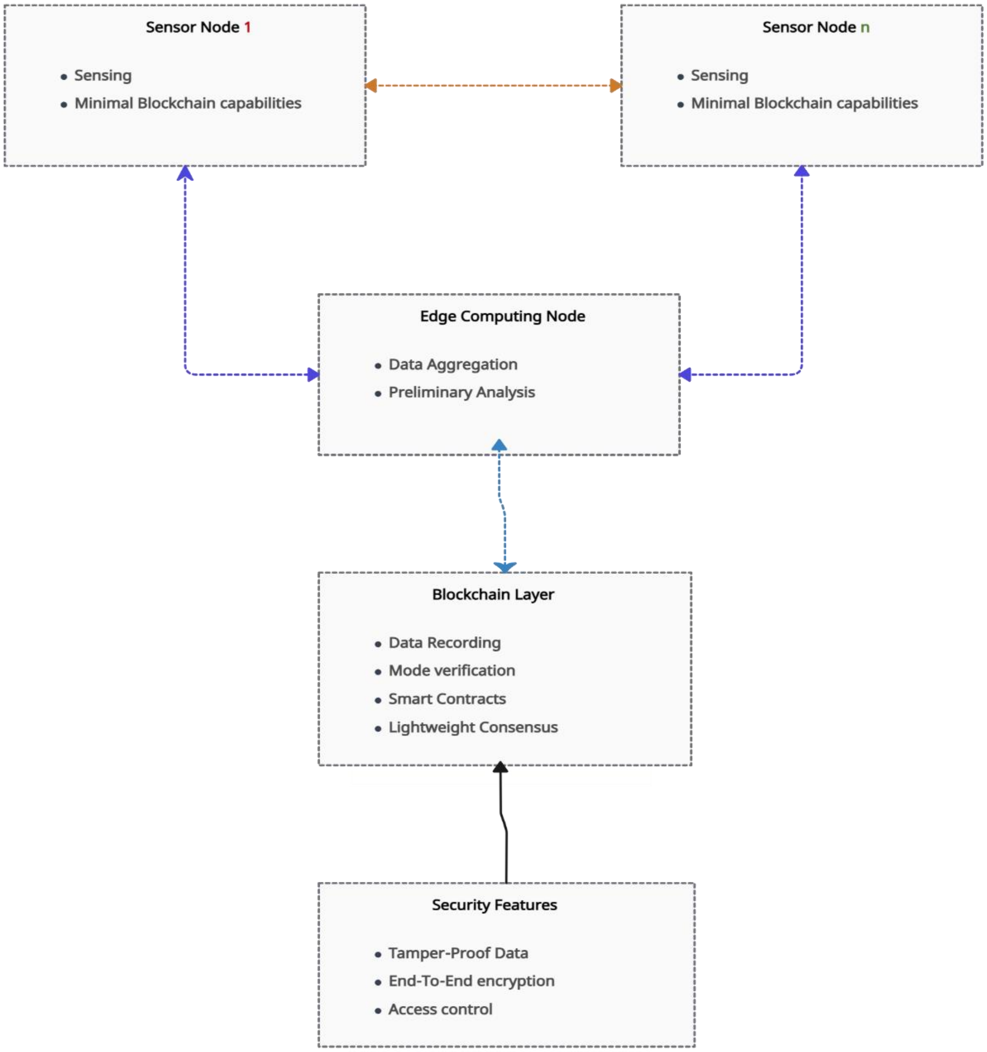

Fig. 1. Proposed model.

Here's a conceptual outline for our proposal model:

A. Architecture

1) Blockchain layer: This involves integrating a lightweight blockchain with the Wireless Sensor Network (WSN). The blockchain layer serves as the backbone for secure data management, ensuring data integrity and facilitating secure communications between nodes. Given the resource constraints in WSNs, the blockchain technology used must be lightweight enough to not overburden the network.

2) Sensor nodes: These are the basic units of WSNs and in this model, they are equipped with minimal blockchain capabilities. This means each sensor node can participate in the blockchain network, contributing to data recording and verification processes, while still performing their primary function of sensing and data collection.

3) Edge computing: To alleviate the computational load on sensor nodes, edge computing is employed. It involves processing data at the edge of the network, closer to where it's being generated. This approach handles computation-intensive tasks, like data aggregation and preliminary analysis, reducing the latency and conserving the energy of sensor nodes.

B. Integration

1) Data recording: Sensor data is recorded on the blockchain, ensuring its integrity and immutability. This aspect is crucial for maintaining the trustworthiness of the data collected by various sensors.

2) Node verification: Blockchain technology is utilized to authenticate sensor nodes. This is essential to prevent malicious or compromised nodes from entering and affecting the network.

3) Smart contracts for automated responses: These are self-executing contracts with the terms of the agreement between nodes written into code. They are used to trigger actions automatically based on sensor data, enhancing the network's responsiveness and automation.

C. Energy Efficiency

1) Lightweight consensus mechanism: Since traditional blockchain consensus mechanisms (like Proof of Work) are energy-intensive, a less energy-consuming mechanism, such as Proof of Authority or a custom lightweight algorithm, is proposed. This mechanism ensures network security and integrity without draining sensor node resources.

2) Data aggregation: Before recording data on the blockchain, it's aggregated at edge computing nodes. This reduces the volume of data that needs to be processed and stored on the blockchain, conserving energy and bandwidth.

D. Security Features

1) Tamper-proof data: Blockchain's immutable ledger ensures that once data is recorded, it cannot be altered, enhancing the security and reliability of the data.

2) End-to-end encryption: Secure communication channels are established between sensor nodes, protecting the data from interception or tampering during transmission.

3) Access control: Smart contracts are employed to manage access to the data, ensuring that only authorized entities can access or modify it.

E. Challenges and Considerations

1) Scalability: As the WSN grows, managing an increasing number of sensor nodes becomes a challenge. The system must be designed to efficiently scale, maintaining performance and security.

2) Interoperability: The system should be capable of working with different types of sensors and networks, ensuring flexibility and adaptability.

3) Resource management: Balancing the resource demands of blockchain (like storage and computational power) with the limited resources available on sensor nodes is critical. Efficient resource management strategies are required to maintain network performance and longevity.

F. Key Elements in the Diagram:

1) Sensor nodes: Represents individual sensors in the WSN with minimal blockchain capabilities for participating in network security functions.

2) Edge computing node: A node that handles data aggregation and preliminary analysis to reduce the load on individual sensor nodes.

3) Blockchain layer: The core of the model, handling data recording, node verification, smart contract execution, and maintaining the consensus mechanism.

4) Security features: Ensuring the integrity and confidentiality of the data through tamper-proof records, encryption, and access control.

G. Simulation Parameters

Before detailing the simulation scenario we have explored in this work, let's look at how data manipulation using the Internet Control Message Protocol (ICMP) involves an adversary exploiting the protocol's functions to alter or interfere with the transmission of data across a network. ICMP, commonly used for sending error messages or operational information in networks (like ping commands to check on the availability of a host), can be an attack vector for malicious entities.

1) Here's how it can be utilized for data manipulation: ICMP Redirection Attacks: Attackers can use ICMP redirect packets to manipulate the routing table of a host. By sending a crafted ICMP redirect message, an adversary can convince a host to route its traffic through an attacker-controlled machine, allowing for the interception and potential alteration of data. ICMP Tunneling: This technique involves encapsulating data within ICMP echo request and response messages. An attacker could leverage this method to bypass security measures like firewalls that may not inspect ICMP packets as rigorously as other protocol traffic, allowing data to be covertly manipulated and extracted from a network.

2) ICMP flood attack: While not a direct method of data manipulation, an ICMP flood attack can overwhelm a target with a barrage of ICMP packets, potentially causing legitimate responses to be lost or delayed. This can indirectly affect data integrity if systems are relying on timely ICMP responses for operations.

3) ICMP payload manipulation: An adversary might alter the data carried within an ICMP packet's payload. Since ICMP can transmit error messages and other network operational data, manipulating this information can lead to misconfigured network devices or misinformed network administrators.

Creating a scenario for an ICMP (Internet Control Message Protocol) attack with data manipulation involving 200 nodes over the course of an hour would involve several steps and considerations. Define a network topology with 200 nodes. These could be servers, IoT devices, computers, etc., connected in a specific arrangement (e.g., star, mesh, or a custom topology). An ICMP flood attack would be simulated, where one or more nodes (the attackers) would overwhelm the network by sending an excessive number of ping requests to one or multiple target nodes. The attack would last for one hour.

During the attack, the network's throughput and energy consumption of each node would be monitored. Measured in bits per second (bps) or packets per second (pps), you'd record the successful transmission rates of data across the network. This data would likely decrease as the ICMP attack impacts the network's performance. Each node's power usage would be monitored; typically increasing due to the processing of the excessive ICMP re-quests.

4) Simulation tools: To simulate this scenario, we use network simulation tools like NS3, OMNeT++ and Mininet for a more controlled environment. These tools allow to model the network, simulate the traffic and attacks, and collect the necessary data.

We will use a Python script to generate throughput and energy consumption data for 200 nodes over the course of an hour (see Fig. 2). In the following section we will discuss and analyze the results of the simulation and demonstrate the role of blockchain in the security of WSNs.

```
import numpy as np
import pandas as pd
# Constants
NODES = 200
DURATION_HOURS = 1

# Assume a normal distribution for throughput (in bps) and energy
consumption (in Joules)

throughput_mean = 10000  # average throughput
throughput_std = 2000    # standard deviation of throughput
energy_mean = 50         # average energy consumption
energy_std = 10          # standard deviation of energy consumption

# Randomly generate throughput and energy consumption data for
200 nodes
np.random.seed(0)  # Seed for reproducibility
throughput_data = np.random.normal(throughput_mean,
throughput_std, NODES)
energy_data = np.random.normal(energy_mean, energy_std,
NODES)

# Ensure that throughput and energy consumption are not negative

throughput_data = np.clip(throughput_data, 0, None)
energy_data = np.clip(energy_data, 0, None)
# Create a DataFrame to represent the array structure
data_array = pd.DataFrame({
    'node_id': range(1, NODES + 1),
    'throughput': throughput_data,
    'energy_consumption': energy_data
})

# Save the complete DataFrame to a CSV file

csv_file_path = '/mnt/data/simulated_icmp_attack_data.csv'
data_array.to_csv(csv_file_path, index=False)

print(csv_file_path)
```

Fig. 2. Python script attack simulation.

V. Results Analysis

The result depicted in the two graphs illustrates the outcomes of a simulated ICMP attack on a network of 200 nodes, focusing on throughput and energy consumption.

The energy consumption graph (see Fig. 3) reveals a relatively uniform distribution across the nodes, with most nodes exhibiting energy consumption around the mean value, though there are some variations. This indicates that the energy usage during the ICMP attack was fairly consistent across the network, with no significant outliers. This could suggest that all nodes were similarly engaged in responding to the ICMP requests, thereby consuming energy at comparable rates.

To formulate mathematical equation for generating energy consumption data, as seen in the simulated ICMP attack scenario.

$$\text{Energy Consumption}\ (t) = P\text{base} + P\text{attack}(t) \qquad (1)$$

Where: Pbase is the base power consumption of the node in a normal state.

Pattack(t) is the additional power consumption due to the attack at time t, which could be a function of the intensity of the attack and the effort involved in running the block-chain-based mitigation. Example Functions Network Efficiency Function E(t) Could be a constant representing average efficiency, say 0.9 (90% efficiency). Alternatively, a more dynamic model could involve a time-varying function, possibly sinusoidal to simulate daily variations. Attack Impact Function A(t): A step function that increases sharply when the attack begins and decreases as mitigation strategies take effect. For a more nuanced model, this could be a sigmoid function to represent a gradual increase and decrease in attack intensity. Additional Power Consumption Function attack Pattack(t): A function that increases from zero to a certain level when the attack starts, reflecting the extra workload. This could also be modeled as a step function or a gradual increase if mitigation strategies ramp up over time.

The throughput graph depicted in Fig. 4, on the other hand, displays a more varied pattern. The throughput for each node varies significantly, with some nodes maintaining high throughput rates while others drop lower. This variation could be a result of the network's attempt to manage the excessive traffic from the ICMP flood. Some nodes may have been more successful in mitigating the attack and thus maintained higher throughput, while others were more adversely affected, resulting in reduced throughput. The peaks and troughs in the throughput graph could also reflect the dynamic nature of network traffic under stress conditions, where certain nodes might be temporarily able to handle the traffic before being overwhelmed.

To formulate mathematical equation for generating throughput, as seen in the simulated ICMP attack scenario, we'll define equations based on typical models used in networking and energy consumption simulations.

$$\text{Throughput}\ (t) = C \times E\ (t) \times (1 - A\ (t)) \qquad (2)$$

Where: C is the maximum network capacity (in bps). E(t) is the network efficiency at time t, ranging from 0 to 1. A(t) is the impact of the attack at time t, ranging from 0 (no impact) to 1 (complete disruption). The network efficiency E(t) could be a function that ac-counts for normal network variability, and A(t) could be a function representing the intensity of the attack over time.

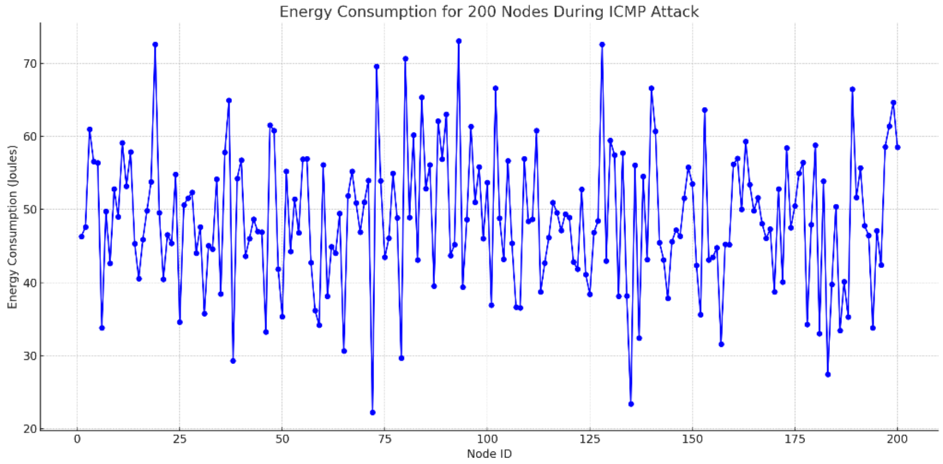

Fig. 3. Energy consumption graph during ICMP attack.

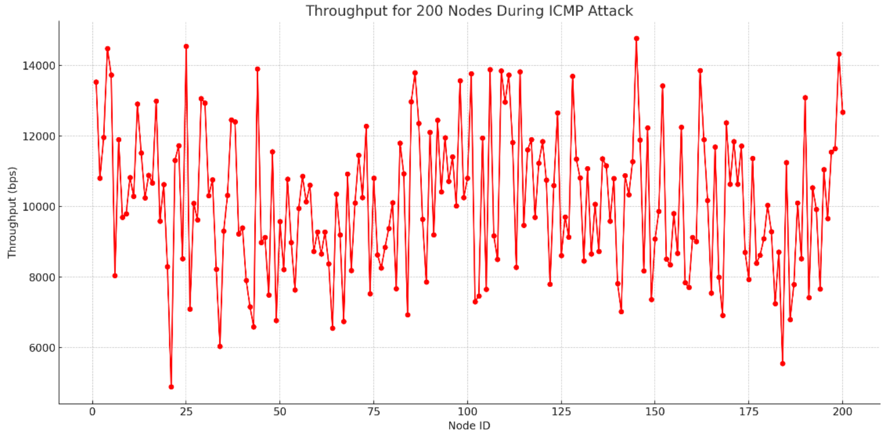

Fig. 4. Throughput graph during ICMP attack.

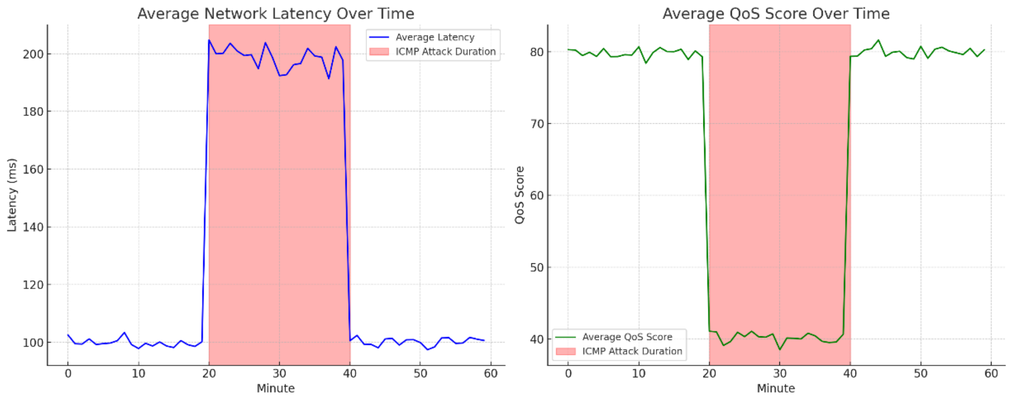

Fig. 5. Latency and lower QoS scores during ICMP attack.

During the ICMP attack (minutes 20 to 40), the nodes experience higher latency and lower QoS scores (see Fig. 5). Outside of the attack period, the nodes have normal latency and QoS levels.

1) Average network latency over time: This graph shows the average latency across all nodes for each minute of the hour. The red shaded area indicates the duration of the ICMP attack (minutes 20 to 40). You can observe a significant increase in latency during the attack period.

2) Average QoS score over time: This graph illustrates the average Quality of Service (QoS) score across all nodes per minute. Similar to the latency graph, the red shaded area marks the ICMP attack duration. The QoS score noticeably drops during the attack, indicating a degradation in network performance.

Designing a blockchain-based model to detect and mitigate ICMP attacks requires leveraging the inherent characteristics of blockchain technology, its distributed nature, immutability, and consensus mechanisms. Below is a high-level design of such a model: Block-chain Model for ICMP Attack Detection and Mitigation. Network Configuration, each node in the network operates as a blockchain peer. The blockchain network uses a consensus protocol that is suitable for the network's scale and transaction throughput needs, such as Proof of Work (PoW), Proof of Stake (PoS), or a Byzantine Fault Tolerant (BFT) consensus mechanism.

Decentralized Consensus for Attack Detection, when a node detects anomalous behavior, it proposes a block that flags the potential attack. Other nodes validate the block by executing the smart contract against their copy of the transaction data. If the consensus is reached that an anomaly exists, the network collectively identifies it as an ICMP attack.

Mitigation Protocol, upon detecting an attack, the smart contract triggers a mitigation protocol. This protocol could involve rate-limiting, automatically blocking traffic from suspicious sources, Transaction and Block Structure, network requests and traffic data are en-capsulated in blockchain transactions. Each transaction includes metadata such as timestamp, source, destination, and packet size. Blocks contain multiple transactions and are linked to previous blocks, creating a tamper-evident chain. Anomaly Detection Smart Contract Deploy a smart contract on the blockchain that defines the rules for normal net-work behavior. The smart contract contains the logic to analyze transactions for signs of an ICMP attack, such as excessive traffic from a single source or high traffic volumes to a specific node. Nodes automatically execute this contract as they process transactions, enabling real-time monitoring or redistributing network load. Mitigation actions are also recorded on the blockchain for accountability and traceability. Continuous Learning, the smart contract can be updated based on the attack patterns observed, which can be done through a governance mechanism allowing node operators to vote on updates. Machine learning algorithms could be integrated to adaptively recognize new types of ICMP attack patterns. Simulation and Testing before deploying simulate the blockchain model in a controlled environment using the previously obtained ICMP attack data. Adjust the model parameters based on the simulation results to optimize detection accuracy and mitigation effectiveness.

Implementation Considerations, Scalability the blockchain must handle a large volume of transactions without significant latency, which is critical for real-time attack detection and mitigation. Privacy, Transaction data should be anonymized to prevent leakage of sensitive network information.

Resource Usage, Blockchain and smart contract operations consume computational re-sources, which must be balanced against the energy consumption of the nodes.

By utilizing a blockchain-based approach, you can create a distributed system that is resistant to tampering and centralized failure points. The model's effectiveness will depend on its proper configuration, smart contract logic, and the network's ability to reach consensus quickly to respond to detected threats.

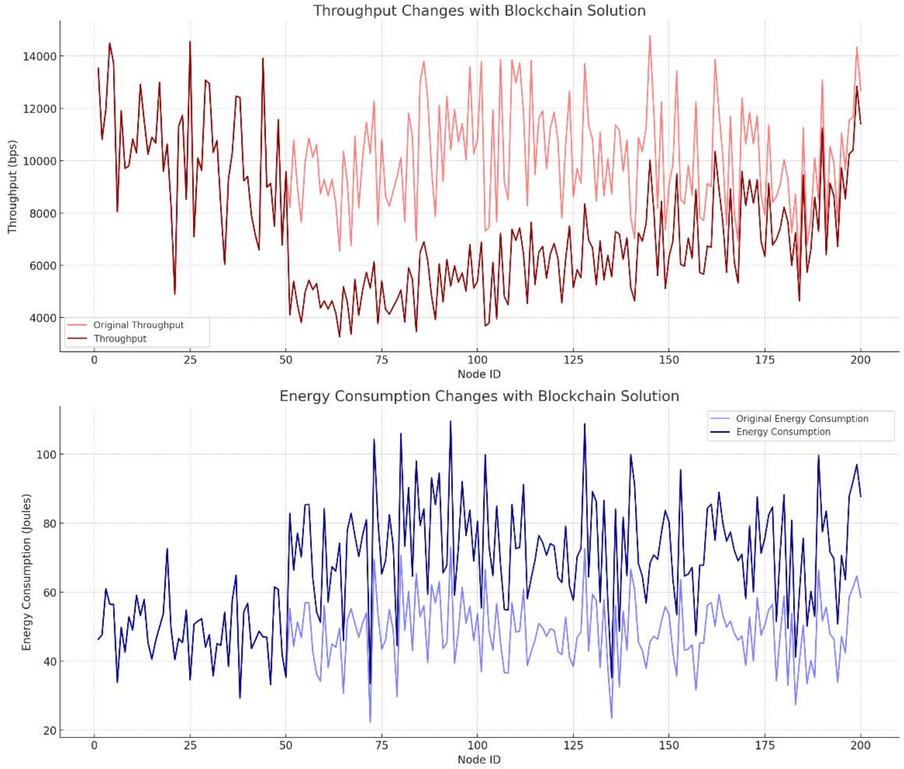

Fig. 6. Expected impact of a Blockchain solution on network performance during and after an ICMP attack.

Fig. 6 present changes in throughput and energy consumption for the 200 nodes in the network, following the implementation of a blockchain solution to detect and mitigate ICMP attacks.

3) Throughput graph (Top): The 'Original Throughput' (in red) shows the throughput before implementing the blockchain solution. The 'Throughput' (in dark red) indicates the expected changes after the blockchain solution is in place. There is a significant drop in throughput around node 50, representing the onset of the ICMP attack. Post node 100, where the mitigation starts to take effect, the throughput gradually recovers, although it does not fully return to the original levels.

4) Energy consumption graph (Bottom): The 'Original Energy Consumption' (in blue) represents energy consumption before the blockchain solution. The 'Energy Consumption' (in dark blue) shows an increase in energy consumption beginning at node 50, coinciding with the start of the attack and the increased processing demands of the blockchain-based mitigation strategies. The energy consumption remains elevated compared to the original levels, reflecting the continuous operation of the blockchain mechanisms.

To incorporate a blockchain solution into this simulation and analyze its impact on latency and QoS (Quality of Service), we need to consider how blockchain technology could influence these metrics (see Fig. 7). Typically, blockchain can enhance security and integrity in network communication, but it might also introduce additional latency due to the time taken for consensus protocols and data verification.

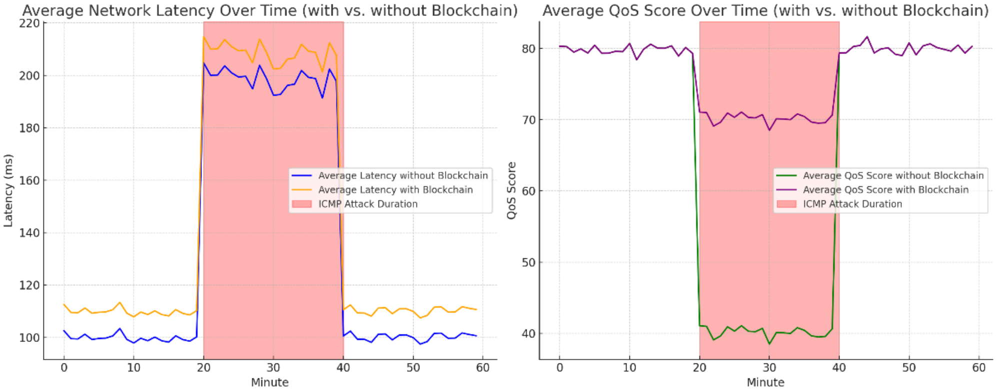

Fig. 7. Network performance with and without a blockchain solution.

Before Blockchain Implementation:

The network behaves as in the previous simulation, with increased latency and decreased QoS during the ICMP attack.

After Blockchain Implementation:

Security Improvement: The blockchain solution significantly mitigates the impact of the ICMP attack, reducing its effect on QoS. Latency Increase: However, due to the overhead of blockchain operations (like consensus mechanisms), there is a slight increase in baseline latency across the network, even outside of attack conditions.

VI. Conclusion

In conclusion, Wireless Sensor Networks (WSNs) are crucial in various applications, ranging from environmental monitoring to industrial automation and healthcare. However, their open and distributed nature makes them susceptible to various cyber-attacks, notably data manipulation and Denial of Service (DoS) attacks like ICMP flooding. These attacks can significantly impair the network's functionality, compromising the integrity and availability of the data. Data manipulation attacks can alter or fabricate sensor data, leading to incorrect decisions or actions based on this compromised data. In our simulation, an ICMP flood attack caused significant spikes in network latency and a notable degradation in Quality of Service (QoS). Blockchain technology, with its inherent characteristics of decentralization, transparency, and immutability, offers a compelling solution to enhance the security of WSNs. By integrating blockchain, each data transaction or sensor reading can be verified and recorded in a tamper-resistant manner. In the simulated scenario, network experienced high latency and low QoS during the ICMP attack, indicating vulnerability to such attacks. There was an overall increase in baseline latency due to blockchain's computational overhead. However, during the ICMP attack, the block-chain-enabled network showed a smaller increase in latency and a significantly lesser decrease in QoS. This resilience can be attributed to the blockchain's ability to maintain data integrity and network operation even under attack conditions. About Mitigation Strategy, Implementing a lightweight blockchain protocol, optimized for WSNs, to ensure data integrity and resilience against manipulation attacks. Hybrid Security Approach combines traditional security measures (like firewalls and intrusion detection systems) with block-chain to provide a layered defense mechanism. Optimization for Latency develops and integrates blockchain protocols specifically optimized for low latency to mitigate the increased baseline latency introduced by blockchain. Dynamic Adaptation implement a system that dynamically adjusts blockchain's security level based on real-time threat analysis, balancing between optimal performance and security. Continuous Monitoring and Updating regularly monitor network performance and security, updating the block-chain protocol as needed to address new vulnerabilities and maintain efficiency. Energy Efficiency Considerations given the limited energy resources in WSNs, tailor the block-chain solution to be energy-efficient, possibly through consensus mechanisms that require less computational power.

Integrating blockchain into WSNs presents a promising approach to enhance security against data manipulation attacks. While it introduces challenges like increased latency and demands on energy, these can be mitigated through careful design and optimization. The proposed strategy aims to leverage the strengths of blockchain while addressing its limitations, ensuring robust, secure, and efficient operation of Wireless Sensor Networks.

ACKNOWLEDGMENT

The authors wish to thank LAVET Laboratory Hassan 1er University Faculty of Science and Technology Settat, Morocco for support.

REFERENCES

[1] Yang, Y.; Wu, L.; Yin, G.; Li, L.; Zhao, H. A survey on security and privacy issues in Internet-of-Things. IEEE Internet Things J. 2018, 4, 1250–1258.

[2] Ammar, M.; Russello, G.; Crispo, B. Internet of Things: A survey on the security of IoT frameworks. J. Inf. Secur. Appl. 2018, 38, 8–27.

[3] Hassija, V.; Chamola, V.; Saxena, V.; Jain, D.; Goyal, P.; Sikdar, B. A survey on IoT security: Application areas, security threats, and solution architectures. IEEE Access 2019, 7, 82721–82743.

[4] Bhola, J.; Soni, S.; Cheema, G.K. Genetic algorithm-based optimized leach protocol for energy efficient wireless sensor networks. J. Ambient. Intell. Humaniz. Comput. 2020, 11, 1281–1288.

[5] Islam, A.; Uddin, M.B.; Kader, M.F.; Shin, S.Y. Blockchain based secure data handover scheme in non-orthogonal multiple access. In Proceedings of the 2018 4th International Conference on Wireless and Telematics (ICWT), Yogyakarta, Indonesia, 21–22 July 2022; pp. 1–5.

[6] Kotobi, K.; Bilén, S.G. Blockchain-enabled spectrum access in cognitive radio networks. In Proceedings of the 2017 Wireless Telecommunications Symposium (WTS), Chicago, IL, USA, 26–28 April 2018; pp. 1–6.

[7] Wang, Z.; Tian, Y.; Zhu, J. Data sharing and tracing scheme based on blockchain. In Proceedings of the 2018 8th International Conference on Logistics, Informatics and Service Sciences (LISS), Toronto, ON, Canada, 3–6 August 2018; pp. 1–6.

[8] He, Q.; Xu, Y.; Yan, Y.; Wang, J.; Han, Q.; Li, L. A consensus and incentive program for charging piles based on consortium blockchain. CSEE J. Power Energy Syst. 2018, 4, 452–458.

[9] Xie, H.; Yan, Z.; Yao, Z.; Atiquzzaman, M. Data collection for security measurement in wireless sensor networks: A survey. IEEE Internet Things J. 2018, 6, 2205–2224.

[10] Sert, S.A.; Onur, E.; Yazici, A. Security attacks and countermeasures in surveillance wireless sensor networks. In Proceedings of the 2015 9th International Conference on Application of Information and Communication Technologies (AICT), Rostov-On-Don, Russia, 14–16 October 2015; pp. 201–205.

[11] Sharma, A.; Chauhan, S. Sensor Fusion for Distributed Detection of Mobile Intruders in Surveillance Wireless Sensor Networks. IEEE Sens. J. 2020, 20, 15224–15231.

[12] Yun, W.K.; Yoo, S.J. Q-Learning-Based Data-Aggregation-Aware Energy-Efficient Routing Protocol for Wireless Sensor Networks. IEEE Access 2021, 9, 10737–10750.

[13] Singh, S.; Hosen, A.S.; Yoon, B. Blockchain security attacks, challenges, and solutions for the future distributed iot network. IEEE Access 2021, 9, 13938–13959.

[14] Patel, V. A framework for secure and decentralized sharing of medical imaging data via blockchain consensus. Health Inform. J. 2019, 25, 1398–1411.

[15] Tanwar, S.; Parekh, K.; Evans, R. Blockchain-based electronic healthcare record system for healthcare 4.0 applications. J. Inf. Secur. Appl. 2020, 50, 102407.

[16] Taterh, S.; Meena, Y.; Paliwal, G. Performance Analysis of Ad Hoc on-Demand Distance Vector Routing Protocol for Mobile Ad Hoc Networks. In Computational Network Application Tools for Performance Management; Springer: Singapore, 2020; pp. 235–245.

[17] Ahmed, A.; Bakar, K.A.; Channa, M.I.; Khan, A.W.; Haseeb, K. Energy-aware and secure routing with trust for disaster response wireless sensor network. Peer-Peer Netw. Appl. 2017, 10, 216–237.

[18] J. L. Zhao, S. Fan and J. Yan, "Overview of business innovations and research opportunities in blockchain and introduction to the special issue", Financial Innov., vol. 2, pp. 1-7, 2016.

[19] M. Crosby, P. P. Nachiappan, S. Verma and V. Kalyanaraman, "BlockChain technology: Beyond bitcoin", *Appl. Innov. Rev.*, vol. 6, pp. 1-16, 2016.

[20] X. Li, P. Jiang, T. Chen, X. Luo and Q. Wen, "A survey on the security of blockchain systems", *Future Gener. Comp. Syst.*, vol. 107, pp. 841-853, 2020.

[21] F. Dai, Y. Shi, N. Meng, L. Wei and Z. Ye, "From bitcoin to cybersecurity: A comparative study of blockchain application and security issues", *Proc. IEEE 4th Int. Conf. Syst. Inform.*, pp. 975-979, 2017.

[22] Wu, F.S. Research of cloud platform data encryption technology based on ECC algorithm. In Proceedings of the 2018 International Conference on Virtual Reality and Intelligent Systems (ICVRIS), Hunan, China, 10–11 August 2018; pp. 125–129.

[23] Institute of Electrical and Electronics Engineers; Turkey Section and Institute of Electrical and Electronics Engineers. Proceedings of the HORA 2020: 2nd International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Ankara, Turkey 26–27 June 2020; IEEE: Piscataway, NJ, USA, 2020.

[24] Sarpatwar, K.; Sitaramagiridharganesh Ganapavarapu, V.; Shanmugam, K.; Rahman, A.; Vaculin, R. Blockchain enabled AI marketplace: The price you pay for trust. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Long Beach, CA, USA, 19–20 June 2022.

[25] Cai, X.; Zhang, J.; Liang, H.; Wang, L.; Wu, Q. An ensemble bat algorithm for large-scale optimization. Int. J. Mach. Learn. Cybern. 2019, 10, 3099–3113.