

DESIGN AND DEVELOPMENT OF SECURE ROUTING TECHNIQUES FOR IOT BASED NETWORKS

**A Thesis Submitted
in Partial Fulfillment of the Requirements for the
Degree of**

DOCTOR OF PHILOSOPHY

**in
Computer Science & Engineering**

**by
Vishal Sharma
(2K20/PHDCO/506)**

Under the Supervision of

**Dr. Rohit Beniwal
(Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University

**Prof. Vinod Kumar
(Co-Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University



Department of Computer Science & Engineering

**DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daultpur, Main Bawana Road, Delhi-110042, India**

June, 2025

DESIGN AND DEVELOPMENT OF SECURE ROUTING TECHNIQUES FOR IOT BASED NETWORKS

**A Thesis Submitted
in Partial Fulfillment of the Requirements for the
Degree of**

DOCTOR OF PHILOSOPHY

**in
Computer Science & Engineering**

**by
Vishal Sharma
(2K20/PHDCO/506)**

Under the Supervision of

**Dr. Rohit Beniwal
(Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University

**Prof. Vinod Kumar
(Co-Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University



Department of Computer Science & Engineering

**DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India**

June, 2025

ACKNOWLEDGEMENT

I express my profound gratitude to Almighty God for providing me the strength, resilience, and guidance to pursue and complete this research journey. I am deeply indebted to my supervisors, Dr. Rohit Beniwal and Prof. Vinod Kumar, for their invaluable mentorship, constant encouragement, and insightful suggestions throughout this journey. Dr. Beniwal's technical expertise and thoughtful guidance have been instrumental in overcoming the challenges faced during my research. Prof. Vinod Kumar has been a constant source of motivation and support. His leadership and vision inspired me to strive for excellence. Also, my sincere thanks to Prof. Manoj Kumar, HOD, (Dept. of Computer Science and Engineering) for insightful comments and valuable suggestions. I extend my heartfelt thanks to the esteemed faculty members of the Department of Computer Science and Engineering for their unwavering support and encouragement. Their advice and collaborative spirit have enriched my academic experience and contributed significantly to my personal and professional growth.

I would also like to acknowledge the continuous support and encouragement provided by Prof. Prateek Sharma, Vice-Chancellor. His dedication to fostering a research-oriented environment has been a significant driving force behind my accomplishments.

Finally, with a heart full of love and longing, I offer my deepest gratitude to supreme Supervisor my Daddy, Shri Shiv Kumar Sharma, and my Mummy, Mrs. Asha Sharma. Though they are no longer with me, their blessings, sacrifices, and unconditional love continue to shape every part of who I am. Their values and teachings remain my guiding light, and it is their memory that gives me strength and purpose on this journey. This acknowledgment is a humble testament to the collective efforts and support of all these individuals, whose contributions have been pivotal to the successful completion of my doctoral research.

Vishal Sharma

Roll No. 2K20/PHDCO/506



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE DECLARATION

I, Vishal Sharma (2K20/PHDCO/506), Research Scholar in the Department of Computer Science & Engineering, hereby declare that the work which is being presented in the thesis entitled “Design and Development of Secure Routing Techniques for IoT based Networks” in partial fulfillment of the requirements for the award of the Degree of Doctor of Philosophy, submitted in the Department of Computer Science & Engineering, Delhi Technological University is an authentic record of my own work carried out during the period from January, 2021 to June, 2025 under the supervision of Dr. Rohit Beniwal (Supervisor) and Prof. Vinod Kumar (Co-Supervisor) of Department of Computer Science and Engineering, Delhi Technological University, Delhi, India. The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Vishal Sharma

Place: New Delhi

Date: 30-6-2025



DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daultapur, Main Bawana Road, Delhi-42

SUPERVISOR(s) CERTIFICATE

This is to certify that the work embodied in this thesis entitled “**Design and Development of Secure Routing Techniques for IoT based Networks**” done by Vishal Sharma, roll no. 2K20/PHDCO/506 in the Department of Computer Science & Engineering, Delhi Technological University is an authentic work carried out by him under our guidance.

This work is based on original research and the matter embodied in this thesis has not been submitted earlier for the award of any degree or diploma to the best of our knowledge and belief.

Dr. Rohit Beniwal

Assistant Professor
Department of Computer
Science & Engineering
Delhi Technological University
Delhi, India

Prof. Vinod Kumar

Professor
Department of Computer
Science & Engineering
Delhi Technological University
Delhi, India

Date: 30-6-2025

LIST OF TABLES

Table No.	Name of the Table	Page No.
Table 2.1	Metaheuristics classification	22
Table 2.2	Security services	24
Table 3.1	Relation between coati behavior and coatinet module	46
Table 3.2	Parameter settings	55
Table 3.3	Scalability evaluation	61
Table 3.4	Method comparison	62
Table 3.5	Performance evaluation of attack detection	63
Table 4.1	Parameter settings	82
Table 4.2	Comparison table	89
Table 5.1	Parameter setup	111
Table 6.1	Aligning of research objectives and publications	125

LIST OF FIGURES

Figure no.	Name of Figure	Page No.
Figure 1.1	IoT standards for restricted systems and security mechanism	5
Figure 1.2	IoT routing security challenges	7
Figure 2.1	Search selection steps	12
Figure 2.2	Metrics contribution	25
Figure 3.1	Healthcare IoT network system model	31
Figure 3.2	Proposed Multi-cluster security framework	33
Figure 3.3	Network lifetime vs percentage of malicious	57
Figure 3.4	Network throughput vs percentage of malicious	58
Figure 3.5	Detection rate vs percentage of malicious	58
Figure 3.6	False positive rate vs percentage of malicious	59
Figure 3.7	Performance comparison analysis	60
Figure 3.8	Packets processed vs window size	60
Figure 3.9	Ablation results	67
Figure 4.1	Architecture for the Proposed BONY-ISHO Based Secure Clustering and Energy Efficient Routing Technique	71
Figure 4.2	Blowfish algorithm	75
Figure 4.3	Function module	75
Figure 4.4	Flowchart for BONY encryption	77
Figure 4.5	Energy consumption	83
Figure 4.6	End-to-end delay	84
Figure 4.7	Throughput analysis	85
Figure 4.8	Packet delivery ratio analysis	86
Figure 4.9	Alive sensor nodes analysis	87
Figure 4.10	Encryption time analysis	87
Figure 4.11	Decryption time analysis	88
Figure 5.1	The architecture of the proposed ML-HSOR methodology	95
Figure 5.2	Multi-level hierarchical trust evaluation	98
Figure 5.3	Interactions between the different nodes in the trust evaluation	99
Figure 5.4	Illustration of time window	99
Figure 5.5	Energy consumption analysis	115
Figure 5.6	End-to end delay analysis	117
Figure 5.7	Throughput analysis	118
Figure 5.8	PDR analysis	119
Figure 5.9	Network lifetime analysis	120
Figure 5.10	Detection rate analysis	121

LIST OF ABBREVIATIONS

IoT	Internet of Things
WSN	Wireless Sensor Networks
BLE	Bluetooth Low Energy
LORAWAN	Long Range Wide Area Network
6LoWPAN	IPv6 over Low Power Personal Area Network
ESP	Encapsulating Security Payload
RPL	Routing Protocol for Low Power and Lossy Networks
IPSec	IP security
AH	Authentication Header
IKE	Internet Key Exchange
IETF	Internet Engineering Task Force
MANET	Mobile Ad Hoc Networks
BONY	Blowfish-Honey
ISHO	Improved Spotted Hyena Optimization
RB BFT X	Redundant Byzantine Fault Tolerance with Extensions
CoatiNet	Coati-based network
ML-HSOR	Multi-level Hierarchical Secure and Optimal Routing
CH	Cluster Head
DODAG	Destination oriented directed acyclic graph
DIS	DODAG Information Solicitation
UDP	User datagram protocol
CNN	Convolutional Neural Network
UAV	Unmanned aerial vehicle
PLA	Physical layer authentication
BEAST	Behavior-based Enhanced Trust management System
DDoS	Distributed Denial of Service
DoS	Denial of Service
SL	Subjective Logic
ECHSA	Efficient cluster head selection algorithm
CTES	Context-based trust evaluation system
DHT	Distributed Hash Tables
GA	Genetic Algorithm
SMPC	Secure Multi-Party Computation
SHA	Secure Hash Algorithm
MAC	Message Authentication Codes

PSK	Pre-Shared Keys
IBC	Identity-based Cryptography
TLS	Transport Layer Security
DTLS	Datagram Transport Layer Security
MHS	Main Hospital Servers
MITM	Man-in-the-Middle
RBAC	Role-Based Access Control
MQTT	Message Queuing Telemetry Transport
CoAP	Constrained Application Protocol
HTTP	Hypertext Transfer Protocol
GQI	Generalized Quasi-identifiers
GSA	Generalized Sensitive attributes
ACM	Access Control Matrix
FBA	Federated Byzantine Agreement
AES	Advanced Encryption Standard
RUO	Resource Utilization Optimization
IDE	integrated development environment
FL	Federated learning
FPR	False Positive Rate
SC	Smart Contracts
BFA	Brute Force Attacks
RSA	Rivest–Shamir–Adleman
GFLOPS	Giga Floating Point Operations per Second
IBFA	Improved Blowfish Algorithm
PL-COA	Polarity Learning-based Chimp Optimization Algorithm
SN	Sensor Nodes
MN	Member Nodes
QoS	Quality of Service
PDR	Packet Delivery Ratio

TABLE OF CONTENTS

TOPIC	PAGE NO.
Title Page	i
Acknowledgement	ii
Candidate Declaration	iii
Certificate By Supervisor (s)	iv
Abstract	v
List of Publications	vi
Table of Contents	vii
List of Tables	xi
List of Figures	xii
List of Abbreviations	xiii
 Chapter 1: Introduction	 1-11
1.1 The Internet of Things (IoT)	1
1.2 IoT Based Networks	2
1.3 Major Challenges	3
1.4 IoT Routing Security	4
1.5 Motivation	9
1.6 Methods Overview	9
1.7 Outline of the Thesis	9
1.8 Chapter Summary	11
 Chapter 2: Literature Review	 12-27
2.1 Overview	12
2.2 Security for IoT based Networks	13
2.2.1 Routing defense in IoT	14
2.2.2 Metaheuristics for secure IoT Routing	19
2.3 Research Gaps	26
2.4 Research Objectives	26

2.5 Chapter Summary	27
Chapter 3: A Multi-Cluster Security Framework for IoT Based Networks	28-68
3.1 Introduction	29
3.2 System Model	29
3.2.1 Attacker Model	30
3.2.2 Design Goals	32
3.3 Proposed Secured Healthcare IoT Framework	33
3.3.1 Initialization	34
3.3.2 Autonomous cluster formation	38
3.3.3 Two-phase Approach	39
3.4 Experimental Methodology and Performance Analysis	54
3.4.1 Experimental Setup	54
3.4.2 Evaluation Metrics	55
3.4.3 Experimental Analysis	57
3.4.4 Performance Evaluation of Attack Detection	63
3.4.5 Ablation Study	64
3.5 Chapter Summary	67
Chapter 4: A Blockchain Based BONY-ISHO Protocol for Secure Clustering and Routing in IoT based Networks	69-90
4.1 Introduction	69
4.2 Proposed Blockchain based BONY-ISHO Protocol	71
4.2.1 Initialization	72
4.2.2 Clustering	72
4.2.3 Data Aggregation and BONY Encryption	74
4.2.4 Optimal Route Selection	77
4.3 Implementation and Results	81
4.3.1 Experimental Setup	81
4.3.2 Comparative Analysis	82
4.3.3 Computational Complexity	88
4.4 Chapter Summary	89

Chapter 5: Multi-Level Trust based Secure and Optimal IoT-WSN Routing for Environmental Monitoring Applications	91-123
5.1 Introduction	91
5.2 Proposed ML-HSOR Methodology	93
5.2.1 Registration	95
5.2.2 Clustering	96
5.2.3 Authentication by multi-level hierarchical trust evaluation	97
5.2.4 Optimal route selection	104
5.3 Simulation Results and Analysis	111
5.3.1 Simulation Setup	111
5.3.2 Evaluation Metrics	111
5.3.3 Comparative Analysis	113
5.4 Chapter Summary	122
 Chapter 6: Conclusion, Future Scope and Social Impact	 124-128
6.1 Research Summary	124
6.2 Limitations of the Work	127
6.3 Social Impact and Future Scope	127
 References	 129-145

ABSTRACT

The rapid proliferation of Internet of Things (IoT) networks necessitates the development of secure and efficient routing protocols to address challenges such as energy constraints, heterogeneous device communication, and security vulnerabilities. This thesis focuses on the design and development of secure routing techniques for IoT-based networks by leveraging advanced cryptographic methods, hierarchical clustering strategies, and metaheuristic optimization algorithms. Protocols, Blowfish-Honey Improved Spotted Hyena Optimization (BONY-ISHO) and Multi-level Hierarchical Secure and Optimal Routing (ML-HSOR), are introduced to address the dual challenges of security and routing efficiency in IoT networks. BONY-ISHO combines blockchain authentication, hybrid Blowfish-Honey cryptography, and the Improved Spotted Hyena Optimization algorithm to ensure secure clustering, robust data encryption, and energy-efficient routing. Similarly, ML-HSOR employs a Markov model for adaptive clustering, multi-level trust evaluation for malicious node detection, and the Polarity Learning-based Chimp Optimization Algorithm (PL-COA) for optimal data routing. These approaches significantly enhance key performance metrics, achieving high packet delivery ratios, increased throughput, reduced energy consumption, and low latency. These metaheuristics are systematically evaluated for their potential to address routing and security challenges in IoT networks. The protocols developed demonstrate superior scalability, adaptability, and resilience against common network threats like eavesdropping and grey-hole attacks. The findings of this research provide a comprehensive framework for secure and energy-efficient IoT routing, paving the way for resilient smart city applications and beyond.

CHAPTER 1

INTRODUCTION

In recent times, the world has witnessed a profound transformation brought about by the rapid advancement of modern technologies. The World is becoming intelligent with the advent of modern technologies. These technological innovations have empowered individuals to achieve feats once considered unimaginable. Today uncountable quantities of sensors and physical objects are linked together from various areas to advance our lives. This chapter covers an introduction to the Internet of Things (IoT) and IoT based networks. Also, the significance of IoT is discussed in detail along with IoT applications. Subsections explain IoT routing security, attacks, and motivation behind IoT routing security. Outline of the thesis is presented briefly in the end with a summary of the entire chapter.

1.1 The Internet of Things (IoT)

Among the technological marvels driving this transformation, the IoT has emerged as a pivotal contributor. Individuals can do many unthinkable things to which IoT provides many contributions. IoT is based on the ubiquitous existence of items that may interact with each other and work together to achieve common goals. IoT, with its vast network of interconnected sensors, has permeated various facets of our lives, ushering in a new era of convenience and efficiency. Fundamentally, IoT networks link several devices and sensors to the Internet so they may talk to one other and the cloud-based systems that handle and process their data. The IoT is an expanding network of devices that are connected to gather and share data in real-time. This massive network of smart, connected devices that are able to collaborate and communicate with one another in order sense different physical events happening across the globe inevitably turns us back into IoT. These devices gather, display and work on the data via multiple servers and repositories. The IoT, advances the use of intelligent devices in healthcare, environmental applications providing real-time data monitoring for a number medical health conditions and allowing solutions to many day-to-day challenges that arise. Standards that work together are needed to meet the demands of IoT devices for effective, secure, and efficient communication [1] [2].

In simple words IoT is just network of devices which are able to communicate and exchange data with each other over the internet. These devices, which are often equipped so with sensors and software and other technology features, collect a huge amount of data to cover various services such as home automation to industrial monitoring functions. The group of networks connected with the main network, which enable communication through a send or receive from IoT devices is known as "IoT

Network" These networks play an important role in making the different applications of IoT possible across several fields ranging from industrial automation to smart homes and cities. Following are key components of IoT:

- **Devices/Sensors:** Physical objects with sensors and actuators that collect data and perform actions, such as smart thermostats, health monitors, and industrial machines.
- **Connectivity:** IoT devices utilize protocols like Wi-Fi, Bluetooth, Zigbee, and cellular networks to connect and share data.
- **Data Processing:** Data from IoT devices is processed either locally (edge computing) or on centralized servers (cloud computing) to generate insights.
- **User Interface:** Applications and dashboards enable users to interact with IoT systems, control devices, and visualize data.

1.2 IoT Based Networks

The growth of the Internet of Things makes substantial use of mobile computing and wireless communications to monitor, control, and process large amounts of data about the physical world [3]. IoT-based networks are systems made up of physical devices that are connected and able to communicate with each other through the internet or other communication channels. These devices are equipped with sensors, software, and various technologies that allow them to gather information, share it, and respond to it automatically. By using IoT-based networks, farmers gain access to real-time data regarding soil moisture, weather conditions, and crop health, thus enabling precision irrigation and enhancing crop yields. Similarly, in the healthcare sector, IoT-based networks offer continuous monitoring of patients' vital signs, transmitting crucial data to healthcare providers for remote patient care, and timely medical interventions. In the medical sector, IoT-WSN integration is a game-changer, with wearable devices collecting patient health data, transmitting it to healthcare providers, and enhancing remote patient care and early disease detection. In addition, the power industry uses IoT-based networks for grid management, making monitoring and improving energy distribution smarter. With the help of IoT and WSN based networks can trigger energy efficient applications and would make the use of power resources very efficient.

The collaboration between IoT and Wireless Sensor Networks (WSN) has brought about transformative advancements in various sectors. Wireless sensors lie at the core of this technological revolution. Within this expansive landscape, WSN stand as an integral component of IoT systems. Wireless Sensor Networks use sophisticated, intelligent sensors to boost data collection as the IoT grows in and around us. These small, battery-powered physical objects and devices can sense in and around surroundings and can collect data for analysis. WSNs are an essential component in the expansion of the IoT, and their data-gathering capabilities are provided by low-cost smart sensors. Usually, in an IoT system, sensors send their data directly to the internet, while in WSNs, sensors transmit to some cluster head or node. An IoT system can connect to a WSN's router to fetch the environmental data. These physical objects/device sensors are placed in suitable locations to monitor physical conditions and produce large amounts of real-time data, which they send to the central device or

the cloud. These devices can be used in large groups to identify important occurrences or provide periodic environmental information. Thus, it makes them appealing for a variety of applications that require remote monitoring capabilities.

IoT devices provide services like healthcare, smart transportation, smart homes, etc. in which WSNs are an integral part of IoT. Such IoT-based networks' key feature encompasses their ability to coordinate with large networks, allowing them to sense and detect events around them and regularly transmit environmental information. IoT-based networks have become essential in today's fast-developing, innovative environment. Such smart infrastructure supports ubiquitous applications like real-time data collection, traffic management, and environmental monitoring. IoT applications are being developed for a wide range of fields, like the transportation sectors, smart cities, the energy and medical sectors, military, and agricultural monitoring, forest monitoring, etc.

IoT networks connect multiple devices and sensors to the Internet, enabling them to send and receive data. Routing, which refers to the process of selecting paths in a network along which data is transmitted, plays a crucial role in directing this data through various network paths to reach cloud-based systems, where it is processed, stored, or used to trigger specific actions. Efficient routing ensures that information travels securely and reliably between devices and the cloud, even across complex and dynamic network environments. To manage this data exchange, protocols—defined as standardized sets of rules that determine how data is formatted, transmitted, and received—are essential. In IoT systems, multiple protocols must operate together to ensure communication is effective, secure, and timely, especially considering the constraints of low-power devices and heterogeneous network conditions. This collaboration among protocols is key to achieving network efficiency, which refers to the optimal use of network resources to deliver data with minimal delay, low energy consumption, and high reliability. Ultimately, the efficiency and stability of data transmission among physical objects in the Internet of Things depend heavily on the performance of routing protocols and their ability to support seamless, scalable communication.

1.3 Major Challenges

However, this enormous networking growth opens a window to many new paradigms and brings significant challenges. The usage of efficient energy consumption and proper security contributes to the functioning of IoT networks and influences the life of the network [4] [5]. The complexity of addressing requirements and operate in diverse application domains [6]. Availability, authenticity, and confidentiality are distinct qualities for network protection against attacks under fundamental security criteria [7]. Unauthorized users can access sensitive data when transmitted across IoT networks that are insufficiently secured, which is a significant issue. IoT security is challenging because it is susceptible to data integrity compromises caused by data packet alteration and nonrepudiation caused by delivering erroneous network information.

Another significant IoT challenge is preserving user and device privacy [8]. The IoT sensor devices engendered increased network privacy requirements, which further led to secure routing on a heterogeneous network, suggesting the realization of sensing and communication systems by explaining how things integrate into an interoperable network with information processing objectives. Even while IoT networks have authentication methods and cryptographic safeguards in place to protect users' privacy, problems affecting communication include heterogeneity, battery life limitations, and device resource limitations [9]. The IoT networks generate huge amounts of data collected through IoT objects and are vulnerable to malicious targets, resulting in compromised data privacy. For example, in agriculture and farming-related applications, nodes show inaccurate data collection due to unauthorized access to IoT-WSN data, affecting crop yields. Similarly, if patient data routed through IoT-based networks is compromised in healthcare networks, it could seriously affect patient health and privacy. Thus, the sensor-generated data highlights the severe security risks in IoT-based network applications and the importance of strong security measures. This is why routine security must be ensured so that the integrity and reliability of data generated through sensors and physical objects enable the secure functioning of the IoT network.

A growing number of IoT physical objects present a challenge of charging / replacing dead batteries for IoT-based networks. IoT networks aim to conserve energy and implement widespread use of IoT nodes in challenging environments. These nodes have limited power that makes network's life difficult and provides malicious operators greater control, making it hard for nodes to work efficiently over time. Computational complexity and burden on the devices result in high energy consumption.

Clustering techniques power hierarchical networks. It enables the network to conserve energy using a particular schedule, such as wake-up and sleep network nodes. It minimizes retransmissions, which is advantageous for longevity and offers a practical solution in such scenarios. Multi-hop communication in itself is not a new concept. It involves nodes to pass data to the base station concerning the environment's short communication range and limited energy. In the context of multi-hop communication, the compromised node may intercept or alter the data. The inefficient clustering and unsecured routing implementation of IoT-based networks causes reduced network life as well as malicious nodes to infect the network and leak information. The ability to optimize routing data transmission in the clustered network is one of the useful features that can extend the IoT network lifespan. Utilizing network resources in the context of clustered networks is the practice of fixing these issues to keep the network functioning well.

1.4 IoT Routing Security

The IoT involves constrained devices at different layers, which makes it vulnerable to various security risks. These risks include unauthorized access to private data and attacks that can harm the reliability and availability of IoT-based systems. Physical devices and objects are interconnected, and a security breach on one device can compromise the entire network, creating widespread risks. The security threats related

to data privacy, integrity, authentication, and authorization have revolutionized proper encryption and security measures during data transmission within IoT networks. However, in the context of IoT security, given that diverse types of IoT devices have limited resources, low battery life, processing power, and storage — no single security technique can claim to be suited to the needs of all roles.

Fig 1.1 visually represents the IoT standards for restricted systems and associated routing security support mechanisms.

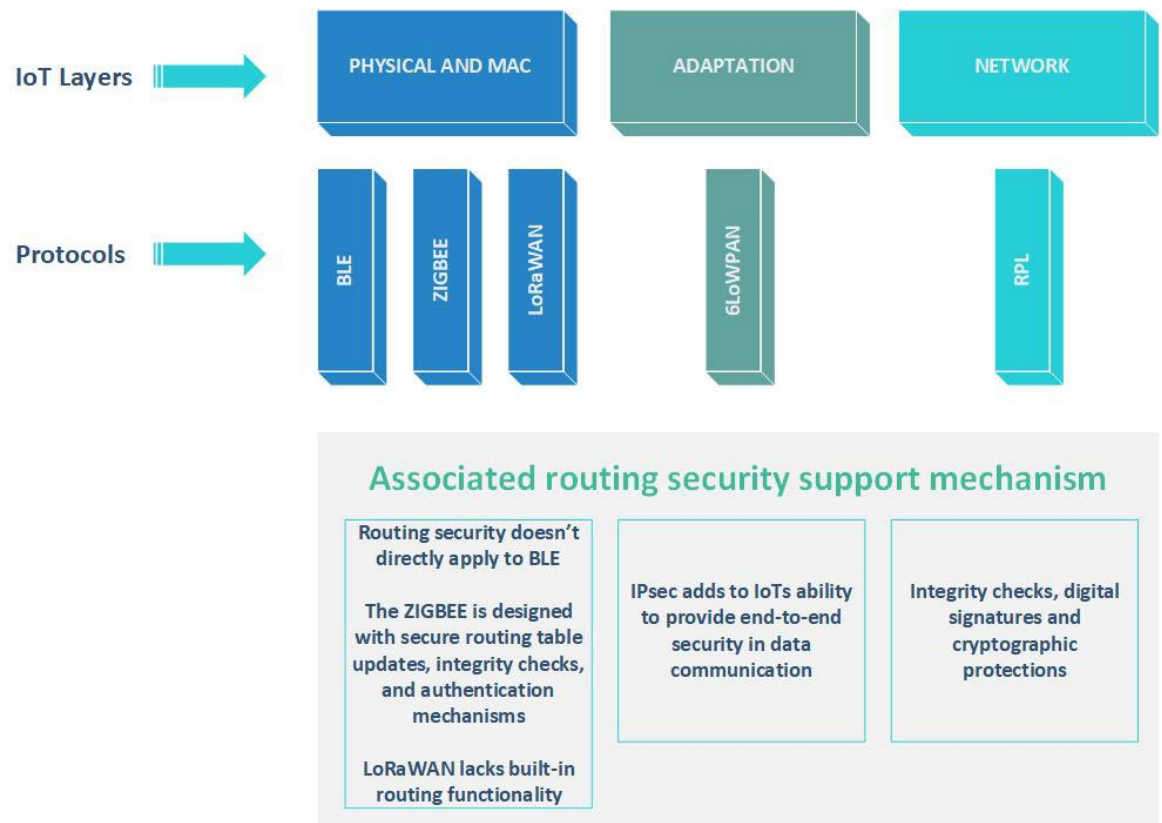


Figure 1.1 IoT standards for restricted systems and security mechanism

The protocol for short-range wireless communication is called Bluetooth Low Energy (BLE) [10], with a similar scope to classic Bluetooth, encompassing monitoring and control applications. BLE is designed to be energy-efficient, making it suited for use with low-power devices capable of handling the demands of IoT applications.

ZigBee operates using low power and is ideal for low-data rate applications [11]. ZigBee is a versatile communication protocol that offers several network topologies to meet different requirements. The protocol can be used with star, tree, or mesh topologies, depending on the application's particular requirements. The network topology used in ZigBee determines the routing algorithm used. Routing security is an important aspect to consider in Zigbee deployments to protect data transmission between devices. Zigbee networks rely on key management to ensure secure communication between devices. The Zigbee routing protocol is designed with features, including secure routing table updates, integrity checks, and authentication

mechanisms. By implementing secure key management practices, utilizing encryption for communication channels, securing the join process, following secure routing protocols, and addressing physical security concerns, the routing security within a Zigbee deployment can be enhanced. It is essential to follow best practices and guidelines provided by the Zigbee Alliance or other relevant organizations to ensure a robust and secure Zigbee network.

Long Range Wide Area Network (LoRaWAN) typically uses a star network topology, where multiple end devices (sensors, actuators) communicate with a central gateway [12]. LoRaWAN supports many devices in a network, allowing for the deployment of extensive IoT solutions.

Network technology offers an abstraction layer known as IPv6 over Low Power Personal Area Network (6LoWPAN) [13], whereby two components of IP security (IPsec) are enabled: Authentication Header (AH) and Encapsulating Security Payload (ESP). The initial component ensures the confidentiality, integrity, and authentication of data, while the second, the AH component, defines application data and IPv6 headers to ensure the integrity of the complete IPv6 datagram. The IPsec design facilitates network layer authentication and encryption of IP packets during a communication session. Routing protocols, including RPL, are used by 6LoWPAN networks to choose the best routes for data transmission inside the network. One significant barrier to adopting IPsec and Internet Key Exchange (IKE) as network layer security in 6LoWPAN is the sensing nodes' limited resource availability.

The IETF working group [ROLL WG] created RPL [14] because the limited resources of the nodes made it difficult for 6LoWPAN to implement routing functionalities. RPL is an IETF standard for IoT networks that manages routing efficiently for resource-constrained devices and networks. Its lightweight, hierarchical routing scheme reduces network traffic overhead and saves energy. By employing integrity checks, digital signatures, or cryptographic protections, the routing information within RPL can be safeguarded.

The nature of IoT networks can be characterized as similar to wireless sensor networks [15] and Mobile Ad Hoc Networks (MANET) [16] and, therefore, prone to similar routing attacks. Proper route information is transmitted from node to node to establish a desired route essential for an effective routing protocol [17]. Route discovery and route forwarding lead not only to the perpetration of malicious activity by the unwanted node but also to several types of routing attacks. A likelihood of scalability based on energy constraints and computation functions with optimal storage is a fundamental requirement in routing protocols. The research in routing security focuses on creating energy-efficient and secure methods that protect data and privacy while using minimum power. The key concern for optimal routing protocol lies in ensuring the security and efficiency of these methods. Fig 1.2 shows the challenging issues related to IoT routing security.

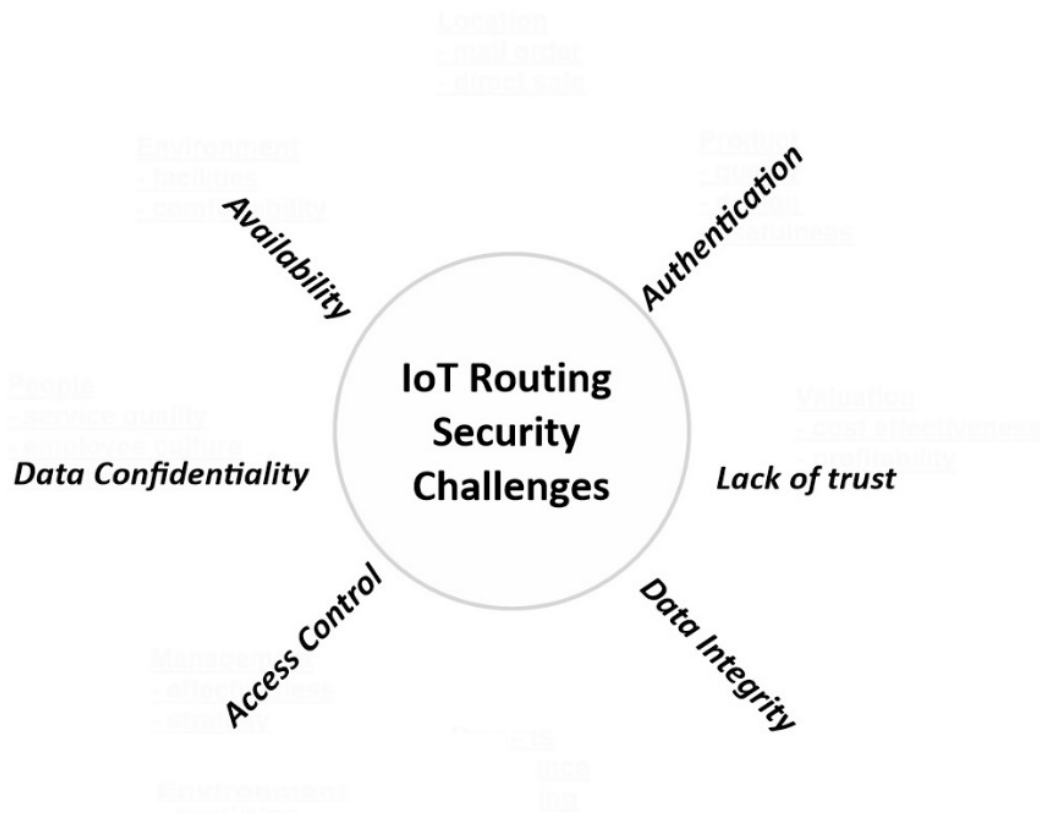


Figure 1.2 IoT routing security challenges

It is worth noting that routing data in such environments is a significant challenge for the purpose of securing data. Routing attacks on IoT include bad-mouthing, grey-hole, denial-of-service, whitewashing, replay attacks, and eavesdropping etc.

Sybil attack: In an IoT Sybil attack, an attacker creates many fake identities or "Sybils" in order to fool the IoT system and obtain unauthorized access to resources or data [18]. Identity theft serves as a starting point for more malicious actions. With greater control over the network, the node may utilize this to promote badmouthing.

Wormhole attack: A wormhole attack in IoT involves an attacker making a tunnel amid two distant points in the network and forwarding the data packets through this tunnel, which can lead to various security threats [19]. These attacks can influence routing protocol by causing traffic to be diverted through the shortest path and altering network topology by forming connections between two network segments.

Sinkhole attack: By employing a routing metric falsely advertised to draw traffic and changing topology, the compromised node with a substantial communication range and many neighbours creates the preferred path for innocent nodes, who take this path as shorter than absolute network paths. A sinkhole attack in IoT redirects packets to a compromised node, leading to many threats, such as data loss, privacy violations, and denial of service attacks [20].

Black hole attack: Black hole attack involves an attacker dropping or deleting all the data packets, which can lead to security threats and data loss [21]. A Blackhole attack causes a breakdown in communication between the trustworthy sender and receiver nodes, which is exacerbated if the malicious node begins changing packets before passing them.

Greyhole/selective forwarding attack: This attack in IoT involves an attacker dropping or selectively forwarding certain data packets, which can lead to threats and data manipulation [22].

Bad-mouthing attack: Bad-mouthing attacks in IoT involve an attacker spreading false information to damage the reputation of a legitimate node or the overall IoT network, which can lead to a range of security threats, such as loss of trust and reduced functionality due to an isolated network [23].

Garnishing attack: The malicious node in the garnishing attack exhibits positive and adverse behavior to gain trust over time and do damage while avoiding detection to the greatest extent possible [24].

Whitewashing attack: This attack occurs when system discovers and separates an unwanted node from the network [25]. The removed node attempts to re-enter a system with a new identification and reputation to deceive the system and get a fresh trust value by deleting or altering data from the network to hide its disruptive behavior.

Hello flood attack: Hello flood attacks in IoT involve an attacker sending many hello messages to flood the network and overwhelm the nodes, leading to threats and network congestion [26].

DoS attack: Denial of Service (DoS) attacks flood the network or a specific node, causing system overload, making it unavailable, or slowing down its performance [27]. Various layers can be affected by the denial-of-service attack.

Eavesdropping attack: Eavesdropping attacks in IoT involve an attacker intercepting and listening to the network communication, which can lead to several threats, such as loss of privacy, data leakage, and unauthorized access [28].

Spoofing attack: In an IoT spoofing attack, an attacker poses as a genuine node in order to obtain access to the network [29]. This can result in a number of security risks, including illegal access, data theft, and privacy violations.

Replay attack: A replay attack refers to an attack where an adversary intercepts and maliciously retransmits previously captured data to gain unauthorized access or perform unauthorized actions [30]. Using network eavesdropping to intercept a data packet is the basis of a replay attack. This attack exploits the deficiency of appropriate authentication and verification protocols in IoT device-to-device communication.

In the case of these attacks, (i) the node's energy can be drained, (ii) network functioning can be disrupted, and (iii) attackers can steal sensitive information. Eventually, IoT routing vulnerabilities can spread false information, block services, manipulate identity, and intercept information about the devices on IoT-based networks. Routing protection specifies every critical aspect, technique, and procedure

that must be adhered to ensure the secure transmission of data at IoT-based networks. However, the IoT network layer is vulnerable to routing attack due to the scarcity of defined security standards at this layer, despite the protection offered for information exchange at the upper layer and the availability of security at the link layer. Numerous publications [31] [32] [33] have covered a detailed study of such attacks. Routing security [34] requirements motivate the research community to improve the routing process via authentication across heterogeneous devices and the provision of information confidentiality and network service availability. Despite the numerous traditional routing protocols, achieving the performance for resource-constrained and dynamic systems takes a lot of work, as they were implemented for resource-rich IoT environments.

1.5 Motivation

The motivation stems from the need to establish foolproof authentication and authorization processes. Prolonging the network's lifetime and reducing energy consumption while maintaining data transmission security are key motivating factors. Further, nature-inspired algorithms have demonstrated their effectiveness in solving complex optimization problems. The motivation is to harness the inherent collaborative and trust-based behaviors observed with different heuristics, translating them into an optimization algorithm that can improve routing security in IoT based networks.

1.6 Methods Overview

The research focuses on improving routing security in IoT-based networks. We implement security at intra-cluster and inter-cluster levels, ensuring the network's integrity. The proposed methods combine cryptographic techniques, optimization algorithms, and multi-level trust systems to improve the security and performance of wireless sensor networks in IoT environments. A promising field in network communication, metaheuristic-based routing techniques provide seamless data flow for physical objects, energy efficiency, and system security. The proposed approaches have successfully enhanced network security, efficiency, and lifespan using advanced techniques like adaptive clustering, blockchain technology, and new cryptographic methods. These improvements are evident in packet delivery ratio, throughput, energy consumption, delay, and network lifetime. Overall, the findings confirm that the proposed solutions effectively address the complex challenges of IoT routing security, offering a promising future for secure and efficient IoT networks.

1.7 Outline of the Thesis

The thesis consists of six chapters describing the entire study in a very concise and precise way. Each chapter is summarised below:

Chapter 2: In this chapter, we present a brief description of the routing defense techniques and previously developed data, communication, deployment, and learning mechanisms of diverse and heterogeneous IoT networks to discuss the broader range of routing defense techniques for routing security at route-over forwarding (layer 3) in

IoT networks. This chapter also covers the background detail of routing security and performance attributes associated with metaheuristics algorithms for the robust operation of the IoT network.

Chapter 3: In this chapter, Dynamic Routing algorithms in the Redundant Byzantine Fault Tolerance with Extensions and Coati-based network, called RB BFT X and CoatiNet, is proposed, which considers the behavioural anomaly detection and role-based access control to identify potential security threats. In order to validate the authenticity of the proposed method, performance evaluation is done based on metrics such as accuracy, precision, recall, and F score, which show improvement with the other existing techniques. Furthermore, the offered configuration capacities make it possible to seize existing regulatory and resource conditions and adjust to the changing conditions.

Chapter 4: As the security and energy efficient techniques of the IoT based networks is dependent on several factors such as clustering, encryption, and optimal route selection etc. Therefore, in this chapter, we have proposed a secure and energy-efficient BONY-ISHO routing technique. BONY-ISHO protocol has four stages, namely initialization, clustering, encryption, and optimal route selection. Since the advantages of the blowfish and honey algorithm are effectively utilized, any intrusion is extremely unlikely and BONY encryption effectively secured data from unwanted access. The performance of the proposed protocol provides better performance than the existing techniques in terms of network lifetime, throughput, energy consumption, PDR, delay, number of alive nodes, encryption time, and decryption time.

Chapter 5: In order to provide secure and optimal routing, this chapter proposed a multi-level hierarchical secure and optimal routing (ML-HSOR) protocol that uses a Markov model with adaptive weighting mechanism to choose the most suitable node as the cluster head (CH), enhancing network lifespan and performance. ML-HSOR addresses the problem of malicious nodes that cause various attacks, such as garnishing and bad mouthing, by introducing a multi-level hierarchical trust evaluation approach. Trust is evaluated at both intra-cluster and inter-cluster levels, considering factors such as interaction trust, data trust, validation trust, transmission trust, and identity trust. This approach significantly enhances security and reliability, a novel contribution that safeguards environmental monitoring systems. ML-HSOR's high detection rate of malicious nodes, achieved through a multi-level hierarchical trust evaluation, ensures the security and reliability of environmental monitoring networks.

Chapter 6: In this chapter, the conclusion, future scope of the current research, and limitations have been discussed. Currently, methods combining cryptographic techniques, optimization algorithms, and multi-level trust systems improve the security and performance of networks in IoT environments. In the future, Incorporating advanced machine learning models, such as federated learning, can improve anomaly detection accuracy while preserving data privacy. Hybrid cryptographic approaches combining lightweight and quantum-resistant techniques can further secure the system against future threats. The future of IoT based networks and secure routing techniques is very bright and strongly opens new flaps for the connected world.

1.8 Chapter Summary

This chapter covers the overview of IoT-based networks, the major challenges, and routing security. We also described the associated routing security mechanisms for various standards based on the Internet of Things. The study also highlights various routing attacks in IoT based networks including bad-mouthing, grey-hole, denial-of-service, whitewashing, replay attacks, and eavesdropping. In this chapter, we cover the overview of the entire thesis, describing the basic idea of each chapter. An overview of the motivation and methods of the entire work is also presented here.

CHAPTER 2

LITERATURE REVIEW

Recent advances in the sensing, communication, and integration of the environment have resulted in networks of physical objects available to make connectivity possible in things. At the same time, developments in the communication of IoT networks have expanded the capacity of intermediate devices and objects to develop secure routing techniques grounded in path. In this chapter, related work comes from research fields: IoT-based networks and hierarchical clustered networks, where we find the inspiration for this thesis; Encryption and Trust, where we find the methods and techniques to address the defined problems; and heuristics, where we learn about the optimization that help to design optimal security solution. Section 2.1 covers the overview of the chapter. Section 2.2 extensively describes the literature that provides routing security for IoT-based networks. Section 2.3 provides the research gaps and limitations of the previous studies. Section 2.4 concludes with a summary of the chapter.

2.1 Overview

IoT represents an interdisciplinary field in connected engineering, offering solutions [35] in various fields, including traffic control, energy management, healthcare, and mobility due to the expansion of IoT [36] and the surge in smart physical objects. Cloud services provide high computing infrastructure with the ability to use and connect to the internet, offering an IoT ecosystem with various stand-alone devices [37] from smart cities and industries. An online search was conducted for the articles on the IEEE Xplore, WoS, and Elsevier's platforms. Fig. 2.1 outlines the search selection steps.

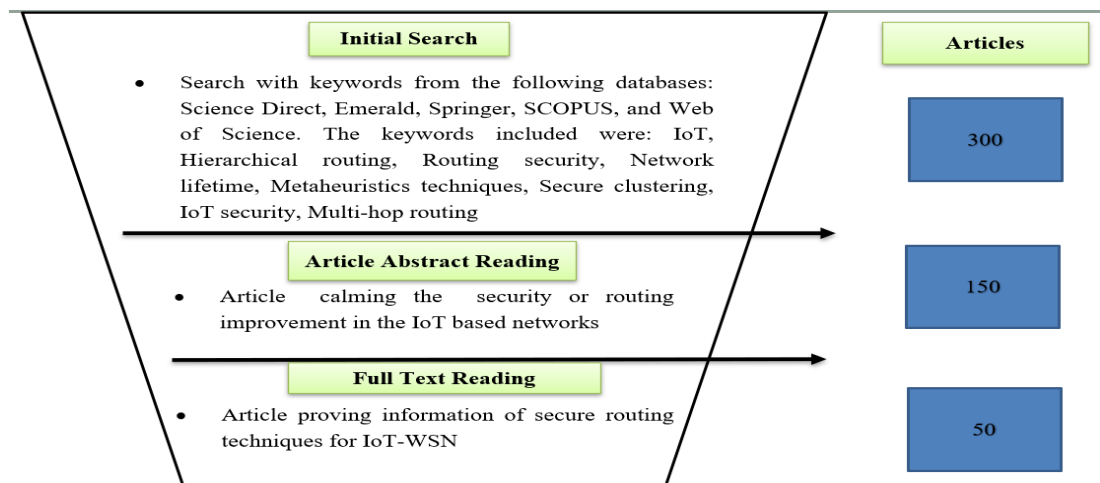


Figure 2.1: Search selection steps

2.2 Security for IoT based Networks

We have explored defense mechanisms [38][39] to describe how security solutions integrate with the IoT infrastructure for a standard set of routing attacks. Literature review establishes an integration perspective, an essential foundation for the security solution development of IoT infrastructure in anticipated heterogeneous systems. We aim to create a cohesive routing security solution structure by consolidating the diverse yet related research on IoT routing security that we have identified.

Bang et al. [40] presented an extensive survey on security breaches and methods for preventing routing attacks in Routing Protocol for Low Power and Lossy Networks (RPL). Their study introduced a categorization system for attacks and their defense mechanisms. Additionally, the paper provided an in-depth analysis of routing attacks from a statistical perspective. Airehrour et al. [41] extensively analyzed the routing protocols and techniques currently in use to ensure safe communication among IoT devices. They extensively examined the security aspects of IoT in their study. Additionally, their research briefly explores the profound implications of trust management within the complex architecture of secure routing. Muzammal et al. [42] conducted a comprehensive analysis of trust-based methods for ensuring secure routing in IoT networks in their study. This study focuses on elucidating the security challenges, particularly emphasizing the RPL routing protocol. Furthermore, the authors proposed several mitigation strategies to address secure routing in IoT networks. Finally, they explored the potential impact of integrating trust-based mechanisms as a security measure within IoT networks. Ahanger et al. [43] comprehensively categorized recent research endeavors focused on applying learning techniques in the context of the IoT. It introduces a unique classification encompassing IoT vulnerabilities, attackers, effects, threats, remedies, and authentication technologies. In order to illustrate the seriousness of the issue and offer organizational knowledge resources for mitigation, the study emphasizes IoT manipulation through passive measures. Yugha and Chithra [44] highlighted the secure routing of information over the internet as a significant challenge for IoT applications, considering limited resources. The survey emphasizes trends and techniques for analyzing IoT layer protocols, notably regarding security and routing. Ejaz et al. [45] emphasized learning methodologies' significant contribution to improving IoT system security at various phases. In addition to classifying communication and computing paradigms, this study reviews learning methodologies for IoT systems. Additionally, it presents research on integrating learning strategies into IoT from an optimization perspective, categorizing optimization objectives as maximization or minimization with corresponding applications. Altaf et al. [46] focused on the models and trust methods developed for the IoT context. They identified trust components and functional requirements specific to the IoT, which must be considered when designing a trust model. Additionally, they discuss several crucial security requirements and explore various solutions available in the existing literature. Noor et al. [47] demonstrate that, despite the possibility that authentication by itself is insufficient to ensure IoT security. Current IoT security techniques strongly emphasize lightweight, shared, and multi-factor authentication, particularly at the network and application levels. Souissi et al. [48] focused on trust management models within the IoT

ecosystem, upholding the trustworthiness of information and ensuring that only dependable data could be transmitted to the application layer. Additionally, they presented well established approaches for trust and classified them based on the trust evaluation method employed. According to the Mousavi et al. [49], essential security considerations in the IoT context include privacy, safe storage and administration, authorization, exchange of information, and access control. All of these factors also present significant challenges. Additionally, the physical layer security can facilitate the exchange of cryptographic keys or directly enable safe data communication.

2.2.1 Routing defense in IoT

In order to shed light on the latest research in this field, we explore the literature about study of defense classification and attacks on routing in IoT networks. Our focus has been directed explicitly toward selecting and examining studies emphasizing crucial aspects of this research field.

Solutions based on encryption is suggested by some authors. In this category of solution mechanism, the techniques used for the routing security consider the cryptographical methods for executing the defense activities. By encrypting the data before transmitting it over the network and decrypting it at the receiving end, the IoT system can ensure that only the intended recipient can access the data [50]. Using an end-to-end feedback technique to identify and mitigate unreliable links—mainly packet-dropping attacks—improves network layer routing security. The method secured packets in transit using lightweight symmetric-key cryptography, guaranteeing effectiveness and resilience against various denial-of-service attack variations, including greyhole attacks [51]. Hammi et al. [52] ensured routing security at the network layer by including authentication, integrity, and confidentiality measures in the routing process. It uses cryptographic methods for data encryption to prevent unwanted access, routing message integrity verification, and node authentication. By defending against typical threats like replay attacks, spoofing, and eavesdropping, these measures make sure that data is transmitted securely over the network. Liu et al. [53] provided routing security using reserved bits of the Zigbee MAC header's frame control field and specific lightweight cryptographic techniques. Encrypting protocol-specific unique data before transmission and decrypting it at the receiving end represents a crucial security measure within the IoT ecosystem. Kim and Suh [54] discussed the susceptibility to eavesdropping on infrared communication, specifically as it applies to the Internet of Things devices, and suggested a solution to improve network layer routing security. It emphasizes using simple, practical, and cost-effective encryption to secure IR communication. A timer within a microcontroller, often found in a remote control, is cycled repeatedly to provide the encryption key. The life of this key is constrained because this data must be generated each time the power button on the remote control is touched.

Some authors have proposed solutions based on packet-filtering. By analyzing the content of the messages and filtering out the suspicious ones, the IoT system can protect the network from malicious nodes. An IoT system can effectively fortify its network against the intrusion of malicious nodes by meticulously analyzing data content within incoming messages and the subsequent filtering of any suspicious or

potentially harmful ones. By incorporating several forms of data analysis within the RPL-based IoT networks, Ribera et al. [55] proposed hybrid detection approaches. With reduced false positives and increased accuracy, the system's ability to adapt and respond within the RPL-based IoT networks, while the extensive data usage during the filtering stage, acted as a detection technique to cover a broader range of threats. Pu and Choo [56] suggested a lightweight Sybil attack detection technique for improving network layer routing security in IoT based on a Physical Unclonable Function (PUF) with a Bloom Filter. PUF employs DODAG hashes for PUF replies to make a Bloom Filter array against sybil nodes.

Numerous aspects of node data, including energy, load, location, and others, are used to choose the best route for data packets traveling over the IoT nodes from source to destination. Routing protocols, which base their judgments on the node data of the route, choose this route. The node selects the best neighbor for forwarding based on this route information. This defense mechanism effectively manages security by employing protocol-specific data as a fundamental component of the overall security infrastructure. This approach to safeguarding IoT systems and data is characterized by strategically utilizing particular protocols and technologies' unique attributes and requirements. In the framework of RPL, Ankam and Reddy [57] addressed the use of different protocol data for routing security. It is specifically concerned with reducing flood attacks caused by DIS (DODAG Information Solicitation). Topology control messages are a component of the protocol data utilized to enhance routing security. By analyzing the transmission and reception of packets, these messages constitute a forward ratio of UDP and various versions of DODAG. Farha et al. [58] applied date and time sequence to provide the timestamp data and proposed an enhanced timestamp system by replacing frame counters with synchronized timestamps to maintain the freshness of messages and effectively protect ZigBee networks from replay attacks. SeungJae et al. [59] proposed a countermeasure for replay attacks in LoRaWAN, which uses XOR masking and unique DevNonce values to keep the network layer secure. With the help of a global end-device identifier and a random code value, security is improved without requiring complex computations, and intercepted join request packets cannot be used by attackers. Farha et al. [60] proposed a data mechanism that uses timestamp data for routing security in ZigBee. Yin et al. [61] presented a secure routing algorithm based on a multiple attribute decision-making model that takes into account node qualities including energy transmission efficiency, load, and packet loss rate in order to mitigate selective forwarding attacks in scale-free networks. The algorithm makes use of the route data, which includes node load—the total amount of data a node needs to send and forward from other nodes in a unit of time—energy transfer efficiency, which represents the energy state of node j at the moment by taking into account both the energy consumption and residual energy—and packet loss rate, which predicts a node's likelihood of being malicious based on the behavior of selective forwarding attacks, which is basically malicious packet loss. Madria et al. [62] proposed a secure routing protocol based on route discovery data by utilizing pair-wise shared key and routing beacon between neighbors. Nodes register their neighbors as parents after accepting their neighbor's first routing beacon before forwarding a changed beacon. In the recursive algorithm, each node chooses as its parent the first neighbor it hears. The authors provided a mechanism to secure a route

against wormhole attacks. Likewise, Deepavathi et al. [63] developed the ESWSIoT protocol, which uses check packet timestamps to calculate the shortest path and asks intermediary nodes to report their identities within a set time to prevent wormhole attacks. To provide a safe path for communication, they combined the Schnorr Digital Signature Scheme and Check timestamp settings. Kaliyar et al. [64] proposed methods for protecting the RPL-based IoT network layer from wormholes and Sybil attacks. Each non-leaf node maintains a Sybil detection table, the entries of which are updated in response to data packet reception. A periodic network-level timer is used to initiate the detection method for Sybil attack detection. It uses a wormhole detection table to identify wormhole attacks, using nodes in the DODAG structure to run detection algorithms regularly. In susceptible IoT network environments, the route data utilized in these techniques is essential for guaranteeing the security and integrity of routing activities. Sharma et al. [65] proposed a two-step verification model, a neighboring nodes cooperation technique, to make the route more secure and protected from black hole attacks. They validated the black hole node's participation in a neighboring node coordination approach using a suspect-reply packet from authentic nodes and no response from black hole nodes, discovering a secure route with this data.

Also, a channel-based mechanism is a security approach that relies on the communication channel's unique physical attributes or characteristics to grant or deny access to a system or facility. Jakubisin et al. [66] focused on the vulnerability of underwater routing systems to malicious behavior during the network route discovery process. It addresses protecting route discovery in underwater Internet of Things networks. It utilized simulation-based results to examine how malicious behavior and channel conditions affect link and route discovery through beacon transmissions. This method relies on the emission of beacons to illustrate the channel's effect. Li et al. [67] proposed a technique by using the concepts of both modulation and carrier frequency in the communication channel. Wormhole attacks in wireless networks were detected using physical layer network coding. The authors have made the network layer routing security more efficient without utilizing additional hardware or node time synchronization. Huang et al. [68] introduced a physical layer authentication approach for improving network security. In the proposed approach, the unique identifications for IoT devices were established by utilizing channel state information. This was based on training a CNN with CSI data from multiple places. Based on this approach, spoofing and Sybil attacks were detected from the data. Chulerttiyawong and Jamalipour [69] described a network-level security solution that uses the unique characteristics of UAV flight patterns and network dynamics. It employs physical layer data of radio signals to detect the Sybil attack that minimizes additional communications overheads in FANETs-based IoFT. Wu et al. [70] addressed physical layer authentication (PLA) for spoofing detection in the context of routing security in the Internet of Things. It presents a game theoretic method to simulate how several malicious spoofers and legitimate receivers interact during the PLA process. The authors used physical layer features to facilitate upper-layer encryption authentication based on radio frequency fingerprint and carrier frequency offset and employed received signal strength indication and channel impulse response to identify genuine transmitters. Anajemba et al. [71] presented a counter-eavesdropping method designed to mitigate eavesdropping attempts and improve privacy in Wireless Industrial IoT

connections. The concept suggests an ideal method for approximating the channel to counteract eavesdropping assaults effectively. When detecting forgery in IoT systems, the received signal strength (RSS) is crucial, presuming a reasonable distance separates the attacker and victim. Techniques based on received signal strength apply to various attacks since signal strength is closely related to the transmitter's location. Saxena et al. [72] suggested a novel approach to network layer routing security for Internet of Things networks, emphasizing identifying and locating attacks based on received signal strength. They used to receive signal strength pattern deviations to identify compromised nodes. Different methods [73] [74] apply a threshold function to identify possible attackers, enhancing resilience and reliability, and finally detect routing patterns to prevent HELLO flood attacks to improve routing security at the network layer. Ghahramani et al. [75] proposed an energy-efficient method using RSS to protect IoT protocols against denial-of-service attacks. Their idea is to utilize the distance generated by the initiator node and calculate the power of the received signals to detect denial-of-service attacks. Nguyen et al. [76] proposed a two-stage strategy to detect impersonation attacks in IoT networks. Authors classified nodes as legitimate or illegitimate based on radio frequency attributes. Then, they employed node authentication to monitor changes in node topology based on the RSS indicator for routing security in IoT networks.

Device-based mechanisms are to improve route security and protect the resources within each IoT device so that the system can detect spoofing and denial-of-service attacks by verifying the validity of network requests. Hendaoui et al. [77] proposed a node authentication solution by generating signatures in a distributed manner from the IoT node keys. This method constructed keys from the distributed devices to generate and validate signatures as part of the authentication process. Pu et al. [78] proposed a device-based countermeasure technique to protect IoT networks from Sybil attacks. They proposed the Proof-of-Assignment (PoA) method to establish a device-based defense by issuing a unique task to each device. Kponyo et al. [79] proposed a lightweight defense system to detect DoS attacks using IoT device CPU and memory behavior. The method obtained 100% accuracy in differentiating between DoS and normal traffic, and it showed great efficiency in CPU and memory consumption, detection and mitigation durations. Wu et al. [70] discussed how IoT devices are susceptible to impersonation attacks. To identify the malicious nodes of an attack, the authors evaluated the RF signatures of IoT devices. Nosouhi et al. [80] leveraged beam pattern uniqueness features in IoT devices based on mmWave standard to design a security mechanism for wireless spoofing attacks. Each base station / access point is responsible for measuring unique beam features of incident RF signals. Without placing more stress on the network or the devices, this method guarantees high precision in identifying unauthorized devices.

A learning-driven approach analyses large amounts of data from IoT systems to identify patterns, anomalies, and potential security threats. The traffic, behavior, and trust of the IoT system serve as the foundation for the study. Using behavior-based approach for IoT routing security involves analyzing the historical data of node's route discovery and entities within an IoT system to detect anomalies, identify threats, and enhance overall security. Behavioral analysis focuses on understanding typical

patterns of behavior and identifying deviations that could indicate security breaches or unauthorized activities. Node's behavior shapes its reputation, as a device's consistent patterns of actions influence how learning systems perceive and evaluate their trustworthiness, ultimately defining their reputation. By collecting feedback from the other nodes in the network and assigning reputation scores to the nodes based on their behavior, the IoT system can identify any nodes that are spreading false information and prevent them from doing further harm. Specifically concentrating on connected automobiles in smart city environments, Huber and Kandah [81] introduced a Behavior-based Enhanced Trust management System (BEAST) for IoT security. The system uses deep learning to create a localized behavioral model by gathering driving statistics and automobile environmental data. Because it ensures that only validated data influences essential decisions within the network, this form of dynamic trust management can mitigate various cybersecurity vulnerabilities. Patel and Jinwala [82] recommended a selective forwarding attack detection mechanism following data forwarding behavior and reputation evaluation for the selection of parent node in IoT environment. Kumar et al. [83] presented a game theoretical protection mechanism called "GameTrust" that protects network layers from Sybil attacks, which exploit network reputation systems. In addition to using centralized and decentralized methods for calculating trust values, this mechanism simulates node interactions as a zero-sum imperfect information game and dynamically adjusts trust levels between nodes. Through the establishment of a global trust barrier for nodes to stay trustworthy, it seeks to make attacks expensive.

Using traffic analysis for IoT security involves monitoring and analyzing network traffic generated by IoT devices to detect anomalies, threats, and unauthorized activities. Traffic analysis can provide valuable insights into IoT devices and help identify potential security breaches. With a primary focus on identifying and averting Distributed Denial of Service (DDoS) attacks, the Gupta et al. [84] suggested a machine learning-based approach for routing security in Internet of Things networks. The method works at the network layer, especially the local network gateway or router, which is in charge of keeping an eye on all incoming and outgoing data. Because of this tactical placement, suspicious activity can be identified, and mitigating measures can be applied at the traffic management point. Hussain et al. [85] addressed using a deep learning model—specifically, a Convolutional Neural Network (CNN) model called ResNet—to improve network layer security against DoS and DDoS attacks in IoT contexts. The procedure entails transforming network traffic data into a format for images that ResNet can handle quickly. Examining patterns in the transformed picture data is a critical factor in detecting and differentiating different kinds of DoS and DDoS attacks on IoT devices due to high recall and precision rates compared to conventional security systems. Mihoub et al. [86] presented a machine learning-based detection of IoT distributed denial of service (DDoS) attacks. In this multi-class classifier scheme, the design blocks rate-limiting traffic as a particular countermeasure for mitigation from recognized sources depending on the detected attack type. Choukri et al. [87] presented a deep learning-based routing security system for blackhole attack detection in unprotected RPL networks. The proposed framework examines network traffic to extract features and establish intrusion detection thresholds. The framework uses an offline process driven by a deep learning algorithm to assess data and identify

anomalous behavior. With the proposed framework exhibiting excellent accuracy and a consistent error rate, this method enables the effective detection of routing attacks.

Trust evaluation has become an alternative mathematical method to protect IoT devices from potential threats, enabling routing security. Trust is commonly used to promote data trustworthiness and devices' integrity in IoT security. It develops over time through authentication. RFTrust, a trust-aware security mechanism proposed by Prathapchandran et al. [88], describes a mechanism that uses the RPL protocol to identify and prevent sinkhole attacks in IoT. This mechanism combines Subjective Logic (SL) with the Random Forest (RF) algorithm to improve network layer security through trust management. Assessing direct and indirect trust metrics local maximum and sparse connectivity problems isolates malicious nodes. This mechanism ensures reliable data transfer in an IoT environment by enabling a dynamic and adaptive security framework. Khan et al. [89] proposed a trust-based, energy-aware routing algorithm called ETERS. Using a multi-trust strategy, ETERS can handle adversarial situations in WSNs to counteract internal attacks. Their idea is to use an irregular attenuation factor to account for external influences impacting communication trust and an efficient cluster head selection algorithm (ECHSA) to increase cluster head performance and accelerate recoveries of trust values under attacks through a Beta distribution-based trust function. Altaf et al. [90] presented a context-based trust evaluation system (CTES) for mitigating service-oriented attacks in IoT networks with an emphasis on the routing security of the network layer. It evaluates the reliability of nodes in an IoT infrastructure for smart cities by combining direct observations with indirect recommendations. CTES dynamically assigns weights to these observations and recommendations, enabling it to effectively detect and counteract malicious behavior from nodes executing Sybil attacks. Similarly, trust evaluations [91] [92] [93] [94] [95] have employed various learning mechanisms that models have primarily focused on the anomaly detection and that the road to IoT reliable data transfer has to start from the accurate and predictable trust computation. Kalkan and Rasmussen [96] worked on protecting routing on the network layer through decentralized communication and trust methods, and it proposes a trust framework for service discovery among IoT devices. Distributed Hash Tables (DHT) are utilized for device trust values maintenance and decentralized service discovery. Results from security research and simulations show how well the framework works to prevent bad-mouthing attacks and provide dependable service discovery in the context of the Internet of Things.

2.2.2 Metaheuristics for secure IoT routing

The evolution of IoT has brought about many improvements in the connectivity of physical objects for data transmission in networks. Data transmission integrity and reliability in IoT devices have always been challenging due to their limited processing capabilities and energy constraints that are susceptible to various network layer attacks, such as Black Hole, Sinkhole, and Wormhole attacks [97]. The need for more resources like storage and power presents a significant challenge in IoT networks. Early sensor energy depletion in IoT reduces network lifespan, and disrupted communication at network nodes leads to frequent routing adjustments. Recent research uses various heuristics to find the best ways to solve these problems.

Heuristic algorithms rely more on the specific problem they are applied to than metaheuristic algorithms [98]. These algorithms are limited to particular specific challenges. In contrast, meta-heuristic algorithms can be applied to nearly any optimization issue due to their utilization of the black box optimizer [99]. Such algorithms play a crucial role in solving complex problems, with characteristics such as self-organization and decentralized as integral components alongside team intelligence [100]. These integrates intelligent behavior observed for solving complex problems of physical phenomenon in nature, leveraging the interactions between living beings to establish problem solving approach at various group levels. Heuristics approaches are one of the best practices to enhance routing and security in IoT networks. IoT devices are expected to counteract various attacks for secure and reliable communication; therefore, robust heuristic methods are to be utilized in routing and security. Heuristic approaches, known for their problem-solving capabilities, especially in pervasive environments, are particularly suitable for addressing the routing security challenges in IoT networks. In IoT, it is essential to understand the security of IoT networks against evolving intricacies of network layer attacks. Such mechanisms require developing innovative strategies to mitigate such threats effectively by blocking harmful and undesired traffic.

By analyzing existing studies and research, the review seeks to identify common metaheuristics techniques used and the corresponding algorithms employed to counteract IoT routing security threats. Driven by the optimized solutions of heuristics in IoT routing security, this literature survey aims to explore the following:

- Which metaheuristic algorithms are used for secure routing?
- What attributes of secure communication are focused in heuristics?
- Which are the metrics used to evaluate metaheuristic algorithms for secure and optimized routing?

Solutions to IoT security challenges through heuristics have recently received much attention in the literature. Considering these recent studies, it is essential to comprehend how metaheuristic approaches might provide the routing and security solutions of ever-growing IoT networks.

Jay Kumar Jain [101] aimed to secure routes with less energy spent in IoT. For this, they utilize genetic algorithm-based crossover for biometric authentication. The result was improved security through fingerprint-based unique value and energy-efficient routing. Their proposed algorithm provided a potential model for securing IoT networks while maintaining energy efficiency. Ji et al. [102] focused on avoiding malicious network analysis in IoT networks by modifying the network topology using closeness centrality. The authors utilized greedy and simulated annealing algorithms and developed update closeness and fast top-rank algorithms for efficient computation of closeness value/rank, demonstrating the efficiency of pruning algorithms in reducing computational time. They outperformed baseline algorithms and are computationally efficient. Majid Alotaibi [103] proposed a novel model for providing security to IoT-based WSNs employing the CM-MH algorithm and the Improved Blowfish algorithm. They successfully achieved better security and efficient routing, demonstrating the effectiveness of the CM-MH model in IoT-based WSNs. Their approach was superior to other approaches regarding security, efficiency, and energy

conservation in WSNs. Fatani et al. [104] used deep learning and improved transient search optimization to make intruder detection for IoT. CNN is used for feature extraction. An intrusion detection method was successfully implemented inside the Internet of Things framework, incorporating deep learning with an enhanced optimization technique for IoT security. Zuleyha Akusta Dagdeviren [105] developed a method to monitor IoT networks using an energy-based link system. The authors utilized a metaheuristic algorithm to improve WCVC solution quality and find optimal solutions in small-size instances. Salim et al. [106] proposed an approach to protect IoT data using primes and compressive sensing in WSNs based on IoT, utilizing prime number properties for efficient clustering and routing and combining the CS method with the RSA algorithm for security and data compression. Their technique enhances power efficiency in IoT-enabled WSNs, provides load balancing and high security, and introduces an adept reconstruction algorithm to significantly improve data restoration, ensuring efficient and secure data gathering and reconstruction. Anusha et al. [107] designed an effective IDS for IoT networks, combining the GSO algorithm and PCA for IDS using the NSL-KDD dataset. Compared to other methods, it has better precision, detection rate, memory, accuracy, and FAR, providing an effective solution for recognizing various attacks on IoT and enhancing network performance. Jain et al. [108] proposed making clear and well-organized healthcare tracking systems that protect healthcare department information using cloud computing and IoT. IoT smart appliances and metaheuristic swarm intelligence are used in a hybrid system to store data. Their technique provided effective clinical healthcare solutions and data protection using blockchain. Ahmed et al. [109] proposed a GA-based energy optimization procedure for intelligent IoT applications using a Genetic Algorithm (GA) for optimization. The authors focused on the intelligent Internet of Vehicles, integrating IoT with vehicular networks for improved performance in IoT applications through energy optimization. Singh et al. [110] proposed a technique for medical treatment based on a genetic algorithm that makes energy-efficient clustering protocols in heterogeneous WSNs. The authors discussed various clustering protocols and approaches for energy management in WSNs, highlighting the importance of clustering protocols in enhancing WSNs' energy efficiency. They emphasized the need for effective clustering protocols for sustainable and efficient WSNs. Almuqren et al. [111] proposed a method to identify botnet attacks in IoT with classification using the MFFO algorithm and a CNN-QRNN mixed model to find botnets for high performance in botnet attack detection and classification. Biradar and Mathapathi [112] proposed an optimal cluster head selection model that is more energy-efficient and safe in the WSN framework. The model was effective regarding residual energy, throughput, and delay based on various restrictions, utilizing the SWFU-CMO method. Hijazi [113] proposed HHO, an effective centralized and distributed model for detecting smart IoT device attacks. HHO was used to find IoT botnet attacks and improve RWN and FS for finding IoT hacking attacks. Prasad and Periyasamy [114] proposed a blockchain and deep learning technique with bio-inspired clustering for safe transportation in WSNs to increase security and optimize performance by decreasing latency, minimizing packet loss, and enhancing energy efficiency. Hosseinzadeh et al. [115] suggested a Fire Hawk optimizer algorithm to provide security optimization, demonstrating efficiency in network security over other methods. Dey et al. [116] reviewed feature selection techniques for cybersecurity and emphasized the role of

wrapper-based feature selection techniques in improving detection accuracy in cybersecurity models. Sharma et al. [117] proposed a protocol called BONY-ISHO, combining blockchain technology, a hybrid Blowfish-Honey cryptographic technique, and the Improved Spotted Hyena Optimization algorithm to provide secure and authenticated clustering, data protection and optimized data routing to enhance the security and efficiency of clustering and data routing for IoT systems in smart cities. BONY-ISHO protocol demonstrated superiority in security, efficiency, and energy conservation compared to traditional methods and is highly effective in addressing security and efficiency challenges in IoT-based WSNs, offering a comprehensive enhancement of network security and data transmission efficiency in the context of smart cities. Sharma et al. [118] proposed a multi-level secure method to enhance IoT-WSN routing using the PL-COA algorithm and multi-level trust evaluations. The authors achieved high routing performance with minimal energy consumption and increased network security, indicating the effectiveness of the methodology in diverse IoT-WSN applications and environments. Reshi et al. [119] developed a novel mitigation algorithm to prevent black hole attacks in the IoT using a solution creating a list of authentic nodes, monitoring packet forwarding, and dynamically updating routing tables. Their strategy significantly improved throughput and PDR, closely resembling the performance of an unaffected network. Maharajan and Kumar [120] proposed an efficient key management system using a whale optimization algorithm and Two-Fish-based 128-bit cryptographic key management for IoT to enhance secure data transfer cybersecurity strategy for maritime applications. Rehman et al. [121] proposed a smart IoT agriculture system Using direct trust computation and metaheuristics-based optimal route discovery for smart agricultural sensors. The Authors demonstrated significant improvement over other methods with improved performance and reliability of smart agricultural networks using an intelligent optimization model.

Table 2.1 presents the metaheuristic solutions that have come up in recent years. Nature-inspired metaheuristics algorithms have garnered much attention due to their theory of natural selection, collective intelligence, behavior of living organisms, and decentralized and self-organized.

Table 2.1 Metaheuristics classification

Algorithms	Evolutionary Algorithms	Swarm Intelligence	Local Search	Other Metaheuristics
Genetic Algorithm [101]	√			
Simulated Annealing [102]			√	
Crossover Mutated Marriage in Honey Bee Algorithm [103]		√		
Differential Evolution [104]	√			

Genetic Algorithm [105]	√			
Bees and Genetic Algorithm [106]	√			
Glow-worm Swarm Optimization algorithm [107]		√		
Firefly algorithm [108]		√		
Genetic Algorithm [109]	√			
Genetic Algorithm [110]	√			
Modified Firefly Optimization and Chaotic Butterfly Optimization Algorithm [111]	√			
Cat and Mouse Optimization [112]		√		
Harris Hawks Optimization [113]		√		
Tasmanian Devil Optimization [114]				√
Fire Hawk Optimizer [115]				√
Grey Wolf Optimization Algorithm and Binary Gravitational Search Algorithm [116]		√		
Improved Spotted Hyena Optimization [117]		√		
Polarity Learning-based Chimp Optimization Algorithm [118]		√		
Genetic Algorithm [119]	√			
Whale optimization Algorithm [120]		√		
Hill Climbing Algorithm [121]				√

Based on the metaheuristic's algorithms adapted by various researchers for routing security, the literature studied can be largely categorized into the security services. Table 2.2 illustrates the various security services incorporated in the referenced metaheuristics studies, as well as their implementation in the study.

Table 2.2 Security Services

Article	Confidentiality	Authentication	Data Integrity	Non-Repudiation
[101]	√	√	√	
[102]	√			
[103]	√	√		
[104]			√	
[105]			√	
[106]	√		√	
[107]			√	
[108]	√	√	√	√
[109]		√		
[110]			√	
[111]	√	√	√	√
[112]		√		
[113]		√	√	
[114]	√	√	√	
[115]		√	√	
[116]	√		√	
[117]	√	√	√	
[118]	√	√	√	
[119]	√	√	√	√
[120]	√		√	
[121]	√	√	√	

Various performance metrics used for evaluation also provide the basis for comparison between heuristics. We recognize standard metrics for evaluating metaheuristic algorithms in secure and optimized routing. Thus, research focuses on evaluating various metrics in algorithms so that performance evaluation is successfully completed to demonstrate the optimized result.

Fig. 2.2 shows the utilization of the various performance metrics in the referenced literature from 2019 to 2024. The top five metrics of interest are delay, throughput, energy consumption, PDR, and network lifetime, with contributions of 13%, 12%, 10%, 9%, and 9%, respectively.

Different research projects have employed various routing security mechanisms for protecting IoT routes. The contributors have significant threat insights into IoT routing for multiple approaches and methodologies. However, the challenges from the viewpoint of data, communication channels, deployment, and learning mechanisms of diverse and heterogeneous IoT networks should also be considered. This includes developing lightweight encryption methods tailored for the constrained resources of IoT devices. Data protection could be enhanced by developing adaptive encryption

algorithms that adjust based on network conditions and threat levels. The investigation of privacy-preserving strategies is another important research area mentioned. The statement calls explicitly for researching novel approaches to protect data privacy while in transit, mainly when encryption might not be practical due to resource limitations. Explore the design of secure communication protocols that can withstand various attack vectors specific to IoT. This involves enhancing the security of both the physical and network layers to prevent eavesdropping, spoofing, and other forms of cyber-attacks.

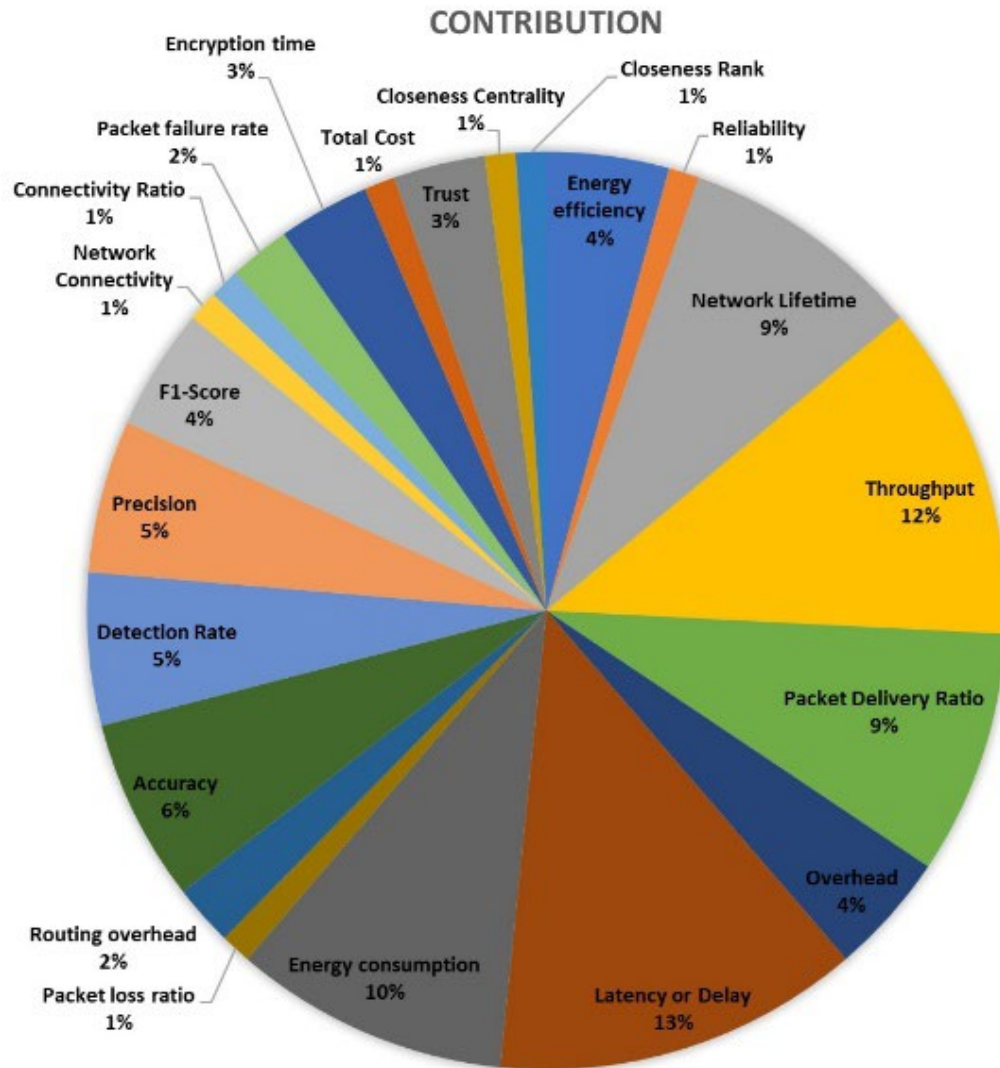


Figure 2.2 Metrics contribution

The creation of decentralized security mechanisms is recommended to improve distributed routing security. By doing this, communication pathways may be made more resistant to attacks and centralized points of failure. Utilize machine learning and artificial intelligence to predict, detect, and respond to security threats in real-time. Investigating federated learning for privacy-preserving collective intelligence and

adaptive security measures tailored to the evolving landscape of IoT threats could be pivotal. Through sophisticated data analysis, using machine learning algorithms to examine trends and abnormalities in network data to anticipate and counteract possible security threats in real time seeks to improve the proactive protection capabilities of IoT networks.

2.3 Research Gaps

IoT-based networks require robust systems that are utilized to examine if a node is authentic or if data is authorized and make sure that only legitimate nodes are allowed to join the network. Cryptographic methods on the basis of keys and mathematical models can provide complete security measures to a specific extent and offer reliable data transmission. However, traditional cryptographic key lengths cannot accommodate the resources of IoT nodes that are restricted in computations and storage. Hence, it is not easy to maintain the integrity and security of IoT networks. Also, poor clustering and routing can reduce the network's lifespan. WSNs typically operate in resource-constrained environments, where energy conservation is of utmost importance.

We realize several research findings related to the security and efficiency of IoT routing. First, it has been identified that choosing a route without considering both the distance to the destination and the quality of the communication link significantly decreases the network's lifetime, leading to inefficiencies. Additionally, if a Cluster Head (CH) within the network is compromised, it can disrupt the entire communication flow, highlighting the vulnerability of centralized network points. The computing demands on IoT devices are a problem because they lead to higher energy use, which is a major limitation in IoT systems. Real-time environmental monitoring systems are particularly prone to security threats, which can further reduce the network's lifespan.

2.4 Research Objectives

The primary aim of this research is to design and develop secure routing techniques for IoT based networks. In order to accomplish this aim, the following Research Questions (RQs) have been established:

- **RQ1:** What are the state-of-the-art secure routing techniques used in IoT-based networks?
- **RQ2:** How can a framework for cluster-based security be designed to enhance the security of IoT-based networks?
- **RQ3:** What are the key factors to consider when implementing techniques for optimal routing and secure communication in IoT-based networks?
- **RQ4:** How do the newly developed routing and security techniques compare to existing approaches in terms of efficiency, scalability, and security?
- **RQ5:** How does the proposed cluster-based security framework impact the overall network performance and resilience in IoT environments?
- **RQ6:** What are the optimal techniques for ensuring both efficient routing and secure communication in IoT-based networks?

Hence, the main goal of our research is to strengthen the routing security of IoT-based networks. For finding the solution to the above queries, the Research Objectives (ROs) of the work undertaken are:

Research Objective 1: To perform the systematic literature review on secure routing techniques used in IoT based Networks

Research Objective 2: To develop framework for cluster security in IoT based networks.

Research Objective 3: To implement technique for optimal routing and secure communication.

Research Objective 4: To do a comparative analysis of proposed approaches with existing approaches.

Consequently, this study contributes to security solutions in IoT routing that can handle emerging threats by understanding the role of different security methods in envisioning the application of secure and reliable IoT-based networks.

2.5 Chapter Summary

In this chapter, the routing defense techniques are discussed in literature using data, communication, deployment, and learning mechanisms of diverse and heterogeneous IoT networks. The literature review is presented based on a broader range of routing defense techniques for routing security at route-over forwarding (layer 3) in IoT networks. We also study routing security and performance attributes associated with metaheuristics algorithms and provides an exhaustive review of secure routing across different heuristics techniques for the robust operation of the IoT network. Overall, this chapter sheds light on the solutions in IoT routing security highlighting research gaps and objectives. Chapter 2 provides a comprehensive overview of existing research on secure routing in IoT networks. It explored various approaches, including encryption techniques, packet filtering, protocol-aware strategies, channel and device-level methods, trust-based models, machine learning solutions, and optimization algorithms. While notable advancements have been made, challenges such as limited device resources, ever-changing network conditions, and centralized security flaws persist. Addressing these gaps, the following chapters introduce a cluster-based framework and innovative secure routing techniques specifically designed for the unique demands of IoT environments.

CHAPTER 3

A MULTI-CLUSTER SECURITY FRAMEWORK FOR IOT BASED NETWORKS

IoT gadgets usually work with limited resources, making it hard to keep them secure without compromising performance. Moreover, the vast number of linked devices makes it easier for hackers to find weak spots to attack [122]. Blockchain provides a way to keep data safe and clear without central control, but it needs a lot of computing strength and space, which might not work well for IoT gadgets that have little power [123]. Like how a detective spots something out of place, machine learning [124] helps find odd patterns. Still, these brainy calculations need much power, which might be too much for more straightforward gadgets. Fog computing is like moving the brain closer to the senses [125]. It takes the heavy lifting of crunching numbers and storing information closer to where the action is, like IoT devices. This move cuts down on wait times and makes things safer by handling data right where it happens. In Secure Multi-Party Computation (SMPC), multiple people can work together on data analysis without having to show what each person brought to the table [126]. This keeps everyone's information private but still lets the data crunching happen. Even though the safety provided by distributed ledger technology [127] is solid, it does demand many resources. Also, systems that use AI to find threats do a good job, but they might need more computing power than what IoT gadgets have on hand. These methods need to be designed with the limitations and needs of IoT environments in mind [128]. Hashing techniques are helpful in protecting data and detecting any changes in the data flow [129]. The Secure Hash Algorithm (SHA 2) creates fingerprint verification and identification of alterations to safeguard the data [130]. There are other variants like SHA 3 [131] that have some strategies for security in the future. Message Authentication Codes (MAC) [132], which includes HMAC [133], utilize hash functions along with keys to give assurance to the receivers that messages have not undergone alterations. Management and deployment of keys also act to maintain communication channels. For this purpose, Pre-Shared Keys (PSK) [134] can be used, but they come with an added need for distribution strategies concerning security perils. In contrast, Identity-based Cryptography (IBC) presupposes the existence of a trusted authority for managing keys, with the public keys being derived from the identity of the device in question; hence, there is no need for pre-configuration. For confidentiality and integrity, TCP-based real-time protocols like Transport Layer Security (TLS) [135] and Datagram Transport Layer Security (DTLS) [136] utilize the encrypted channel of transfer. It is, however, very accurate that while TLS 1.3 is optimized for

some settings, DTLS is designed to work on unreliable datagram-based connections. Techniques like Diffie Hellman Key Exchange [137] and other essential agreement procedures allow devices to create keys for conversation without using essential reserve secrets. This is made possible through computational techniques to implement a secure channel. However, some shortcomings should be solved such as problems of complexity, scalability issues and efficient key management.

3.1 Introduction

Some of the challenges include time efficiency, accuracy, scalability, data novelty, and interpretability in IoT environments, which the proposed methodology overcomes as it includes advanced techniques in a single framework. Therefore, the suggested approach outlines a new security architecture that implements robust cryptographic methods, optimal key management, and secure communication methods. The proposed framework includes two key components from their previous versions: RB-BFT X and CoatiNet. RB-BFT X refines the improved BFT [138] by overcoming the drawbacks like poor scalability and high computational cost of the improved BFT; the modifications incorporate confidentiality, integrity, and efficiency to ensure that the network satisfies security requirements. CoatiNet enhances the Coati algorithm [139] by focusing on three main areas: optimization of agents in the communication paths, continuous reallocation of data according to feedback received, and improved security frameworks to overcome the limitations of the Coati algorithm, including the inefficient communication and static handling of data in IoT networks, lower latency, better network performance and security of the communication paths in healthcare IoT networks. Section 3.2 gives the proposed system model and a detailed design goal of the framework; section 3.3 presents the secured healthcare IoT Framework; section 3.4 explains experimental results and performance analysis; and the summary of the proposed work is given in section 3.5.

3.2 System Model

As shown in Fig. 3.1, the healthcare IoT network and its ecosystem are built based on a specific structure containing complex elements divided into several clusters based on their responsibilities essential to the system's performance. These clusters from the larger picture also consist of patients (PC_i), doctors (DC_j), receptionists (RC_k), and administrators (AC_l), and the house IoT devices that are relevant to each of them. PC_i incorporated systems include wearables in the form of health monitoring devices and medical terminals (D_{PC_i}), the operations of which are to capture and forward patient care data for further analysis and monitoring. In contrast, DC_j host devices (D_{DC_j}) provide access to patient records, diagnosis, and communication channels for health professionals. RC_k coordinate administrative work, appointment setting, and client relations through the IoT devices (D_{RC_k}), although AC_l supervise physical facilities, IoT systems and compliance with legal requirements through their devices (D_{AC_l}).

The basic structure of this network architecture is based upon a hierarchical scheme to enhance communication and management of the entire system. Main

hospital servers are primarily responsible for granting patient record information and managing the network systems. Then, the Cluster Heads (CH_i, CH_j, CH_k, CH_l) are appointed to regulate communication and coordinate within the determined areas of responsibility. This design can enhance data flow, IoT security, and network expansion since different groups are hierarchical.

Data flow in the healthcare IoT network is well planned and implemented to ensure the optimized transfer of data and the information transfer is secure. Within each cluster, the corresponding IoT devices are paired with their CH , which is the primary way of communication between devices of the same cluster. These CH serve as middlemen in conveying messages with the main hospital servers (MHS), where necessary, so that information flows between Clusters and servers. The communications cycle architecture of this method reduces latency as well as privacy and security barriers and allows for the easy sinking of IoT in the healthcare system.

3.2.1 Attacker Model

The proposed methodology covers the following different types of threats targeting healthcare IoT networks through the attacker model.

Passive Eavesdropping on Patient Clusters

A threat actor tries to eavesdrop on communication data between the IoT devices of two patients in the same cluster, and the Cluster Heads are responsible for managing several patient clusters to obtain personal medical information without modifying the original transmitted data. As a result of this process, there is a high tendency for data leakage to occur.

Active Intrusion and Tampering on Doctor Clusters

The adversary actively interferes in the working process of doctor clusters, sending them malignant messages or changing the medical records' content to damage the information's credibility and preserve it responsibly.

Insider Threats within Receptionist Clusters

A group of insiders who have legitimate access rights represent a threat within clusters of receptionists; if they choose to take advantage of existing loopholes, they entice others to participate in the Sybil attack or sell patient information that ought to be kept comfortable.

Denial of Service (DoS) Attacks on Admin Clusters

Adversaries utilize DoS aimed at admin clusters to overload resources hence leading to the deterioration of system administration and access for only those with legitimized permissions with additional concerns towards a worsened latency.

Brute Force Attacks

Brute force attacks primarily target admin clusters, as attackers attempt to repeatedly guess authentication credentials to gain unauthorized access to administrative controls. Compromising these access points can disrupt secure communication and compromise the confidentiality of sensitive system-wide data.

Distributed Denial of Service (DDoS) Attacks

DDoS attacks can affect all clusters, but they are particularly detrimental to admin clusters and doctor clusters. Admin clusters are critical for managing resources and overall system operations, while doctor clusters handle sensitive medical data. Overwhelming these clusters leads to widespread service interruptions, degraded performance, and inaccessibility of crucial systems.

Man-in-the-Middle (MITM) Attacks

MITM attacks are most likely to target patient clusters and doctor clusters, where attackers intercept and manipulate communication between devices. This compromises the confidentiality and authenticity of personal medical data and diagnosis records, posing serious risks to patient privacy and treatment accuracy.

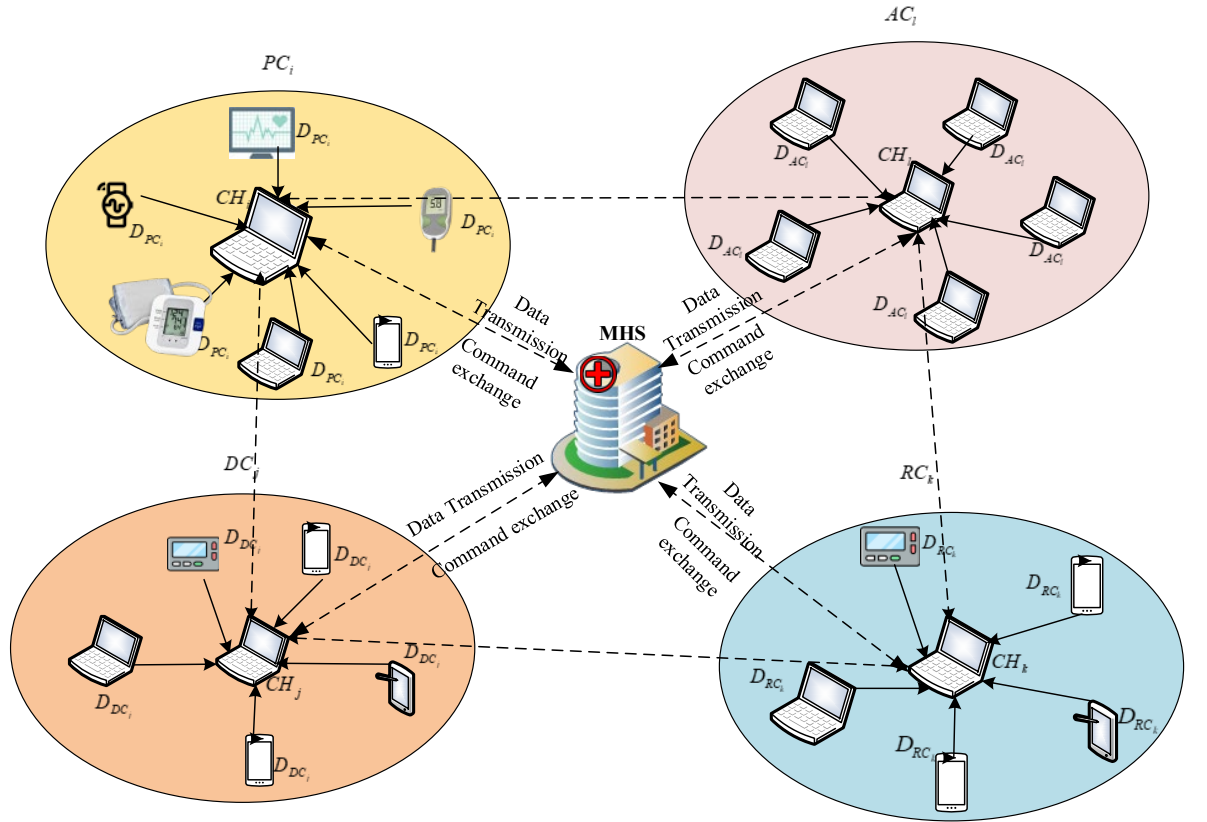


Figure 3.1: Healthcare IoT Network System Model

Data Falsification across All Clusters

Opponents interfere with the data in all the clusters and feed the target recipients with fake information. At the same time, the other relevant processes that need to be conducted for efficient healthcare management are influenced by and based on fabricated data. The danger of data leakage and contamination is amplified.

Sybil Attacks across the Network

Cyber adversaries use several fake identities/addresses to create false request or response flows, control the cluster operations, corrupt data, and cause service

interruption for healthcare systems.

3.2.2 Design Goals

The design goals that seek to integrate design requirements for IoT as a concept to advance the development of a comprehensive framework that is secure, cost-effective and compliant for healthcare IoT systems are as follows:

- ***Autonomous Clustering:*** Healthcare IoT systems need to be self-managing to accommodate a wide range of devices to provide quick responses to the patients as well as efficient utilization of capabilities.
- ***Secure Communication:*** For the privacy of patient information and especially because of the nature of such information, the security of such communication is paramount to prevent instances of invasion of privacy and unauthorized access to such details.
- ***Role-Based Access Control:*** Currently, the roles played in a healthcare organization entail the usability of data at a varying degree. The Role-based access control ensures that only those individuals with the authority can access particular information hence minimizing wrong uses as well as tampering.
- ***Anomaly Detection:*** The threats that healthcare networks encounter is not stagnant, which makes anomaly detection essential to the discovery of threats and possible prevention before occurring in the future and maintaining the security of the networks and their activities.
- ***Data Privacy Protocols:*** Privacy is a cardinal concept in all health practices worldwide when handling patients. Additional safeguards are needed to protect the privacy of the data by providing a way to ensure that patients' trust is not violated by having their information viewed or shared by people who have no business doing so.
- ***Optimal Routing:*** Since enhanced delivery of health care services is anchored on the shortest delivery of messages between devices and health care providers, proper routing algorithms must be in place to reduce time wastage.
- ***Multi-Layered Security:*** There are numerous potential dangers to Health Care IoT settings, as the sector is particularly susceptible to multiple varieties of cyber threats. Advanced measures ensure that the systems are protected from various incurrences while ensuring the delivery of healthcare to patients is not compromised.
- ***Network Resilience:*** Healthcare operations cannot be stopped and need to continue functioning despite the nature of the networks sharing the information. This means that the network has to be reliable to keep offering services with no compromise on patients' well-being.
- ***Regulatory Compliance:*** Healthcare regulatory mechanisms must be strictly adhered to because failure brings about legal consequences that infringe on the care of patients. Regulatory compliance is thus maintained by constant scrutiny and sanctioning so that

laws and rules guiding the provision of health services are complied with and trust is instilled in the sector.

3.3 Proposed Secured Healthcare IoT Framework

The proposed method provides a secure healthcare IoT network through a two-stage process, as illustrated in Fig. 3.2.

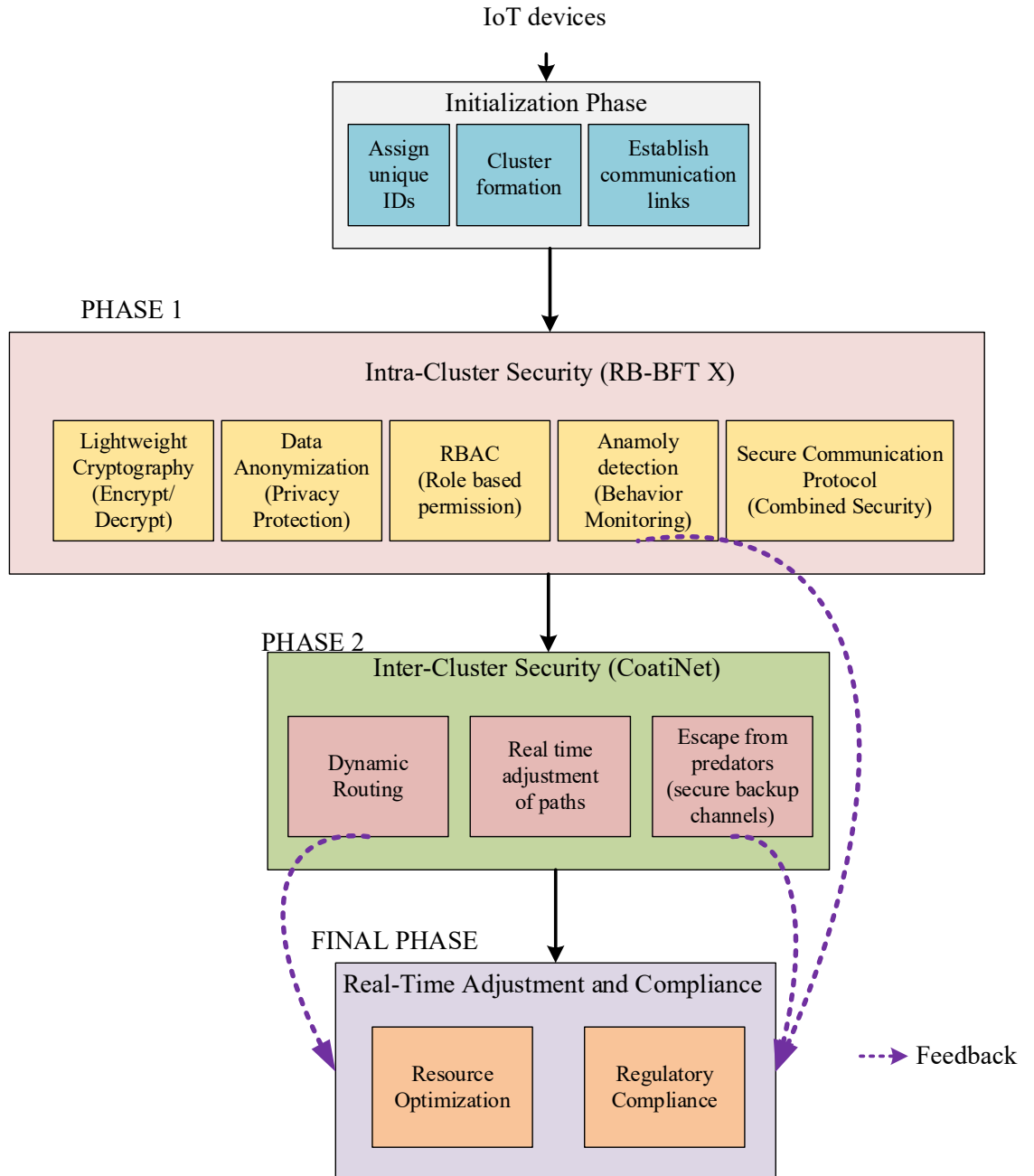


Figure 3.2: Proposed Multi-Cluster Security Framework

The first phase ensures intra-cluster security using RB-BFT X, which groups IoT devices into role-based clusters. This phase employs lightweight cryptography to

secure communication, Role-Based Access Control (RBAC) to restrict unauthorized access, and lightweight anomaly detection to identify potential threats. It begins with the initialization of IoT devices, assigning them unique IDs and locations, followed by dynamic cluster formation with the election of Cluster Heads (CHs) based on processing power and battery life. Within each cluster, secure communication is maintained while continuously monitoring device behaviors for anomalies. The second phase enhances inter-cluster and server communication using CoatiNet, inspired by coati behavior. CoatiNet incorporates dynamic routing algorithms for real-time path optimization, intrusion detection systems, redundancy protocols, and secure backup channels to ensure robust inter-cluster communication. It also provides adaptive self-recovery mechanisms that adjust to network conditions, reroute data, and reallocate resources as needed. Additionally, the methodology ensures regulatory compliance by continuously monitoring operations and optimizing resource utilization. This comprehensive process flow achieves multilevel security and efficiency in handling sensitive healthcare data, addressing both intra-cluster and inter-cluster vulnerabilities effectively.

3.3.1 Initialization

Starting from Phase I where the configuration of the healthcare IoT network is initiated and IoT devices and Cluster Heads are established. Initialization involves setting up the healthcare IoT network, identifying IoT devices, and configuring CH . Let's break down this process:

Let D be the total number of IoT devices in the network, C be the number of CH . Each IoT device $d \in D$ is uniquely identified by an ID d_{id} , where $1 \leq d_{id} \leq D$ and each Cluster Head $ch \in CH$ is uniquely identified by an ID ch_{id} , where $1 \leq ch_{id} \leq C$. Each $d \in D$ is assigned a precise physical location within the healthcare facility, denoted by coordinates (x_d, y_d) on the floor plan. This location is crucial for establishing proximity-based communication and determining the device's spatial relationship with other network entities. The network topology setup involves establishing communication links between d and ch within the healthcare facility. This setup aims to optimize communication efficiency and minimize energy consumption by organizing IoT devices into clusters based on their proximity to Cluster Heads.

Mathematically, we represent the network topology as a graph

$$G = (V, E) \quad (3.1)$$

where V and E are the set of vertices and edges representing d as well as ch . The connectivity between IoT devices and Cluster Heads is defined by a connectivity matrix A , where $A_{rs} = 1$ if there is a communication link between the device named d_r and Cluster Head ch_s which is related to the device d_s and $A_{rs} = 0$ otherwise. The r and s refers to the two clusters that we have assumed, d_r and d_s are the IoT devices within those clusters. Once all d and ch are identified and configured, then the

network topology is established, the initialization phase is complete. The network is now ready for subsequent processes.

Algorithm 3.1: The Proposed Secure Healthcare IoT Framework
<p>Input:</p> <p>$D \leftarrow$ Set of IoT devices $\{d1, d2, ..., dn\}$</p> <p>$CH \leftarrow$ Set of Cluster Heads $\{ch1, ch2, ..., chm\}$</p> <p>Parameters: Processing power (PP), Battery life (BL), Location (x, y)</p> <p>Protocols: MQTT, CoAP, HTTP</p>
<p># Phase 1: Intra-Cluster Security</p> <p>Initialize Network:</p> <p>For each device $d \in D$:</p> <p> Assign unique ID and physical location (x, y)</p> <p> Establish communication links with candidate CHs based on proximity</p> <p>Cluster Formation:</p> <p>For each device $dr \in D$:</p> <p> For each candidate CHs $\in CH$:</p> <p> $Score(chs) = wpp * PP + wbl * BL + wcentrality * Centrality(chs)$</p> <p> Select CH with the highest $Score(chs)$</p> <p> If CH capacity is sufficient:</p> <p> Add dr to CH's member list</p> <p> Else:</p> <p> Select next highest scoring CH</p> <p>Intra-Cluster Communication:</p> <p>For each communication within a cluster:</p> <p> Encrypt data using lightweight cryptography</p> <p> Monitor device behavior Xr</p> <p> If $Xr - \mu > \theta * \sigma$:</p> <p> Trigger anomaly alert and notify CH</p>

Apply RBAC:

If role r has permission for operation s :

Grant access

Else:

Deny access

Phase 2: Inter-Cluster Communication

Dynamic Routing:

For each CH pair (CH_i, CH_j):

Compute latency L_{ij} and security metric S_{ij}

Optimize routing table $R(t)$:

$R(t) = \text{argmin}(L_{ij} + S_{ij})$

Real-Time Adjustments:

Monitor network conditions:

If congestion or insecure path detected:

Identify alternate path with lower latency and higher security

Update routing variables dynamically

Backup and Redundancy:

For each communication path p :

If intrusion detected ($\text{Detect}(a) = 1$):

$\text{Reroute}(p) = \text{Backup}(p)$

Else:

$\text{Reroute}(p) = p$

Escape from Predators (Security Mechanisms):

For each user/device u accessing resource/data d :

If $\text{Auth}(u, d) = \text{True}$:

Grant access

Else:

<p>Deny access</p> <p>Monitor inter-cluster communication with IDS:</p> <p>If Detect(a) = True:</p> <p> Trigger alert and initiate response</p> <p># Network Resilience and Compliance</p> <p>Adaptive Self-Recovery:</p> <p> Adjust configuration dynamically:</p> <p> $C(t+1) = C(t) + \alpha * F(t)$</p> <p> Optimize resource utilization:</p> <p> Minimize $\sum R_u - \sum r_{ui}$</p> <p>Regulatory Compliance:</p> <p> For each operation O:</p> <p> If O complies with regulations:</p> <p> Allow operation</p> <p> Else:</p> <p> Block operation</p> <p>Output:</p> <p> Secure intra-cluster and inter-cluster communication</p> <p> Optimized routing, adaptive self-recovery</p>

The proposed algorithm 3.1 outlines a secure multi-cluster framework for healthcare IoT networks, consisting of two key phases. In Phase 1, intra-cluster security is achieved by organizing IoT devices into role-based clusters, using lightweight cryptography, anomaly detection, and Role-Based Access Control to ensure secure communication and data privacy within each cluster. In Phase 2, inter-cluster communication is optimized through dynamic routing, adaptive self-recovery, and multi-layered security measures, inspired by Coati behavior, to maintain robust and efficient data transmission across clusters. The algorithm also includes real-time adjustments and backup mechanisms to respond to network disruptions and security threats, ensuring the resilience and compliance of the healthcare IoT network.

3.3.2 Autonomous Cluster Formation

After the initialization step, where are identified and configured, d and ch the cluster formation process begins. This involves organizing D into clusters based on their roles, responsibilities and proximity to CH . The cluster formation process aims to optimize communication efficiency, resource utilization and management within the healthcare IoT network.

Unlike traditional methods with pre-designated CH s, autonomous cluster formation [140] empowers devices to elect suitable CH based on the trust worthiness. This election leverages an initial advertising phase where devices broadcast their capabilities, such as processing power, communication range, and battery life, to nearby devices. Each device d_r calculates a scoring function for each potential CH d_s within its communication range, denoted as $C_{candidate}$.

The score $Score(d_s)$ for each candidate CH d_s is computed using a weighted sum of factors like processing power PP_s , battery life BL_s and centrality within the network $Centrality(d_s)$:

$$Score(d_s) = w_{pp} \cdot PP_s + w_{BL} \cdot BL_s + w_{Centrality} \cdot Centrality(d_s) \quad (3.2)$$

where $w_{pp}, w_{BL}, w_{Centrality}$ are pre-defined weights reflecting the relative importance of each factor in CH selection. The device with the highest score among candidates of CH becomes the CH for that cluster:

$$CH_r = \arg \max(Score(d_s)) \quad \forall d_s \in C_{Candidate} \quad (3.3)$$

where $\arg \max$ is an operator that returns the argument (in this case, the device d_s) that gives the maximum value of the function $score(d_s)$. This scoring approach ensures that devices with superior processing capabilities, longer battery life and a central location within the candidate pool are more likely to be elected as CHs.

Following CH selection, each device d_r calculates its distance d_{rs} to nearby CH using a distance metric named as Euclidean distance and transmits a join request to the closest CH , denoted by CH_s . The CH maintains a list of member devices within its communication range, represented by $Member_List_{CH_s}$.

To ensure efficient cluster formation and avoid overloading CH , a capacity check is implemented. The CH verifies if it has sufficient resources (processing power, communication bandwidth) to accommodate the joining device. If d_s falls within the communication range of CH_s and the CH 's capacity allows for inclusion, d_s is added to the $Member_List_{CH_s}$.

This process is mathematically represented as follows:

$$Member_List_{CH_s} = Member_List_{CH_s} \cup \{d_s\} \quad (3.4)$$

This is achieved if ($d_{rs} \leq Communication_range_{CH_s}$) and ($|Member_List_{CH_s}| < Capacity_{CH_s}$). This condition ensures that device d_r joins the cluster managed by CH_s if it is within the communication range and the cluster head has enough capacity to accommodate additional devices.

3.3.3 Two-phase Approach

The proposed methodology comprises two phases: Phase 1 focuses on safeguarding nodes to cluster head communication using RB-BFT X, ensuring secure intra-cluster operations. CoatiNet, as the second phase of the approach in Coati, focuses on optimizing inter-cluster communication, increasing the level of latency, security, and network adaptability in healthcare IoT systems.

a) Phase 1: Safeguarding Nodes to Cluster Head Communication

The proposed methodology employs Redundant Byzantine Fault Tolerance with Extensions (RB-BFT X) to address security challenges in healthcare IoT networks. RB-BFT X was chosen for its scalability, adaptability and lightweight cryptographic techniques, which make it suitable for resource-constrained IoT environments. Unlike traditional fault-tolerance algorithms such as Practical Byzantine Fault Tolerance (PBFT) or Federated Byzantine Agreement (FBA), RB-BFT X incorporates features like behavioral anomaly detection, role-based access control and data anonymization. These features enable it to provide a comprehensive and efficient security solution tailored for dynamic and distributed healthcare IoT networks. PBFT, while robust, suffers from high communication overhead, making it less ideal for large IoT networks. Similarly, FBA is more suited for permissioned environments with pre-established trust, limiting its applicability to decentralized healthcare IoT systems. RB-BFT X addresses these limitations by enhancing scalability, reducing computational complexity, and integrating domain-specific security measures.

RB-BFT X extends the basic Byzantine Fault Tolerance algorithm by incorporating redundancy mechanisms that improve resilience against malicious or arbitrary node behaviors. Its lightweight cryptography ensures that IoT devices with limited processing power can maintain data integrity and confidentiality without significant computational overhead. The inclusion of real-time anomaly detection and dynamic adaptation makes RB-BFT X highly effective for IoT environments where security threats evolve rapidly. The first phase is described below.

i) Lightweight Cryptography

RB-BFT X accomplishes this by utilizing small amounts of computation and lightweight cryptographic algorithms [141] to protect the cluster's communication links. This makes it possible to preserve private information and data authenticity in silence without overburdening contained IoT devices with computational complexity.

Let K_{pub} be the public key and K_{priv} be the private key generated for each d and ch . Encryption E and decryption D procedures may also be described as

$$c = E(m, K_{pub}) \quad (3.5)$$

$$m = D(c, K_{priv}) \quad (3.6)$$

where m is the plaintext message and c is the ciphertext.

ii) Data Anonymization

As for data privacy, data anonymization remains one of the key constituents of the RB-BFT X framework applied to safeguard the protection of sensitive health information exchanged periodically through IoT networks. The use of high levels of anonymity guarantees that the last acknowledged attributes of an individual shall not be subjected to identification attacks and other forms of unauthorized access. Some steps of anonymization apply randomization techniques based on clustering k-anonymity models [142], that is, grouping K records in clusters where it is hard to distinguish individuals from the released data. In this process, however, mathematical derivations are relevant to prove the degree of effectiveness of the particular anonymization method used. For example, let R be the set of the data records, and QI is the other part of the record, which is often called quasi-identifiers. The steps involved in this process include the partition operation, which can be done by clustering the data R , with each cluster $DC_1, DC_2, DC_3, DC_4, \dots, DC_n$ providing the k-anonymity. This can be mathematically described as $\forall r, |C_i| \geq K$, where it is important to observe that every must include at least records. Furthermore, given a particular set of fields, which can be considered as sensitive in each record, the studied anonymization processes are to generalize these fields to ensure data privacy, although data usefulness is also a crucial factor. Using symbols, generalized quasi-identifiers can be denoted as GQI , while generalized sensitive attributes (SA) after anonymization are represented as GSA which is a constant that represents a predetermined level of sensitivity of the system for the selection of anomalies. It points to the fact that the criterion for detecting anomalies is more rigorous when the value is higher. The mathematical formulation involves defining appropriate generalization hierarchies and algorithms to balance privacy protection and data usability. By integrating such mathematical derivations into the RB-BFT X framework, robust data anonymization mechanisms can be established, ensuring secure and privacy-preserving healthcare data collection and transmission within IoT networks.

iii) Role-Based Access Control (RBAC)

RBAC enforces access restrictions based on the specific roles of devices and users within the cluster, enhancing security and maintaining the confidentiality and integrity of sensitive healthcare information. The Access Control Matrix (ACM) defines permissions for each role within the cluster.

The Access Control Matrix is represented as

$$ACM_{rs} = \begin{cases} 1 & \text{if roler has permission for operation } s \\ 0 & \text{otherwise} \end{cases} \quad (3.7)$$

where $ACM_{rs} = 1$ if role r has permission for operation s , $ACM_{rs} = 0$ otherwise, n is the number of IoT devices and m is the number of operations.

Consider the following roles and operations:

Roles: Patient (R1), Doctor (R2), Receptionist (R3), Admin (R4)

Operations: Read Patient Data (O1), Write Patient Data (O2), Schedule Appointment (O3), Access Admin Functions (O4)

The ACM might look like this:

$$ACM = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \quad (3.8)$$

In this matrix; Patients (R1) can only read their own data (O1), Doctors (R2) can read (O1) and write (O2) patient data, Receptionists (R3) can schedule appointments (O3), Admins (R4) have full access to all operations (O1, O2, O3, O4).

The enforcement of RBAC can be mathematically modeled as:

$$Permission\ granted = \begin{cases} 1 & \text{if } ACM_{rs} = 1 \\ 0 & \text{if } ACM_{rs} = 0 \end{cases} \quad (3.9)$$

where r is the role of the device requesting access, s is the operation or resource being requested.

iv) Behavioral Anomaly Detection

Behavioral anomaly detection continuously monitors the activities and behaviors of IoT devices to detect deviations that could indicate security threats. The anomaly detection is based on statistical methods.

A baseline of normal behavior is established for each type of IoT device based on historical data. This baseline is characterized by the mean (μ) and standard deviation (σ) of observed behaviors. As the IoT devices operate, their real-time behaviors (X_r) are monitored and compared against the established baseline. Any significant deviation from the norm could indicate an anomaly.

The mathematical model used for detecting anomalies is based on statistical methods. The model checks whether the observed behavior (X_r) deviates from the mean (μ) by a certain threshold (θ) times σ . The anomaly detection can be expressed as:

$$Anamoly = \begin{cases} 1 & \text{if } |X_r - \mu| > \theta\sigma \\ 0 & \text{otherwise} \end{cases} \quad (3.10)$$

where θ is a predefined threshold that determines the sensitivity of the anomaly detection system. A higher θ value implies a stricter criterion for anomaly detection.

When an anomaly is detected (i.e., when $Anamoly = 1$), the Cluster Head starts a response mechanism. This response may involve notifying the network administrators, quarantining the identified device to avoid further harm possibly in terms of damaging or corrupting other devices and subsystems within the network and carrying ahead with a series of tests to identify the root cause of the anomalous behavior.

v) Secure Communication Protocols

To guarantee secure communication within the cluster, the data transmission must be confidential, integrity and authentic; the RB-BFT X framework integrates robust secure communication protocols. The mechanisms such as cryptography, digital signatures, and hashing algorithms are essential in ensuring security when data is transferred through IoT devices and to the CH. Security, which is the basis of communicating confidentiality, employs encryption as a tool to encode data before it is transmitted. It is unreachable to anybody and is not intended, as seen in the equation. This process of transmission allows, even if the data transmitted also goes through some other taps, the data to remain safe and secure.

The process of decrypting, which is virtually the opposite activity of encrypting, occurs at the receiving end to obtain the actual plaintext message as in the equation. To be able to decipher the message, the private key corresponding to the recipient or the developer is used to decrypt the data.

This gives rise to digital signatures, which help to ensure that the transmitted data is both authentic and of integrity. This does not only allow the receiver to ensure the information passed is accurate but also the sender's credentials. The sender then sends this message m and signs it with their private key K_{priv} to come up with a digital signature.

$$signature = sign(m, K_{priv}) \quad (3.11)$$

The recipient verifies the signature using the sender's public key K_{pub} . Successful verification confirms the message's authenticity and integrity, ensuring it was sent by the sender without tampering:

$$verify(signature, m, K_{pub}) \quad (3.12)$$

To maintain data integrity, hash functions generate fixed-size hash values from the original message. Any message alteration yields a distinct hash value, facilitating easy detection. The hash value for m is computed using a hash function H .

$$Hash = H(m) \quad (3.13)$$

Algorithm 3.2: RB-BFT X

<p>Input: d : IoT devices within the cluster</p>
--

<p>CH : Cluster Head</p>

<p>s: Operations</p>

<p>r: Roles</p>

<p>m: Plaintext message</p>
--

<p>c: Ciphertext</p>

<p>Output: Secure healthcare IoT network with intra-cluster security</p>

<p># Initialization</p>

<p>for device in d:</p>

<p style="padding-left: 40px;">$K_{pub}[device], K_{priv}[device] = generate_keys()$</p>
--

<p>$ACM = \{\}$</p>

<p>for role in r:</p>

<p style="padding-left: 40px;">for operation in s:</p>
--

<p style="padding-left: 80px;">$ACM[(role, operation)] = define_access_control_matrix(role, operation)$</p>

<p>$baseline_behavior = \{\}$</p>

<p>for device_type in $DEVICE_TYPES$:</p>
--

<p style="padding-left: 40px;">$baseline_behavior[device_type] = establish_baseline_behavior()$</p>
--

<p>$epsilon = define_anomaly_detection_threshold()$</p>

```

# Data Anonymization
R = get_dataset()
for i in range(len(CLUSTERS)):
    DC_i = CLUSTERS[i]
    partitioned_dataset = partition_dataset(R, DC_i)
    if len(partitioned_dataset) >= K:
        DC_i = partitioned_dataset

generalize_QI_and_SA(R)

# Lightweight Cryptography
c = encrypt(m, Kpub[CH])
m = decrypt(c, Kpriv[d])

# Role-Based Access Control (RBAC)
for role in r:
    for operation in s:
        if ACM[(role, operation)] == 1:
            permission_granted = 1
        else:
            permission_granted = 0

# Behavioral Anomaly Detection
for device in d:
    behavior = get_behavior(device)
    if abs(behavior - baseline_behavior[DEVICE_TYPE(device)]) > epsilon *
baseline_behavior[DEVICE_TYPE(device)]:
        anomaly = 1
    else:
        anomaly = 0

```

```

# Secure Communication Protocols
signature = sign(m, Kpriv[d])
verify_signature = verify(m, signature, Kpub[d])
hash_result = hash(m)

# Response
if anomaly == 1:
    alert_administrators()
    isolate_device(d)
    initiate_diagnostics()
end if

```

Thus, the RB-BFT X algorithm (algorithm 3.2) is chosen for its ability to address the limitations of traditional fault tolerance algorithms like Practical Byzantine Fault Tolerance (PBFT) and Federated Byzantine Agreement (FBA). PBFT's quadratic communication overhead and FBA's reliance on quorum slices hinder scalability and adaptability, especially in resource-constrained environments like healthcare IoT systems. RB-BFT X overcomes these challenges by integrating lightweight cryptographic techniques to reduce computational overhead and role-based access control to enforce strict data policies. It also incorporates behavioral anomaly detection to identify and mitigate malicious activities in real time and employs secure communication protocols to ensure data integrity and confidentiality. These enhancements make RB-BFT X resource-efficient, scalable, and robust against adversarial attacks, providing a holistic solution for secure and reliable intra-cluster communication. This makes it an optimal choice for addressing the unique security challenges of healthcare IoT systems.

b) Phase 2: Inter-Cluster and Server Communication Optimization using CoatiNet

Phase 2 of the proposed methodology, known as "Inter-Cluster and Server Communication Optimization" using CoatiNet, represents a significant advancement in healthcare IoT network management.

CoatiNet Module

Drawing inspiration from the coordinated behavior of coatis, CoatiNet introduces a dynamic, resilient and secure communication system that addresses the unique challenges of IoT environments. Its mechanisms aim to optimize data transmission paths, improve latency, and bolster network security against potential threats while adapting to evolving conditions.

At its core, CoatiNet employs dynamic routing algorithms to optimize communication paths continuously. These algorithms adjust routes based on real-time network feedback, ensuring minimal latency and robust security. CoatiNet also integrates adaptive configuration mechanisms, enabling the system to reallocate resources and reroute data when faced with congestion or security threats. Its multi-layered security approach leverages intrusion detection systems, redundancy protocols, and secure backup channels to safeguard communication even under adversarial conditions. Furthermore, CoatiNet mimics coatis' ability to evade predators by introducing a proactive threat response mechanism, which includes path recalibration and enhanced authentication. By drawing inspiration from coati behaviors, CoatiNet introduces several advantages over traditional inter-cluster communication frameworks. Its dynamic routing algorithms prevent attackers from exploiting predictable paths, while its adaptive mechanisms enhance network resilience by redistributing resources during disruptions. The module's layered security ensures data integrity and minimizes vulnerabilities, making it particularly effective for healthcare IoT environments. Table 3.1 provides a conceptual mapping of coati behaviors to the key functionalities of the CoatiNet module, illustrating how the natural adaptability, cooperation, and resilience of coatis inspire its design.

Table 3.1: Relation between Coati behavior and CoatiNet module

Coati Characteristic	Coati Behavior Explanation	CoatiNet Module Implementation
Cooperative Behavior	Coatis exhibit cooperative behavior within their groups, working together to forage, protect against predators, and care for young.	Collaborative optimization of communication paths between IoT devices and clusters, mirroring the cooperative behavior seen in coatis.
Adaptability	Coatis are highly adaptable animals, able to adjust their behavior and habitat to changing environmental conditions.	Dynamic rerouting of data and adjustment of network configurations based on real-time feedback, reflecting the adaptability of coatis to changing circumstances.
Dynamic Routing	Coatis frequently change their routes while foraging to optimize food collection and avoid predators.	Employment of dynamic routing algorithms to continuously adjust communication paths between clusters, akin to the dynamic routing behavior observed in coatis.
Collaboration	Coatis collaborate within their groups to defend against predators, detect threats, and coordinate activities.	Collaboration with intrusion detection systems, authentication mechanisms, and redundancy protocols to defend against attacks and

		ensure uninterrupted communication, resembling the collaborative behavior seen in coatis.
Resilience	Coatis demonstrate resilience to environmental changes, adapting their behavior and strategies to survive in various conditions.	Prioritization of network resilience through redundancy mechanisms, backup channels, and adaptive configurations to withstand security breaches, network disruptions, and regulatory changes, similar to the resilience displayed by coatis in challenging environments.

i) Encryption of Data

To prevent passive eavesdropping, all data packets transmitted between IoT devices and Cluster Heads (CHs) are encrypted using a robust encryption algorithm like AES_256 [143]. The encryption process ensures that even if an attacker intercepts the communication, they cannot access the sensitive information without the decryption key. Mathematically, for a data packet $data$ and an encryption key key , the encryption function $E(data)$ is represented as:

$$c = E(data) = AES_256(data, key) \quad (3.14)$$

This ensures confidentiality of the transmitted data.

ii) Dynamic Routing

Dynamic routing algorithms are employed to avoid predictable communication paths that attackers could target. These algorithms often alter the data communication paths, and antagonists can hardly track and constantly interrupt the data transfer process.

To formalize the optimization of routing in the network, consider the following variables:

N is the number of IoT devices within each cluster, M is the number of clusters in the network, $D(i, j)$ is the distance matrix of the cluster-to-cluster separation i.e. between cluster C_i and C_j , $R(t)$ is the routing table at time t which shows the current routes for data transfer.

This is a formal optimization problem that seeks to achieve a minimal total communication delay while guaranteeing the security of the links chosen as optimal.

The latency L is modeled as:

$$L = \sum_{i=1}^N \sum_{j=1}^M D(i, j) \times f_{ij}(t) \quad (3.15)$$

where $f_{ij}(t)$ is a binary function that indicates whether data is being transmitted between clusters i and j at time t . It takes the value 1 if data transmission is occurring and 0 otherwise.

One of the main directions of the algorithm and the focus on the dynamic routing aspect of the algorithm is the process of the change of the value of the routing variables $f_{ij}(t)$ that are associated with the threats and changes in the network conditions. This updating process makes it possible to ensure that the algorithm changes as the network environment and does not constantly take the same paths because they can be the paths that the attackers can easily notice.

Dynamic Update Mechanism

Essentially, using the probabilities of variation in these conditions and threats, the CoatiNet algorithm alters the path routing variables $f_{ij}(t)$ to reroute data through other paths. Let $f_{ij}^*(t)$ be the updated routing variable between clusters i and j at time t . This updated variable is established on the current network conditions together with security precaution measures.

The new routing variable can be calculated by navigating through an optimization procedure that takes into account the lowest latency while providing best security. This optimization problem can be formulated as follows:

$$f_{ij}^*(t) = \arg \min L_{ij} + S_{ij} \quad (3.16)$$

where L_{ij} represents the latency along the path between clusters i and j , S_{ij} represents a security metric that quantifies the level of security along the path. This metric may include the probability of attack, presence of encryption, and so on and may relate to the issue of trust in the devices being used.

It works according to an iterative approach, which analyses numerous possible paths between clusters i and j , and then chooses the path that will have least overall latency and security measure $L_{ij} + S_{ij}$.

iii) Real-time Adjustments

The algorithm checks whether the existing path from the clusters i and j is the best one to use to deliver packets or if there is a better path to be used to reduce the time taken, or increase security.

This evaluation involves considering factors such as:

- **Network Congestion:** They also said it could select another route in the event that the current one is congested or experiences high traffic turnover, in order to achieve lower latency.
- **Security Threats:** It could also add additional paths between two particular computers if the current set path is considered insecure with respect to likely attacks or malicious devices.
- **Device Availability:** It takes into account the availability of devices on one or other channels or on any other possible route. This is because there is no guarantee that these particular devices are available or connected in the first place, thus keeping the data away from them where it is not needed.

In another manner, the CoatiNet algorithm considers fresh routing $f_{ij}(t)$ variables, which capture up the real-time network conditions and threats by adjusting the data transmission routes to minimize latency while ensuring the security of the healthcare IoT network.

iv) **Escape from Predators**

The "Escape from Predators" approach is focused on improving the protection of the healthcare IoT network by introducing effective preventive measures to protect it against different kinds of threats, including active viruses, intruders, tampering, insiders, and Denial of Service (DoS) attacks. Its components include intrusion detection system, reliable backup channels, and multi-factor authentication and authorization.

Multi-Layered Authentication and Authorization

This module provides a layered and holistic identification system of users and physical devices connecting to the IoT network. The authentication and authorization procedures provide restricted access to essential resources. The mathematical model for authentication and authorization can be illustrated as follows:

Let u represent a user or device and d represent a resource or data:

$$Auth(u, d) = \begin{cases} true & \text{if } u \text{ is authenticated for } d \\ false & \text{otherwise} \end{cases} \quad (3.17)$$

Intrusion Detection System (IDS)

An IDS tracks all the traffic going through the network and determines whether there is any malicious activity or deviation from the intrusion detection policy. When the IDS discovers an intrusion, it launches alerts and responses to counter the threat as per the policy.

The mathematical model for intrusion detection can be represented as follows:

Let a represent an anomaly detected by the IDS:

$$Detect(a) = \begin{cases} true & \text{if } a \text{ is detected} \\ false & \text{otherwise} \end{cases} \quad (3.18)$$

Redundancy and Backup Channels

It is expected to have multiple communication paths and redundant channels for backup communication in case of IoT network disruption. In cases of anomalous activities, the IoT network can switch to other means of passing the data between the two facilities. The mathematical representation for rerouting data can be represented as follows:

Let p represent a communication path:

$$Reroute(p) = \begin{cases} B(p) & \text{if } Detect(a) \text{ is true} \\ p & \text{otherwise} \end{cases} \quad (3.19)$$

This approach helps to safeguard the healthcare IoT network against various security threats by integrating authentication, intrusion detection, and redundancy measures in the system.

v) Adapting to Change: Network Resilience and Regulatory Compliance

Adapting to change makes the healthcare IoT network resilient, uses resources effectively, and complies with the regulatory authority. This approach proposes optimizing resource utilization, adapting configurations, and regulatory compliance.

Resource Utilization Optimization (RUO)

The objective is to minimize the difference between the allocated resources (R_u) and the actual resource utilization ($\sum_{i \in A} r_{ui}$) for each resource u in the network. This is expressed as a minimization problem:

$$RUO = \min_{u \in U} \sum_{i \in A} (R_u - r_{ui}) \quad (3.20)$$

where U represents the set of resources, A represents the set of IoT devices or applications, R_u represents the allocated resources and r_{ui} represents the resource utilization by application i .

Adaptive Configuration

The network configuration ($C(t)$) is adjusted at each time step based on real-time feedback ($F(t)$) and an adaptation rate (α).

The new configuration at the time $t + 1$ is calculated as follows:

$$C(t+1) = C(t) + \alpha.F(t) \quad (3.21)$$

Regulatory Compliance

The compliance of an operation (O) with regulatory standards ($Re\ g$) is determined using a compliance function. If the operation complies with the regulations, the function returns true; otherwise, it returns false:

$$Compliance(O) = \begin{cases} true & \text{if } O \text{ complies with } Re\ g \\ false & \text{Otherwise} \end{cases} \quad (3.22)$$

By implementing these strategies, the healthcare IoT network can dynamically adapt to changing conditions, optimize resource utilization, and ensure compliance with regulatory requirements. The mathematical models provide a formal framework for achieving these objectives, allowing for efficient and effective network management. The CoatiNet algorithm is given below as Algorithm 3.3.

Algorithm 3.3: CoatiNet
Inputs: <ul style="list-style-type: none"> - Number of IoT devices (N) within each cluster - Number of clusters (M) in the network - Distance between clusters C_i and C_j: $D(i, j)$ - Initial routing table $R(t)$ - Network conditions and security factors Output: <ul style="list-style-type: none"> - Secure and optimized data transmission between IoT devices and Cluster Heads - Resilient network with adaptable configuration and regulatory compliance
Initialization: <p>Initialize the routing variables: $f(t)_{ij} = 0$ for all i, j</p> <p>Encryption and Secure Communication:</p> <p>For each data packet to be transmitted:</p> <p>$c = E_AES_256(\text{data}, \text{key})$</p> <p>Transmit the encrypted packet c between IoT devices and Cluster Heads (CHs)</p>

Dynamic Routing Optimization:

For each time step t:

FOR i = 1 to M **DO**

FOR j = 1 to M **DO**

 Compute latency L_{ij} and security metric S_{ij} for the path between C_i and C_j

$f(t)*ij = \arg \min (L_{ij} + S_{ij})$

 Update the routing table $R(t)$ with the new routing variables $f(t)*ij$

END FOR

END FOR

Real-time Adjustments:

1. Monitor network conditions and security threats

2. **FOR** i = 1 to M **DO**

FOR j = 1 to M **DO**

IF network congestion OR security threats on the current path (i, j) **THEN**

 Identify alternative paths that minimize latency and maximize security

 Update the routing variables $f(t)*ij$ accordingly

END IF

IF devices along the current path (i, j) are unavailable **THEN**

 Reroute the data through alternative available paths

 Update the routing variables $f(t)*ij$

END IF

END FOR

END FOR

Escape from Predators (Security Mechanisms):

Implement multi-layered authentication and authorization:

FOR each user/device u and resource/data d **DO**

IF Auth(u, d) **THEN**

 Grant access

ELSE

Deny access

END IF

END FOR

Detect intrusions and anomalies using the IDS:

IF Detect(a) **THEN**

Trigger alerts and initiate response mechanisms

END IF

Reroute data through redundant backup channels:

IF Detect(a) **THEN**

Reroute(p) = Backup(p)

ELSE

Reroute(p) = p

END IF

Adapting to Change (Network Resilience and Compliance):

Optimize resource utilization:

FOR each resource u and application i

DO

Minimize $|R_u - \sum_{i \in A} r_{ui}|$

END FOR

1. Adapt network configuration:

$C(t+1) = C(t) + \alpha * F(t)$

2. Monitor regulatory compliance:

FOR each operation O **DO**

IF Compliance(O) **THEN**

Allow the operation

ELSE

<p>Block the operation</p> <p>END IF</p> <p>END FOR</p>

3.4 Experimental Methodology and Performance Analysis

This section describes the procedure detailing the main experiments, performance measurements, and analysis before implementing the methodology to substantiate its efficacy in defending IoT networks against adversarial activities.

3.4.1 Experimental Setup

The experimental setup is a practical framework within which we are to employ the proposed methodology; we strive to design it so that it would closely emulate the working conditions yet remain manageable, scientifically sound and easily replicable. This setup would incorporate both the hardware and software systems with Python as the preferred language of coding for the supposed methodology. For the hardware components, we use a standard computer system characterized by advanced computational power, adequate memory, and requisite networking devices, including routers and switches, to offer an actual network platform.

Python is used as the main programming language to implement the identified methodology due to its rich repository of libraries like TensorFlow and PyTorch for performing machine learning tasks. The development and testing of the methodology are carried out using PyCharm, an integrated development environment (IDE) that provides robust tools for code development, debugging, and execution. We evaluate and analyze the system perspective through performance analysis based on accuracy, precision, recall rates, time for each request (latency), and utilization of resources. After testing out a methodology, we examine the outcome to assess the effectiveness, efficiency, or otherwise of such methodology. This setup includes IoT devices and cluster heads strategically positioned with unique computational capacities, networking standards, and power supplies, as illustrated by Table 3.2. IoT devices have 1 GHz processors, 512 MB RAM, and battery-solar power, whereas the heads contain 2 GHz processors, 2 GB RAM, 1 TB hard disk, mains power with an available generator, and backup. These points make the proposed methodology analysis systematic, helping to capture and evaluate the effectiveness of tackling the issues of securing IoT networks.

Various malicious attacks targeting healthcare IoT networks are simulated to evaluate the system's resilience. Passive Eavesdropping is emulated using Scapy to intercept and capture packets between patient IoT devices. Active Intrusion and Tampering on doctor clusters is simulated by generating forged traffic with Scapy to manipulate medical records. Insider Threats within receptionist clusters are modeled

by simulating unauthorized access with Python scripts. Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are generated using Hping3 and LOIC for resource overload. Brute Force Attacks on admin clusters are simulated with custom Python scripts leveraging Paramiko. Finally, Man-in-the-Middle (MITM) attacks targeting patient and doctor clusters are carried out using Scapy for packet sniffing and injection. These simulations, developed and executed in PyCharm, provide insights into the system's vulnerability to various attacks and its ability to detect and respond to them.

Table 3.2: Parameter settings

Parameter	IoT Devices	Cluster Heads
Physical Location	(10,20), (15,25), (8,18)	(50,60), (55,65), (48,58)
Processing Capabilities	CPU speed: 1 GHz, Memory: 512 MB	CPU speed: 2 GHz, Memory: 2 GB
Communication Capabilities	Wi-Fi, Bluetooth, Zigbee	Wi-Fi, Bluetooth, Zigbee
Initial Power Level	Battery: 3000 mAh, Power source: Solar	Power source: Mains, Backup: Generator
Additional Parameters	-	Storage capacity: 1 TB
Processing Capabilities	CPU speed: 1 GHz, Memory: 512 MB	CPU speed: 2 GHz, Memory: 2 GB
Security Features	Encryption: AES-256, Firewall: Enabled	Encryption: AES-256, Firewall: Enabled
Data Retention Policy	Retention Period: 30 days, Data Backup: Weekly	Retention Period: 90 days, Data Backup: Daily
Network Protocol	MQTT, CoAP, HTTP	MQTT, CoAP, HTTP

3.4.2 Evaluation Metrics

In the course of the experimental analysis of the presented methodology, several measures of effectiveness are used to make the evaluation. These metrics have given quantitative values so that the performance can be quantitatively measured according to the different aspects, including accuracy, precision, recall, F-score, lifetime of the network, throughput of the network, rate of detection, false positive rate, and the overall system performance.

i) Accuracy

$$Accuracy = \frac{\text{Number of correctly classified instances}}{\text{Total number of instances}} \quad (3.23)$$

Accuracy measures the proportion of correctly classified instances among all instances in the dataset. It provides an overall assessment of the model's correctness.

ii) Precision

$$Precision = \frac{True_p}{True_p + False_p} \quad (3.24)$$

Precision quantifies the fraction of true positive predictions among all positive predictions made by the model, showcasing its capacity to minimize false positives.

iii) Recall (Sensitivity)

$$Recall = \frac{True_p}{True_p + False_n} \quad (3.25)$$

Recall evaluates the ratio of true positive predictions to all actual positive instances in the dataset, illustrating the model's capability to detect all positive instances.

iv) F-score (F1-score)

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3.26)$$

The F-score represents the harmonic mean of precision and recall, offering a balanced assessment of a model's performance by considering both precision and recall.

v) Network Lifetime

Network lifetime refers to the duration for which the network remains operational before the depletion of resources or failure occurs. It is typically measured in rounds or time units.

vi) Network Throughput

Network throughput measures the rate at which data is successfully transmitted through the network. It is usually expressed in packets per second (pps) or bits per second (bps).

vii) Detection Rate

$$Detection\ rate = \frac{Number\ of\ detected\ instances}{Total\ number\ of\ instances} \quad (3.27)$$

Detection rate measures the proportion of correctly detected instances of a specific event or condition among all instances.

viii) False Positive Rate

$$False\ positive\ rate = \frac{False_p}{False_p + True_n} \quad (3.28)$$

False positive rate measures the proportion of negative instances that are incorrectly classified as positive by the model.

These evaluation metrics collectively provide a comprehensive assessment of the proposed methodology's performance across different dimensions, facilitating informed decision-making and optimization efforts.

3.4.3 Experimental Analysis

In Fig. 3.3, we compare the impact of malicious nodes on network lifetime across three methods: Federated learning [144], GALTM [145], and the proposed method.

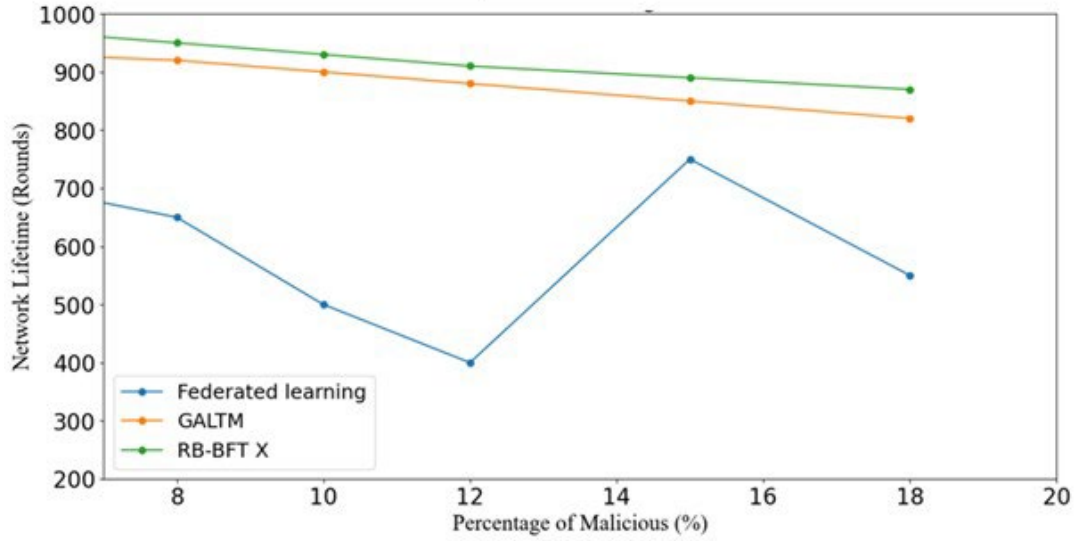


Figure 3.3: Network lifetime Vs Percentage of malicious

For the federated learning method, network lifetime decreases from 650 rounds to 550 rounds as malicious nodes increase from 8% to 18%. Similarly, for the GALTM method, the network lifespan drops from 920 rounds to 820 rounds with the same increase in malicious nodes. In contrast, the proposed method shows more resilience, with network lifetime only decreasing from 950 rounds to 870 rounds, indicating it is less affected by malicious activity. In Fig. 3.4, the analysis based on the effect of malicious nodes on network throughput for Federated learning [144], GALTM [145] and the proposed method are given.

For Federated learning, throughput decreases from 5800 packets to 6000 packets as malicious nodes increase from 8% to 18%, indicating reduced data flow. Similarly, GALTM sees throughput drop from 19,000 packets to 17,200 packets with an increase in malicious nodes. In contrast, the proposed method maintains relatively stable throughput, ranging from 20,000 to 18,000 packets, demonstrating better resilience against malicious activity and ensuring uninterrupted data transmission.

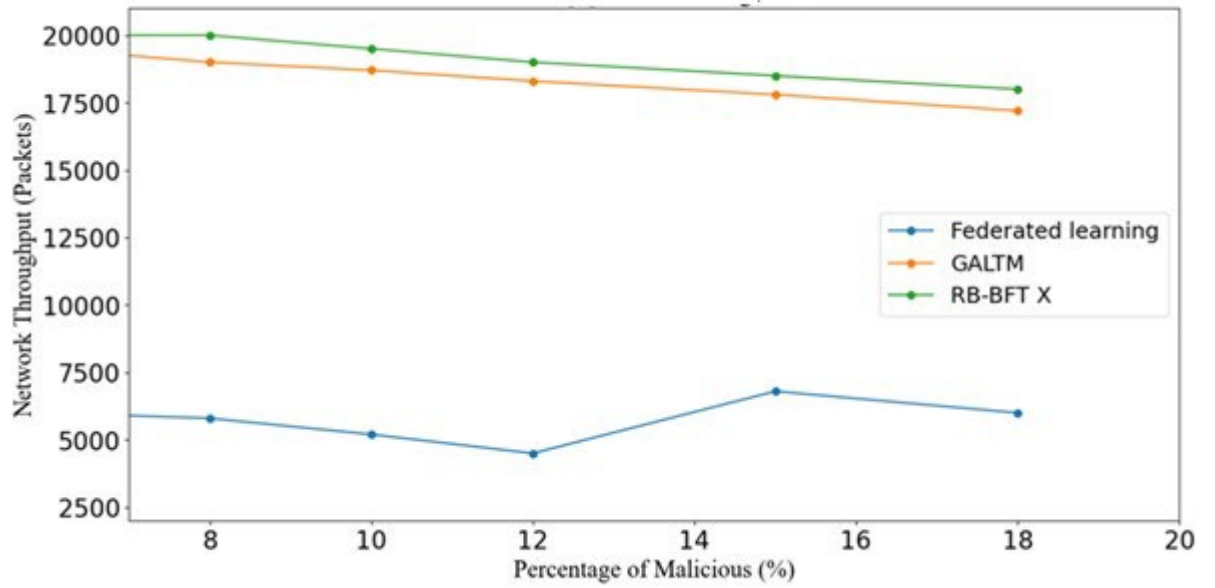


Figure 3.4: Network throughput vs percentage of malicious

Fig. 3.5 shows the relationship between detection rate and the percentage of malicious nodes for Federated learning [144], GALTm [145], and the proposed method. In the federated learning method, the detection rate decreases from 90% to 88% as malicious nodes increase from 10% to 20%. Similarly, for GALTm, detection increases from 92% to 96% as malicious nodes grow from 10% to 20%. In contrast, the proposed method shows a positive correlation, maintaining a detection rate of 94% to 97%, regardless of the increase in malicious nodes from 10% to 20%, demonstrating consistent and effective detection.

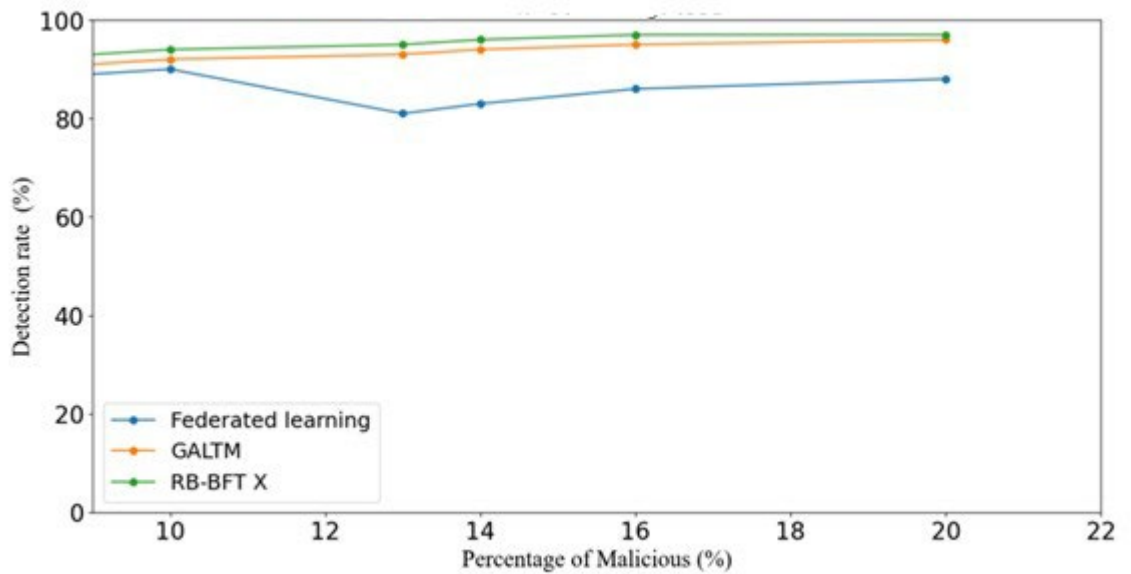


Figure 3.5: Detection rate vs percentage of malicious

Fig. 3.6 examines the impact of false positive rate (FPR) on the percentage of malicious nodes for Federated learning (FL), GALTm, and the proposed method.

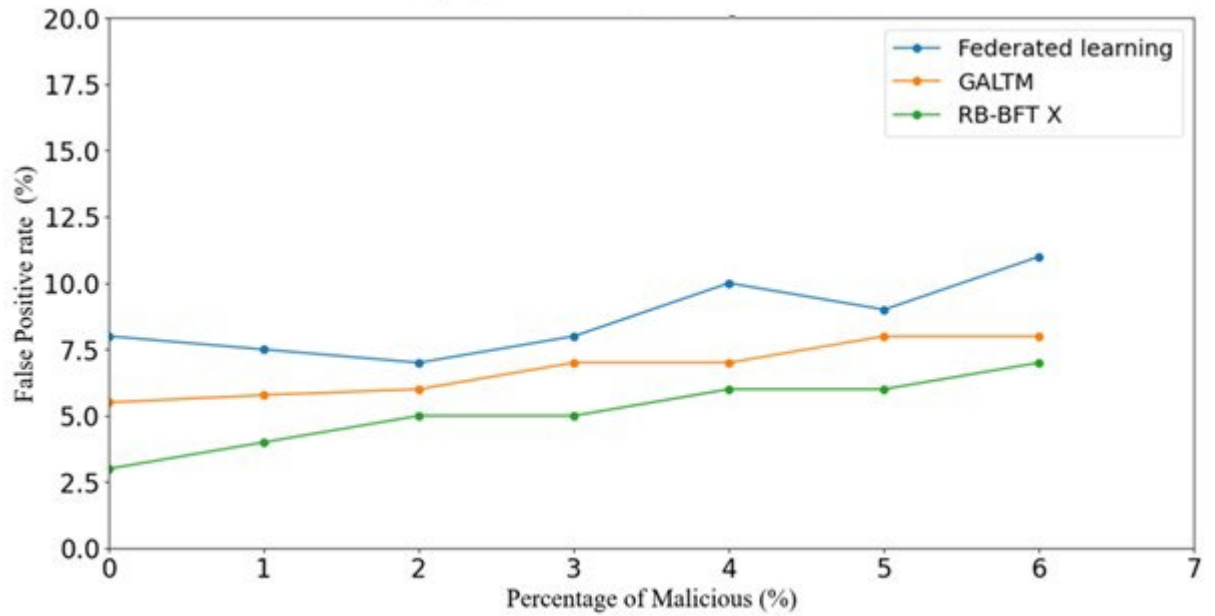


Figure 3.6: False Positive Rate vs Percentage of Malicious

In the federated learning method, the FPR increases from 6% to 11% as malicious nodes rise from 2% to 6%, indicating a higher misclassification rate. For the proposed method, however, the FPR remains stable between 5% and 7%, while the number of identified malicious nodes stays consistent between 2% and 6%, demonstrating the method's ability to minimize false positives while maintaining effective detection. The comparison of various methods in Fig. 3.7 for detecting malicious activities in a network reveals significant differences in performance across multiple metrics: accuracy, precision, recall, and F score.

In terms of accuracy, the Behavioral Fingerprint method [146] has a relatively low accuracy of 78.6%. In contrast, AgroKy [147] shows a substantial improvement with 90.43%, followed closely by Fog-IOT [148] at 90.13%. RDoS-CoAP [149] further improves accuracy to 96.90%, while the proposed method excels with the highest accuracy of 98.20%. When evaluating precision, which reflects the percentage of true positive detections among all positive detections, the Behavioral Fingerprint method achieves 79.2%, while AgroKy improves to 85.27%. Fog-IOT shows a higher precision of 90.10%, and RDoS-CoAP reaches 97.43%. The proposed method outperforms all others with a precision of 98.61%. In terms of recall, which measures the ability to detect all actual positive cases, the Behavioral Fingerprint method performs at 78.0%, while AgroKy achieves 86.31%. Fog-IOT further enhances recall to 89.95%, and RDoS-CoAP reaches 97.55%. The proposed method again outshines the others with the highest recall of 98.86%, demonstrating superior detection of all true positives.

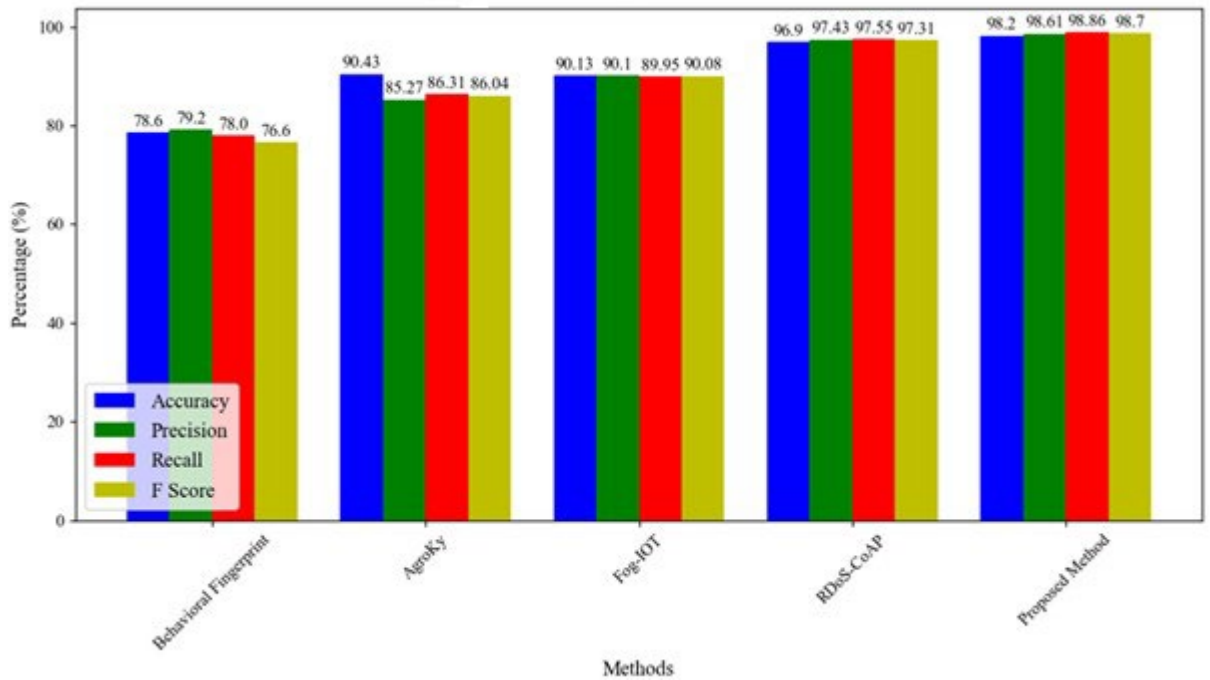


Figure 3.7: Performance comparison analysis

Finally, the F score, representing the harmonic mean of precision and recall, provides a balanced performance measure. The Behavioral Fingerprint method has an F score of 76.6%, while AgroKy improves to 86.04%. Fog-IOT achieves 90.08%, and RDoS-CoAP reaches an impressive 97.31%. The proposed method leads with the highest F score of 98.70%, indicating its overall effectiveness in maintaining a balance between precision and recall. The graph in Fig. 3.8 illustrates the relationship between window size (in seconds) and the number of packets processed in a network system.

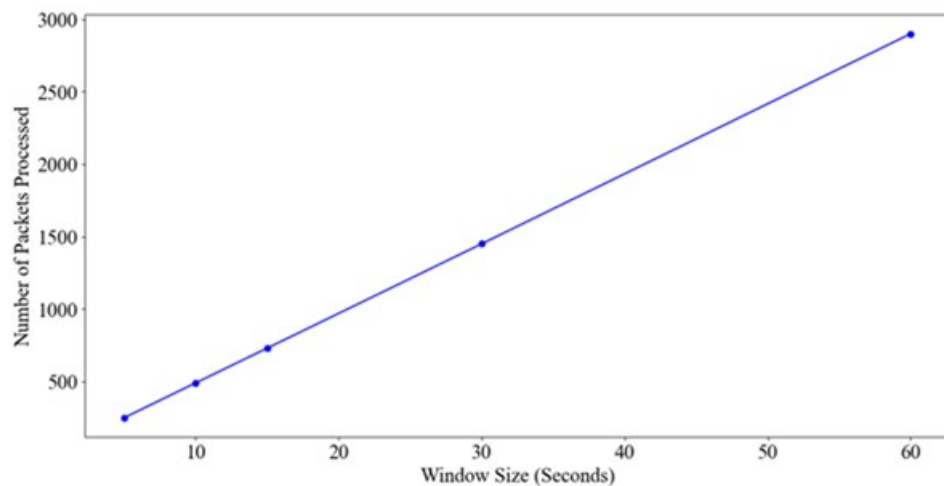


Figure 3.8: Packets processed vs window size

As the window size increases, the number of packets processed also rises. With a 5-second window, 250 packets are processed, suitable for rapid, real-time analysis. Doubling the window to 10 seconds results in processing 490 packets, demonstrating a significant increase. From the results obtained, it can be deduced that, within the 15-second segment, the number of packets that can be processed increases to 730, which shows an improved ability to handle data. Another important fact is that increasing the window of the stream interruption to 30 seconds brings this number to 1450 packets, which indicates that it is possible to collect and analyze more data compared to short windows. Last but not least, when the window size is 60 sec, the system processes 2900 packets, which presents its capacity to process many packets for analysis. In general, the diagram reveals that big windows enable more computations, which makes it possible to process a vast amount of information; on the other hand, possessing more complex windows is useful when performing instant analyses.

Scalability testing, as illustrated in Table 3.3, helps identify how a specific system performs under increased load or demand in resources to pinpoint areas of possible congestion or drag. The following table reveals the responses for all types of systems regarding load and difficulty: standard, average, high, and complex. As the workload increases, which suggests a growing number of devices, the latencies rise, and the resource demand reaches its maximum, though throughput drops slightly. This leads to higher error rates; however, response time stays fairly constant. Also, it was observed that depending on the complexity of the data, the resources are utilized to a different extent, where a higher complexity requires more CPU and memory. These observations allow for adding, removing, and modifying capacity, assigning resources, and enhancing systems with real-life conditions in mind.

Table 3.3: Scalability evaluation

Test Scenario	No. of Devices	Data Traffic (Packets/ Sec/ Device)	Latency (ms)	Throughput (Packets/Sec)	CPU Utilization (%)	Memory Utilization (MB)	Error Rate (%)	Response Time (ms)
Baseline	100	1	10	100	20	500	0.1	5
Moderate Load	500	5	20	2,500	40	1000	0.5	10
High Load	1000	10	35	10,000	60	2000	1.0	15
Peak Load	5000	50	50	250,000	80	5000	2.5	20

Extreme Load	10,000	100	75	1,000,000	90	10000	5.0	25
Low Complexity	1000	10	30	10,000	55	1500	1.0	12
High Complexity	1000	10	45	10,000	65	2500	1.5	18
Peak Complexity	5000	50	65	250,000	85	7000	3.0	22

Table 3.4 describes various schemes according to such parameters as adaptive to the configuration, integrity, protection of privacy data, ID authentication, fairness, authentication, and the time taken for the comparison. There is evidence of adaptive configuration regarding the proposed methodologies to achieve the required dynamics. While the proposed solution and Liu et al. [150] adopt the specification of essential security features, the scheme identified by Bodur et al. [151] needs more fairness. Rabie et al.'s [152] scheme has a notably higher time complexity, potentially impacting scalability. In contrast, the proposed methodology, along with Liu et al.'s, demonstrates linear time complexity, ensuring efficient resource utilization. Overall, the table underscores the importance of adaptability, security, and computational efficiency, positioning the proposed methodology as a promising solution for robust and scalable network systems.

Table 3.4: Method comparison

Method	Adaptive Configuration	Integrity	Privacy Protection	Identity	Fairness	Authentication	Time Complexity
Liu et al. [150]	No	Yes	Yes	Yes	Yes	Yes	$O(n)$
Bodur et al. [151]	No	Yes	Yes	Yes	No	Yes	$O(n \log n)$
Rabie et al. [152]	No	Yes	Not mentioned	Yes	Yes	Yes	$O(n^2)$
Proposed	Yes	Yes	Yes	Yes	Yes	Yes	$O(n)$

3.4.4 Performance Evaluation of Attack Detection

The Performance Evaluation of Attack Detection, focuses on assessing the effectiveness of the proposed security framework in identifying and mitigating various malicious attacks within the healthcare IoT network. This evaluation examines key performance metrics such as detection rate, false positive rate, latency, and throughput to determine how well the system performs under different attack scenarios. By simulating a range of attacks, including Brute Force, DDoS, MITM, and Sybil attacks, this section provides insights into the framework's ability to maintain robust security while ensuring minimal impact on network performance.

Table 3.5: Performance Evaluation of Attack Detection

Attack Type	Detection Rate (%)	False Positive Rate (%)	Latency (ms)	Throughput (pps)
Passive Eavesdropping	96.5	1.7	110	20,200
Active Intrusion	95.3	2.5	140	19,200
Insider Threats	97.0	1.3	115	19,700
Denial of Service (DoS)	96.8	2.1	150	18,500
Brute Force	97.5	1.2	120	19,800
Distributed Denial of Service (DDoS)	96.8	2.1	150	18,500
Man-in-the-Middle (MITM)	95.3	2.5	140	19,200
Data Tampering	97.0	1.3	115	19,700
Sybil Attack	98.2	1.1	100	20,500

Table 3.5 presents performance metrics for various malicious attack types simulated in the healthcare IoT network. The Detection Rate demonstrates the system's strong detection capabilities, with rates consistently above 95% for all attack types. Sybil attacks have the highest detection rate at 98.2%, followed by Brute Force attacks at 97.5%, and Data Tampering at 97%. The False Positive Rate remains low across all attacks, showing the system's ability to accurately distinguish between legitimate and malicious traffic, with the lowest false positive rate recorded for Sybil attacks at 1.1%, and the highest for MITM attacks at 2.5%.

In terms of Latency, DDoS and DoS attacks introduce the highest delays at 150 ms, followed by MITM attacks at 140 ms. Sybil attacks cause the least disruption with a latency of just 100 ms. As for Throughput, the system maintains high data transmission rates despite the attacks, with Sybil attacks achieving the highest throughput at 20,500 packets per second (pps), followed closely by Fingerprinting attacks at 20,200 pps. Brute Force and Data Tampering attacks both maintain a throughput of 19,800 pps and 19,700 pps, respectively, while MITM attacks and DDoS

attacks result in slightly lower throughput at 19,200 pps. These performance metrics underscore the framework's robustness in handling diverse attack scenarios, maintaining high detection rates, low false positives, and efficient network performance even under attack.

3.4.5 Ablation study

Ablation studies, a cornerstone of research methodology, meticulously dissect the impact of individual components on system performance under various method, illuminating crucial insights in various domains. The ablation study results are depicted in graphs, as shown in Fig. 3.9.

Method A: Proposed model

The proposed model achieves the highest performance across all metrics with all components included. The accuracy is 98.5%, precision is 98.2%, recall is 98.8%, and F-score is 98.5%. The latency is 10 ms, throughput is 1500 packets per second, and the security metric (intrusion detection rate) is 99.2%. These results indicate the effectiveness of the integrated approach in providing robust security and high performance in the healthcare network.

Method B: Without RB-BFT X

Eliminating the Redundant Byzantine Fault Tolerance with Extensions component leads to a significant reduction in all key performance metrics. The model's accuracy declines from 98.5% to 95.0%, precision is reduced from 98.2% to 94.5%, recall is lower from 98.8% to 94.8%, and the F-score decreases from 98.5% to 94.6%. This fact demonstrates just how critical RB-BFT X is for the said model, as it ensures the targeted accuracy and precision in terms of data identification and processing. Moreover, latency specifications rise from 10ms to 15ms, which explains slow data transmission and throughput reduction from 1500 packets/second to 1400 packets/second, signifying poor data processing ability. The most sensitive category is security, under which the intrusion detection capability is reduced to as low as 99.2% to 96.0%. This means that RB-BFT X is necessary for the network, as it maintains its strong fault tolerance and security level.

Method C: Without RBAC

Disabling of RBAC leads to a slight general degradation of performance indicators such as accuracy, which in some instances may drop from 98.5% to 96.5%, precision from 98.2% to 96.0%, recall from 98.8% to 96.2%, and the F-score from 98.5% to 96.1%. This points out that RBAC significantly improves scalability, although it is not the only factor contributing to this. The jump from 10 ms of latency up to 12 ms and the ever so slight reduction in throughput from 1500 packets/s down to 1450 packets/s indicate that RBAC plays a role in the effective administration of data from a security perspective. The impact on the security aspect has lowered significantly to a staggering 99.2% to 97.5%. Finally, the practices showed that the total healthcare IoT network traffic is driven toward RBAC, which underlines its

critical role in securing healthcare IoT network access.

Method D: Without Behavioral Anomaly Detection

When using a model without Behavioral Anomaly Detection, it has a moderate decrease in performance compared to the first model, with accuracy from 98.5% to 97.2%, precision from 98.2% to 96.8%, recall from 98.8% to 97.1% and the F-score from 98.5% to 96.9%. From the results, it is clear that latency rises from 10 ms to 11 ms and throughput drops from 1500 packets/s to 1480 packets/s, which means there is a slight delay in communication and a decline in the transmission rate. That effect is reflected in the security aspect, where the intrusion detection rate decreases from 99.2% to 97.8% of the participants agreed that anomaly detection effectively reduces alert fatigue and enables analysts to recognize security threats within the network.

Method E: Without Coati Chase

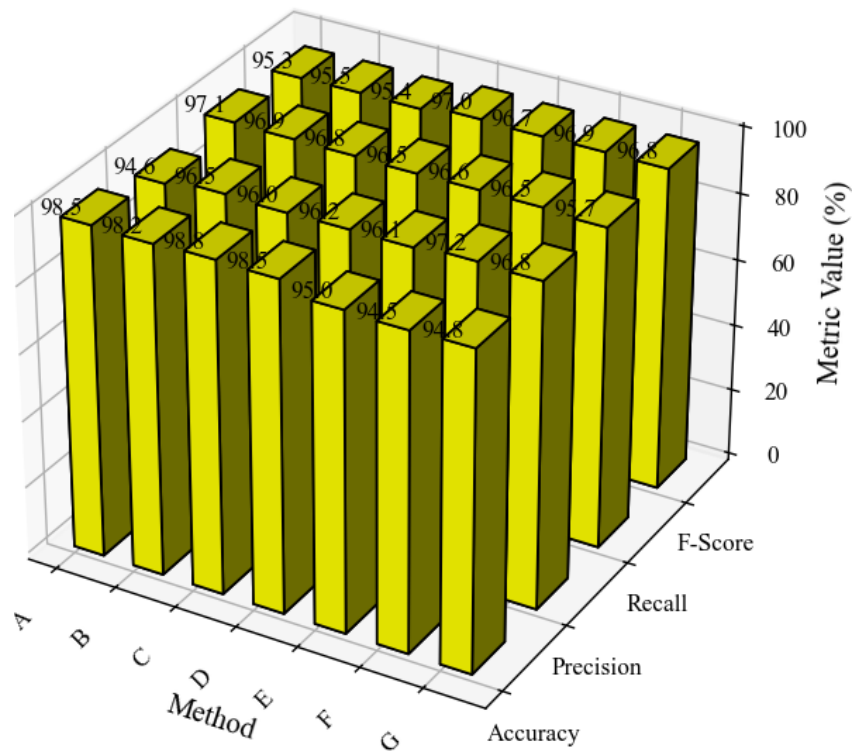
When the Coati Chase component is removed, it positively affects the accuracy ratio negatively, thus producing a lower score of 96.8%, precision from 98.2% to 96.5%, recall from 98.8% to 96.6%, and the F-score from 98.5% to 96.5%. Latency rises from 10 ms to 14 ms, which shows the existence of slow communication paths, and throughput decreases from 1500 packets/s to 1420 packets/s, evidencing the relational mode of Coati Chase in data routing. The security metric is also impacted depending on the IDS type, with the ratio of detected intrusion decreasing from 99.2% to 97.3 %; Coati Chase would hence contribute towards ensuring the safety and reliability of the routing process within the healthcare IoT network.

Method F: Without Escape from Predators

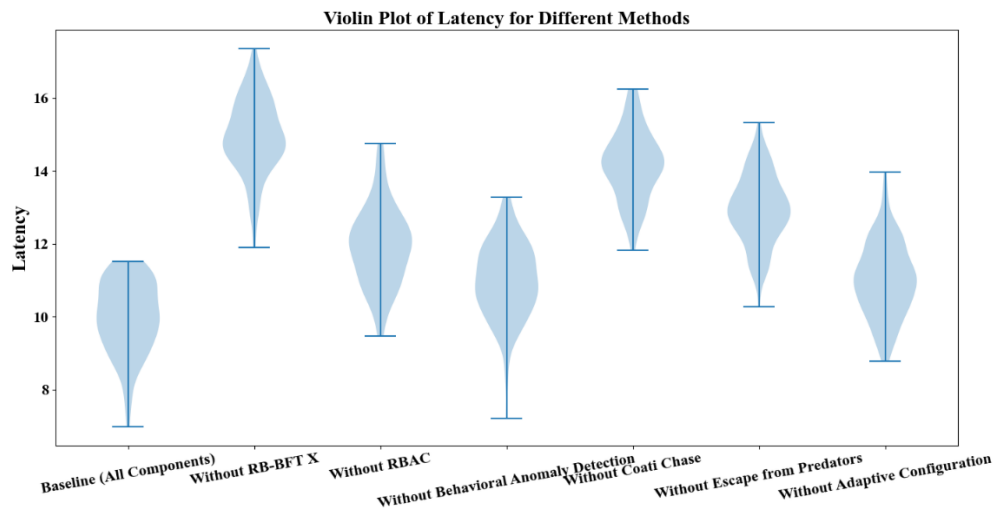
The conclusion that can be drawn from the figures is that the performance parameters are lower after pretraining without the Escape from Predators component, where average accuracy goes down from 98.5% to 95.7%, precision from 98.2% to 95.3%, recall from 98.8% to 95.5 % and F-score from 98.5% to 95.4%. The latency, which was at 10 ms, rises to 13 ms, and the throughput, 1500 packets per second, is reduced to 1430 packets per second, suggesting that data transmission becomes less efficient. The security impact is substantial, with the intrusion detection rate falling from 99.2% to 95.9%, highlighting its role in providing dynamic security mechanisms to protect against potential attacks.

Method G: Without Adaptive Configuration

It demonstrates that when there is no Adaptive Configuration, the accuracy is lowered from 98.5% to 97.0%, precision from 98.2% to 96.7%, recall from 98.8% to 96.9%, and F-score also dropped from 98.5% to 96.8%. Over the five runs, they vary the latency by going from 10 ms to 11 ms while reducing the throughput from 1500 packets/s to 1470 packets/s, suggesting that adaptive configuration helps continually optimize performance. The security metric also declines, with the intrusion detection rate dropping from 99.2% to 97.6%, underscoring the importance of adaptive configuration in ensuring the system can adapt to changes and maintain security.



(a) Based on performance metrics



(b) Based on Latency

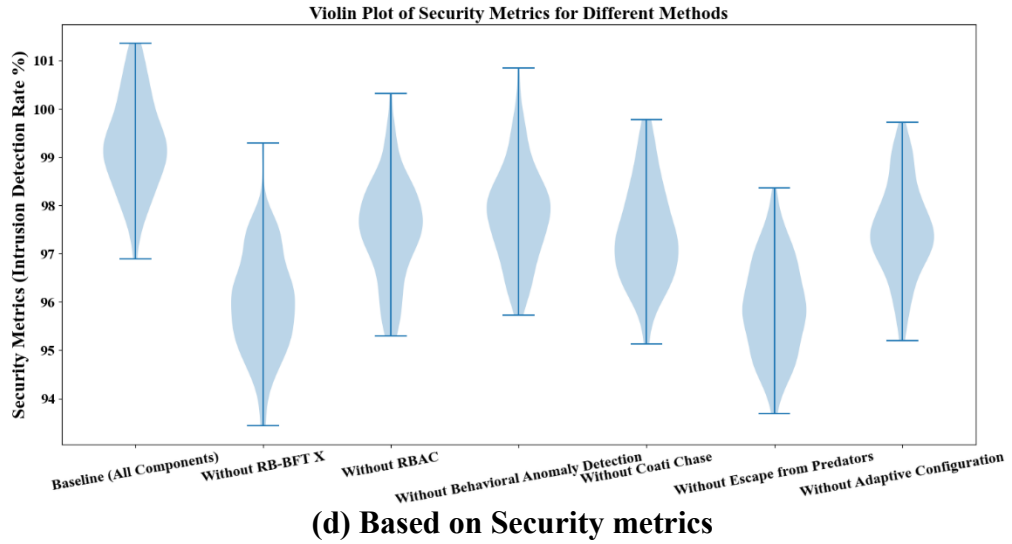
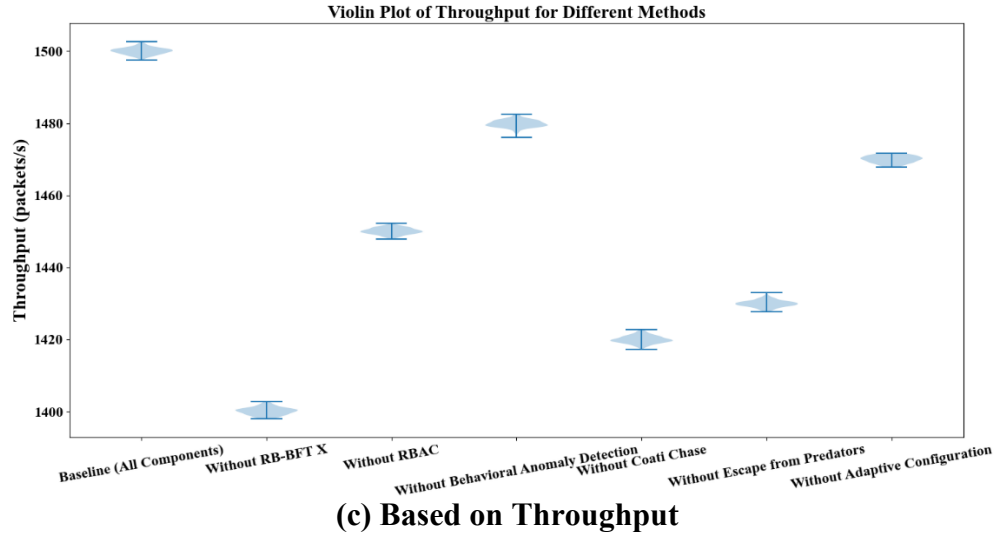


Figure 3.9: Ablation results

3.5 Chapter Summary

In this chapter, a new approach was proposed combining advanced anomaly detection methods with adaptive configuration techniques customized for healthcare IoT environments. Its goal was to strengthen network resilience against nodes while ensuring data transmission and reducing false alarms. By utilizing behavioral Anomaly Detection, Role-Based Access Control, Dynamic Routing algorithms in the Redundant Byzantine Fault Tolerance with Extensions and Coati-based network (CoatiNet), a new solution for identifying and addressing potential security threats has been given. Even with threat activity in the network, it ensures the network is stable and data integrations are provided. Continuously adapting IoT network configurations can customize the network defense mechanism based on evolving and changing threats to benefit the IoT system. The performance evaluation of the proposed method was based on metrics such as accuracy, precision, recall, and F score, which show improvement. As revealed

from the evaluation metrics, CoatiNet performs better than other routing algorithms by keeping the network lifetime and throughput stable while facing malicious nodes. Furthermore, the offered configuration capacities make it possible to seize existing regulatory and resource conditions and adjust to the changing conditions. Thus, RB-BFT X and CoatiNet are the core contributors to the architecture of the presented methodology, where security measures are critical to dynamically adapting the network in healthcare IoT.

CHAPTER 4

A BLOCKCHAIN BASED BONY-ISHO PROTOCOL FOR SECURE CLUSTERING AND ROUTING IN IOT BASED NETWORKS

Among the technological marvels driving this transformation, the IoT has emerged as a pivotal contributor [153][154]. Within this expansive landscape, WSNs stand as an integral component of IoT systems [155]. These sensor nodes, strategically positioned to monitor physical conditions in various locations, generate an immense volume of real-time data, which is subsequently transmitted to other devices or to the cloud [156]. While this proliferation of IoT devices promises unprecedented opportunities, it also presents significant challenges. One such challenge is the rising tide of cybercrime attacks, which have become increasingly sophisticated and difficult to thwart, especially when it comes to securing the vast amount of data generated by these devices [157]. What sets these devices apart is their ability to operate in large networks, enabling the identification of critical events and the periodic collection of environmental data [158]. However, their limited energy resources pose a unique challenge, as replacing depleted batteries can be prohibitively costly or even infeasible in certain hostile environments [159].

4.1 Introduction

To ensure the security and integrity of data in clustered sensor networks, various techniques have been introduced, including trust-based, fuzzy-based, and cryptographic methods such as the Advanced Encryption Standard (AES) and Blowfish [160] [161] [162] [163] [164]. These techniques play a vital role in safeguarding data during its transfer within the network. Previous research in the domain of secure and energy-efficient IoT-based WSNs has displayed notable limitations, underlining the significance of the proposed Blowfish Honey-Improved Spotted Hyena Optimization (BONY-ISHO) protocol. Existing approaches have demonstrated challenges in scalability, adaptability to dynamic network changes, and insufficient security measures. Some methods have shown limitations in data routing efficiency and a need for enhanced security. Others rely on static assumptions about node reliability and require more comprehensive security measures. A few techniques focus on energy efficiency but lack security enhancements, while others have high energy consumption and limited security focus. Additionally, some approaches introduce complexity and potential points of failure. These limitations collectively underscore the need for a comprehensive and innovative protocol like BONY-ISHO, which combines advanced security mechanisms, energy-efficient routing, and adaptive

algorithms to create a robust and secure framework for IoT-based WSNs. This chapter proposes several innovative elements aimed at enhancing security and energy-efficient routing in WSNs. A key innovation involves the integration of a blockchain-based security mechanism, including Smart Contracts (SC), which harnesses distributed ledger technology to ensure the integrity of node identities and data authenticity. This pioneering security measure establishes a solid foundation for secure network access, effectively mitigating the risk of unauthorized node entry. The bidirectional authentication mechanism, operating through blockchain records, creatively establishes secure communication channels, effectively barring unauthorized nodes from participating in data transmission. To ensure the security of data during transmission, a hybrid Blowfish-Honey cryptographic technique is employed, effectively safeguarding encryption keys against unauthorized access, thereby enhancing data privacy and security. Additionally, a novel contribution to this work is the Improved Spotted Hyena Optimization (ISHO) algorithm, drawing inspiration from the social behavior and hunting strategies of spotted hyenas. ISHO introduces a unique approach to energy-efficient routing by factoring in critical parameters such as residual energy, distance, and link quality. Innovative fitness evaluation criteria, including residual energy, distance, and link quality, are introduced to select nodes with optimal fitness for routing decisions, further enhancing energy efficiency. Furthermore, the incorporation of a spiral model within the ISHO algorithm enhances the efficiency and effectiveness of the optimization process, achieving a harmonious balance between exploration and exploitation within the search space. The following are the proposed work's contributions:

- **Secure and Efficient Clustering:** Introduction of blockchain technology to enhance the security and efficiency of clustering. This innovation ensures that only authenticated cluster head and member nodes can participate in the network, significantly improving network security.
- **Hybrid BONY Cryptographic Technique:** Proposal of a hybrid Blowfish-Honey (BONY) cryptographic technique for data protection within the Area of Interest (AOI). This approach offers dual-layer data security, utilizing Blowfish encryption for data protection and Honey encryption for key management, ensuring robust privacy and security.
- **Optimized Routing via Enhanced SHO Algorithm:** Enhancement of the Spotted Hyena Optimization (SHO) algorithm to create ISHO, optimizing data transmission routes based on factors like distance, link quality, and node energy levels. ISHO extends the network's lifetime and conserves energy resources, improving energy efficiency.
- **Performance Evaluation:** Rigorous performance evaluation of the proposed BONY-ISHO protocol against existing methodologies and state-of-the-art techniques. This evaluation demonstrates the protocol's effectiveness in terms of packet delivery ratio, throughput, energy utilization, and latency, showcasing its superiority in real-world scenarios.

The rest of the chapter is structured as follows: Section 4.2 describes the proposed blockchain-based BONY-ISHO protocol, Section 4.3 presents the implementation of the proposed protocol and compares the results with recent works,

and finally, Section 4.4 concludes this chapter.

4.2 Proposed Blockchain based BONY-ISHO Protocol

The architecture for the proposed BONY-ISHO based secure clustering and energy-efficient routing technique is shown in the following Fig. 4.1.

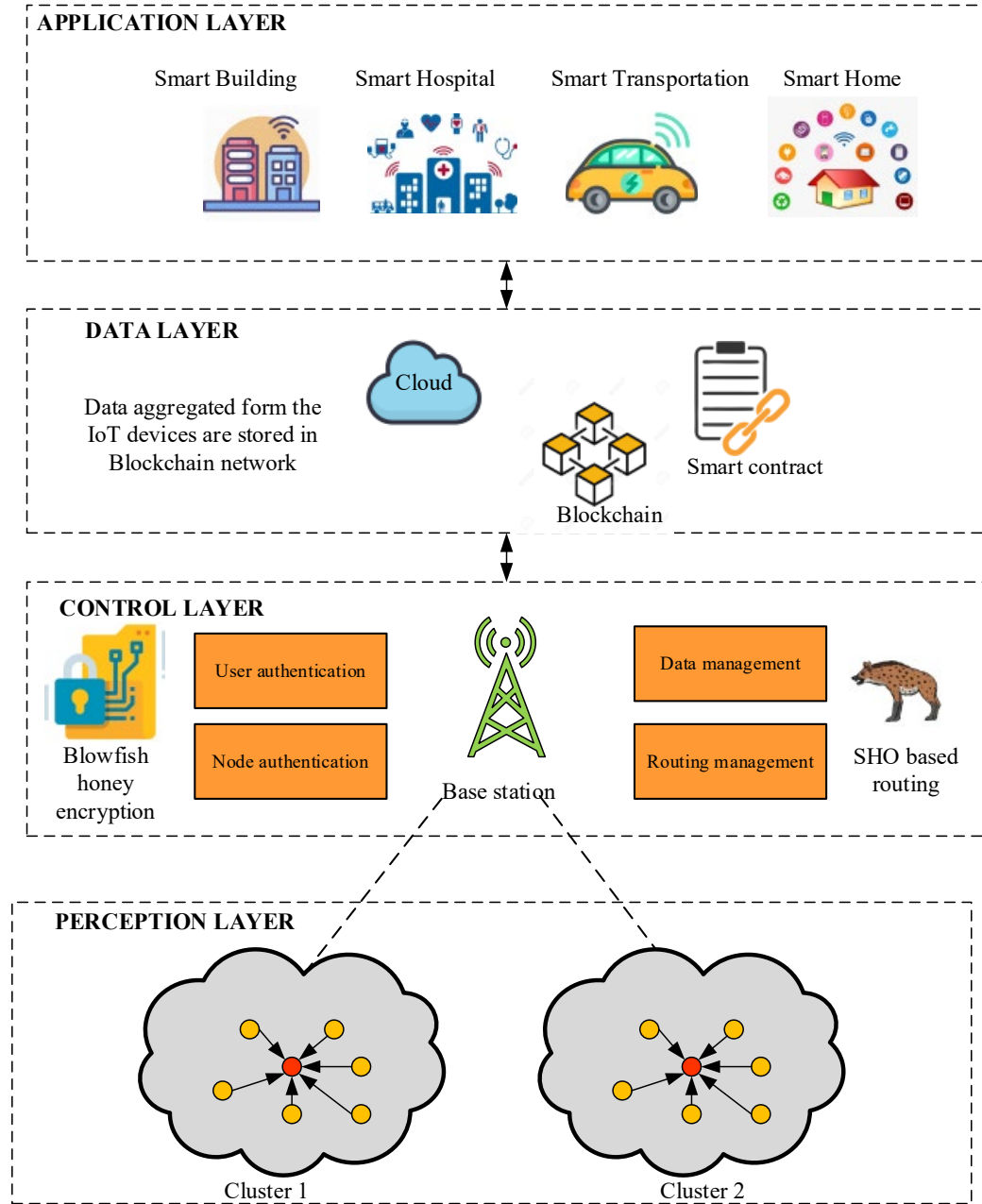


Figure 4.1: Architecture for the Proposed BONY-ISHO Based Secure Clustering and Energy Efficient Routing Technique

The blockchain approach is used in our proposed Blowfish Honey-Improved Spotted Hyena Optimization protocol to improve privacy and security in IoT-based

WSNs. This blockchain consists of a Smart Contract, which is used to verify and authenticate the nodes and the data transferred in the network. Initially, the base station confirms the identities (IDs) of the cluster head (CH) and member nodes when they first connect by comparing them to information recorded in a blockchain.

A node can join the network if its ID is the same as one of those in the blockchain. In this way, the clustering is made more secure, and thus, it becomes impossible for malicious nodes to enter the network. Secondly, the data collected for the Area of Interest is securely transmitted to the destination using a hybrid Blowfish-Honey cryptographic technique. Additionally, the ISHO algorithm chooses the data transmission line that uses the least energy to save energy while enhancing service quality. The following sub-sections discuss the proposed BONY-ISHO protocol steps, including initialization, clustering, encryption, and optimal route selection. Here, each stage is thoroughly explained.

4.2.1 Initialization

Each node has a unique identity, which the base station stores on the public blockchain for each node. The base station must first initialize all nodes in the network before deploying them. To begin, the base station determines each node's identifier, including its own. Since each node in the network has a distinct Ethernet address ER_j , the base station hashes the Ethernet address using the hashing function to obtain the node's unique identification $UID_j = hash(ER_j)$ and delivers it to each node for storage. MID denotes the identity of the regular member node, CHID denotes the identity of the CH node, and BSID denotes the identity of the BS. Finally, each node must have an ID card generated by the base station to verify its identity.

4.2.2 Clustering

Each (IoT device) node in the WSN calculates the distance to the base station, the distance to the neighboring nodes, and the residual energy, and which they share with each other [165]. The nodes then choose a CH by comparing the scores received from each other, and the rest of the nodes join as member nodes in the respective cluster. The smart contract mechanism examines the node ID to see whether it matches any IDs kept in the blockchain when a node seeks to join a cluster. If it does not match any ID stored in the blockchain, the registration process fails, and the BS blocks the respective node from the network. The CH nodes are registered through the public blockchain, and the member nodes are registered via the local blockchain. The next sub-section describes the cluster head and member node registration processes.

(i) CH Registration

Once the CHs are elected, registration request messages are submitted by CH nodes. On the public blockchain, the smart contract will carry out the registration verification procedure and perform the steps below:

- Accessing the identification of the node stored in the public blockchain allows for determining whether CH nodes exist. They are not validated if the node already exists.

- Check the validity of the Ethernet addresses of BS (ER_{BSID}) and CH (ER_{CHID}).

The CH node will not be registered, and a registration error message will be sent if any of the aforementioned stages fail to validate. When all of the procedures are successfully validated, the public blockchain records the CH's identity and broadcasts the validated message.

(ii) Member Node Registration

Member node registration verification takes place on the local blockchain. Because there are many CH nodes, each member node can only join one cluster. When a cluster is selected, the CH receives a registration request message from each member node. When a CH receives a registration request, it first validates if the timestamp is appropriate. The registration process is established in the local blockchain network to activate the smart contract for the member node if the timeliness requirement is met. The following is the order in which the registration procedure is carried out:

- The smart contract's local blockchain obtains the identity details of all nodes from the public blockchain and checks if the identity of the member node seeking registration is already present; if it does, the authentication fails.
- Check and validate the identity of BS.
- Check and validate the identity of the CH node.
- Check and validate the identity of the member node.

The registration error message will be issued if any of the aforementioned stages fail to validate. Once all of the procedures are successfully validated, the local blockchain will upload the member node's identity to the public blockchain. The local blockchain then approves that the member node can connect to the particular cluster network.

(iii) Authentication

The BS will transfer data gathered from the IoT devices to the user. However, battery life is an essential requirement to consider in WSN. A single node failure might cause the entire network to fail. Moreover, since the sensors are usually positioned in hostile places like military bases and smart cities, clustered IoT devices are susceptible to being captured or attacked by intruders. If an intruder manages to compromise a CH, the entire communication within this network gets affected. Therefore, it is crucial that the CH and other nodes not compromise.

If node A wants to communicate with node B , bidirectional authentication between node P and node Q is required to establish a secure channel. Connect request is sent to cluster head node after node A starts authentication process. When the CH node gets the connection request message, the smart contract is triggered, and the authentication request is verified on both the local and public blockchain. The following three steps are followed during verification:

- The validity of the ID P_{Idcard} is ensured by using the node data stored in the blockchain. Proceed to the next steps if it is validated; else, return the error.
- If the identity of node P or Q does not exist in the information recorded on the blockchain, it returns an error.
- Check the state of the nodes P_{MID} and P_{CHID} if neither is alive, an error is returned.

With the node identification information contained in the blockchain, the local blockchain queries the identity of node P and node Q . If they are in the same cluster network, true is returned to the CH nodes immediately, and nodes P and Q will initiate a secure channel. If the nodes belong to a separate WSN subnet, the authentication will be performed by the public blockchain.

4.2.3 Data Aggregation and BONY Encryption

WSNs are devices with limited resources that have the aim of collecting data from the surroundings. They are capable of sensing, handling, and sharing information. The sensors in WSN are able to sense the surroundings and monitor environmental conditions in order to gather the data.

Blowfish-honey cryptographic technique is used to encrypt the data that is collected by the nodes from the area of interest. The cipher text is constructed by encrypting the aggregated data with the blowfish technique. To access the data, the data requester (user) must have this secret key. However, sharing this secret key using conventional technique may cause unauthorized access. As a result, Honey encryption is used in order to increase the security of the key exchange procedure. Honey words are bogus passwords added to the original passwords that are intended to confuse the attackers. Therefore, even if an intruder manages to access the password file (secret key) from the database, the intruder cannot access the data due to the invalid password generated by the honey encryption algorithm.

(i) Blowfish Algorithm

The Blowfish algorithm is a secure, lightweight, and public domain 64-bit block cipher. Fig. 4.2 shows a graphical illustration (Fiestal network) of the Blowfish algorithm.

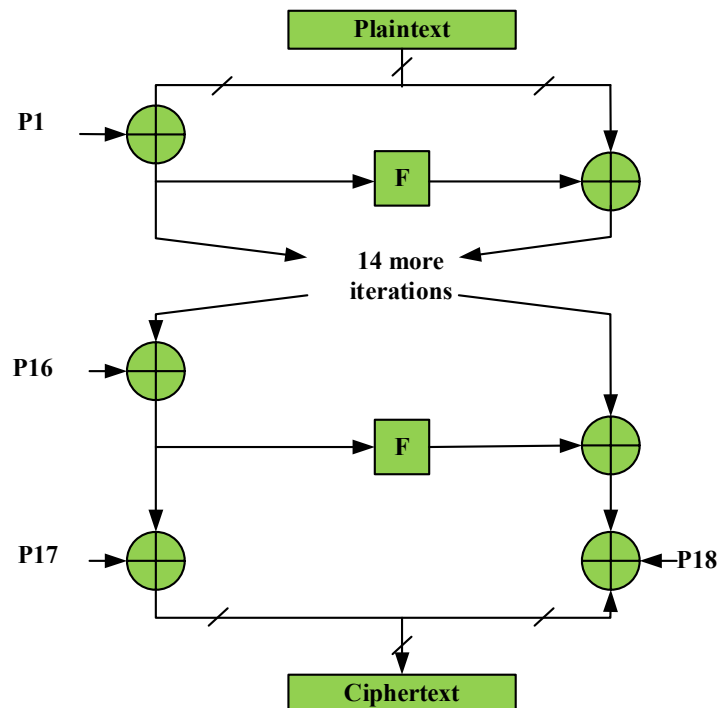


Figure 4.2: Blowfish algorithm

The Feistel network used in Blowfish has 16 rounds. We integrate this technique with Honey Encoding to enhance security. Fig. 4.3 shows a graphical depiction of the Function module.

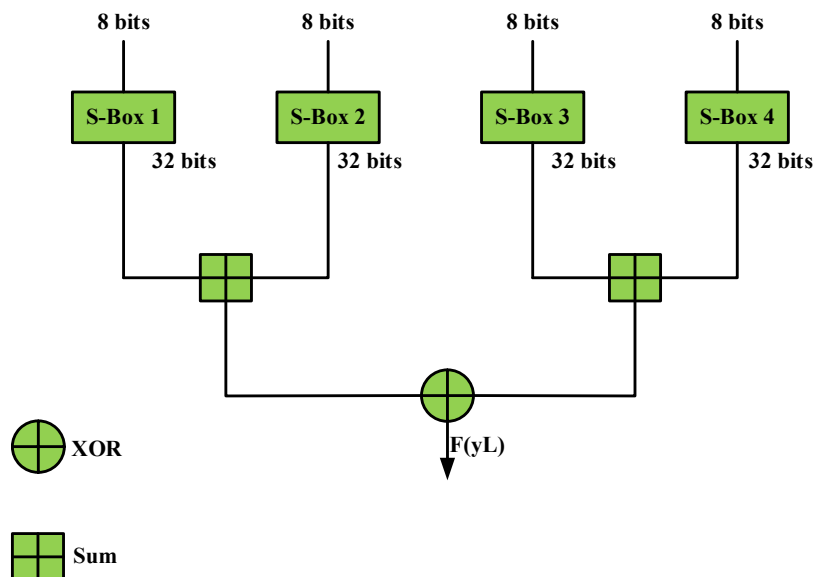


Figure 4.3: Function module

It accepts a 32-bit input and breaks it into four bytes, which are used as S-array indices. The output is created by combining and XORing the lookup results. Since Blowfish is a symmetric method, the decryption and encryption procedures are the

same. The S array and the P-array are effectively created from the user's key. Sub-key creation is the term for this procedure. The S-array and P-array do not need to be recalculated if the key does not change, but they must be kept secret.

Key Generation

1. The P-array and S-boxes are initialized using a predetermined string of pi hexadecimal digits.
2. The first 32 bits of the key are XORed with P1, followed by 32 bits with P2. This process should be repeated until the P-array is XORed with the final key bits.
3. The subkeys blowfish algorithm is used to encrypt every single string of zeros.
4. The results of step 3 are substituted for the P1 array and P2 array, respectively.
5. P3 and P4 are generated by encrypting the step 3 output with updated subkeys.
6. Step 5 results are utilized to replace the P3 and P4 arrays.
7. Repeat the process, substituting the Blowfish algorithm output for each entry in the four S-boxes and P-array.

Encryption

The input of Blowfish is a 64-bit data of Y.
Split Y into two equal halves (Y1, Y2) of 32-bit
For j= 1 to 16
Y1= P18 XOR Y1
Y2= P17 XOR Y2
Integrate Y1 and Y2

The updated blowfish algorithm makes changes to the original function module. The results of S-box 1 and 2 are summed in the original function, and then the output of S-box 3 is XORed with it. It is summed with S-box 4's output to get F(xL). Modulo addition is used here. There are two modulo additions in the original function, which are performed simultaneously. S-box 3 and S-box 4 outputs, as well as S-box 1 and 2 outputs, are summed in parallel. The results are then XORed to get (xL).

Honey Encryption

Since Blowfish is a symmetric key cryptosystem, it has to deal with the problem of securely exchanging secret keys. In order to circumvent these problems, Honey encryption with a larger buffer size of the password was combined with Blowfish while maintaining its benefits and improving its strength. Users' true passwords are stored in a database with the honey words in order to prevent brute force attacks (BFA). A fictional password that is used to confuse attackers who get access to the database's password file is known as honey word. The honey words in the password file prevent attackers from determining the true password. The combined storing of false and actual passwords is referred to as sweet words, and the user's real password is referred to as sugar words. Fig. 4.4 depicts the flowchart for the BONY encryption algorithm.

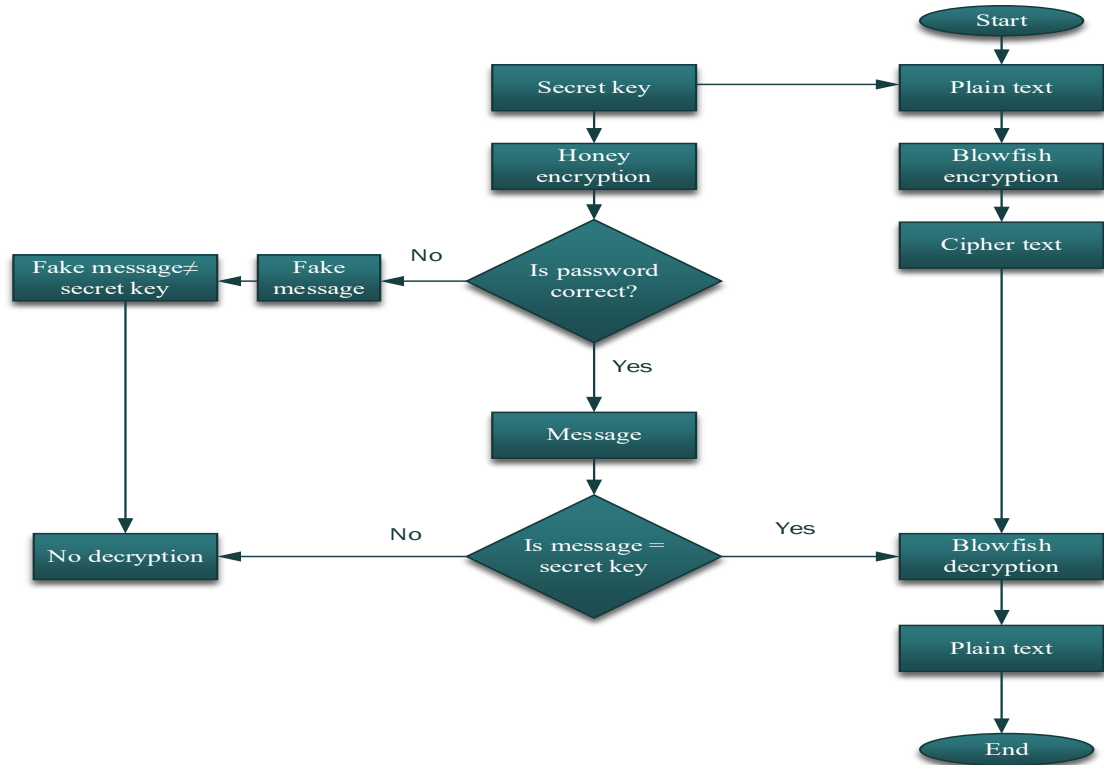


Figure 4.4: Flowchart for BONY encryption

4.2.4 Optimal Route Selection

When transmitting the aggregated data, choosing a route without considering the distance to the destination and link quality decreases the lifetime. Therefore, the best route is selected using the ISHO for transferring the data to the end-user. Female spotted hyenas are dominant and reside in their own clan. Male members, on the other hand, abandon their clan once they reach adulthood and join another clan. They are the lowest-ranked members in a new clan and hence receive the smallest portion of the meal. A new male hyena of the clan always hangs around with the same group (friends) for a long period. A female, on the other hand, is always guaranteed a secure position. The hyenas use sound to communicate with one another when hunting for food.

This algorithm is based on the social relationships and hunting activities of spotted hyenas. The SHO algorithm imitates the trusty spotted hyenas' coherent groupings. Searching, surrounding, hunting, and attacking are the four key phases of SHO. The hunting behavior in the SHO algorithm is steered towards the best search agent and preserves the best optimum solutions by trusted friends (so far, best solutions) in the group.

Fitness Evaluation:

The nodes chosen with the best fitness have the shortest route to the sink, the best link quality, and the least residual energy.

(i) Residual Energy

The energy consumed during transmission and reception is determined [165]. Eqn. (4.1) adds the energy lost during transmit and receive to determine the overall energy consumed.

$$E_{tot(cons)} = E_{trans} + E_{recp} \quad (4.1)$$

Where E_{recp} and E_{trans} are the reception and transmission energy.

The sensor node's remaining energy is referred to as residual energy and is calculated as,

$$residual \text{ energy} = E_{initial} - E_{tot(cons)} \quad (4.2)$$

In which $E_{initial}$ represents a node's initial energy.

(ii) Distance

The node that has less distance to the sink is considered to have better fitness. The distance of node k to sink S is computed using Eqn. (4.3).

$$dis(k, S) = \sqrt{(y_k - y_S)^2 + (x_k - x_S)^2} \quad (4.3)$$

Where (y_k, y_S) and (x_k, x_S) are the coordinates of the node and the sink.

(iii) Link quality:

The link quality of two nodes a and b is computed using Eqn. (4.4).

$$L_{quality} = \begin{cases} 1, & \text{if } dist(Node_a, Node_b) \leq RG_{comm} \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

Where $dist(Node_a, Node_b)$ is the distance between node a and node b, RG_{comm} signifies the node's communication range.

4.2.4.1 Encircling

The distance between the spotted hyena and the prey is computed using Eqn. (4.5).

$$\vec{E}_g = |\vec{V} \cdot \vec{S}_r(b) - \vec{S}(b)| \quad (4.5)$$

$$\vec{T}(b+1) = \vec{S}_r(b) - \vec{W} \vec{D}_g \quad (4.6)$$

where b signifies the present iteration, \vec{S} and \vec{S}_r signifies the position vector of hyena and the prey respectively, \vec{V} and \vec{W} indicates the coefficient vector and is computed using Eqn. (4.7) and Eqn. (4.8).

$$\vec{V} = 2 \cdot q \cdot \vec{e}_1 \quad (4.7)$$

$$\vec{W} = 2 \vec{g} \cdot q \cdot \vec{e}_2 - \vec{g} \quad (4.8)$$

$$\vec{g} = 5 - (Itr * (5 / Max_{Itr})) \quad (4.9)$$

where $Itr = 1, 2, \dots, Max_{Itr}$. In order to have proper balance between exploitation and exploration, \vec{g} is reduced gradually from 2 to 0.

4.2.4.2 Hunting

$$\vec{E}_g = |\vec{V} \cdot \vec{S}_g - \vec{S}_l| \quad (4.10)$$

$$\vec{S}_l = \vec{S}_g - \vec{W} \cdot \vec{E}_g \quad (4.11)$$

where \vec{S}_l and \vec{S}_g indicates the location of the hyena with best fitness and the location of other hyenas.

Cluster with M number of optimal solutions is determined using Eqn. (4.12).

$$\vec{B}_g = \vec{S}_l + \vec{S}_{l+1} + \dots + \vec{S}_{l+M} \quad (4.12)$$

The number of hyenas M is computed using Eqn. (4.13).

$$M = count_{NS}(\vec{S}_g, \vec{S}_{g+1}, \dots, (\vec{S}_g + \vec{N})) \quad (4.13)$$

Where $count_{NS}$ indicates the number of solutions, M signifies the random vector in the range $[0.5, 1]$.

4.2.4.3 Attacking The Prey

The location of the spotted hyenas is updated using Eqn. (4.14).

$$\vec{S}(b+1) = \frac{\vec{B}_g}{M} \quad (4.14)$$

Improved SHO

Spotted hyenas update their locations in traditional SHO based on the location of a group of spotted hyenas in vector C . The value of the group of spotted hyenas is assumed to be a single best optimum solution. The spiral model is used in the method to update the locations of the spotted hyenas in relation to the most optimum solution.

The spiral radius of each turn is computed using Eqn. (15).

$$R = f \times e^{CR} \quad (4.15)$$

$$\vec{S}(b+1) = \vec{E}_g \times R \times \cos(C) + \vec{S}_g \quad (4.16)$$

Where R signifies the random number in the range $[0 \leq C \leq 2\pi]$, f and C are the constants that are used to define the spiral shape. The value of f and C is set to 1.

In this method, the vector V is also changed. In Eqn. (4.17), the control parameter V represents the weight of the prey for computing distance. This parameter's value is significant and is crucial for exploration throughout the search process. SHO with $V > 1$ has a higher exploitation rate than SHO with $V < 1$. Previous generations $V > 1$ allow for better exploration of the search space. It amplifies the influence of prey weight if V it is more than 1, while it lessens the effect if V it is less than 1. On the basis of this, a unique adjustment to the control parameter V has been proposed. The proposed method enhances SHO's exploitation.

$$V = 2R_2 - \frac{bu}{5} \quad (4.17)$$

Algorithm 4.1: ISHO

```
1: Input: Population of hyenas
2: Output: Optimal route
3: Procedure ISHO
4: Initialize parameters  $g$ ,  $W$ ,  $M$ , and  $V$ 
    $g$  – Control Parameter for Search Balance
    $W$  – Weight of Prey
    $M$  – Number of Optimal Solutions (Hyenas)
    $V$  – Spiral Shape Constant
5: Compute the fitness of each solution
6: Group of so far optimal solution =  $\vec{B}_g$ 
7: Best search agent =  $\vec{S}_g$ 
8: While ( $b < \text{max itr}$ ) do
9:   for each solution do
10:    Update the location by Eqn. (4.12)
11:   end for
12:   update  $g$ ,  $W$ ,  $M$ , and  $V$ 
13:   if current solution is better than the previous one do
14:     update  $\vec{B}_g$  and  $\vec{S}_g$ 
15:   end if
16:    $b = b + 1$ 
17: end while
18: return  $\vec{S}_g$ 
19: end procedure
```

4.3 Implementation and Results

In the following sections, we provide an in-depth examination of the experimental setup, present a comparative analysis of the BONY-ISHO protocol alongside existing methods, and highlight noteworthy insights acquired from our simulations.

4.3.1 Experimental Setup

To ensure the reliability of our results, we designed a comprehensive experimental setup. We utilized Python 3.9 as the primary programming language for implementing the algorithms and conducting simulations. Python was chosen for its efficiency in handling complex algorithms and simulations. The entire BONY-ISHO protocol experiments and simulations were coded in Python 3.9. Within the BONY-ISHO protocol, Ethereum blockchain technology played a crucial role in ensuring security and authentication. Smart contracts and distributed ledger capabilities of Ethereum were seamlessly integrated into the protocol to guarantee node and data authentication. The experimental environment was configured with specific hardware and software specifications to facilitate rigorous testing. We utilized Ubuntu 16.04 LTS (Linux) as the underlying operating system, providing a stable foundation for our

experiments. The central processing unit (CPU) employed for running simulations and experiments was an Intel(R) Core (TM)-i5-10210U CPU, operating within the range of 1.60GHz to 2.11GHz. To meet the computational demands of our simulations, the system was equipped with 8.00GB of RAM, ensuring efficient and reliable execution of the proposed BONY-ISHO protocol. The key parameters and characteristics considered during experimentation are provided in Table 4.1.

Table 4.1: Parameter settings

Parameters	Value
Compared methods	HSRM [166], IPF [167], EERAM [101], NFIS [168], DB-SDN [169], ISEC [170], and TSS [171]
Sensing area	$800m \times 800m$
Position of the BS	(500,800)
Sensors in the network	100-500
Size of a data packet	512 bytes
Initial energy of a node	10 J
Malicious nodes	10
Node's communication range	30m
ϵ_{mp}	$0.0013 \text{ pJ/bit/m}^2$
E_{elec}	100 nJ/bit
ϵ_{fs}	10 pJ/bit/m^2
Receiving and Transmission energy	0.01 J
Population	50
Maximum iteration	100
Simulation time	1000 sec

4.3.2 Comparative Analysis

The proposed technique is tested against previous research in terms of packet delivery ratio, throughput, energy usage, and latency etc. In Fig. 4.5, the TSS [171] and ISEC [170] algorithms have been shown to consume significantly less energy than previous techniques. The proposed BONY-ISHO protocol, on the other hand, has been shown to be more energy efficient. The average energy consumption of each technique at 500 number of nodes is HSRM (0.91 joules), IPF (1.02 joules), EERAM (1.13 joules), NFIS (1.24 joules), DB-SDN (1.35 joules), ISEC (1.46 joules), TSS (1.57 joules), and Proposed BONY-ISHO (1.68 joules). The graph shows that the average energy consumption of all techniques increases as the number of nodes increases. However, the proposed BONY-ISHO technique has the lowest average energy consumption at all number of nodes. The reason for low energy consumption is that the proposed BONY-ISHO protocol elects the CH and optimal route based on the nodes that have the highest residual energy.

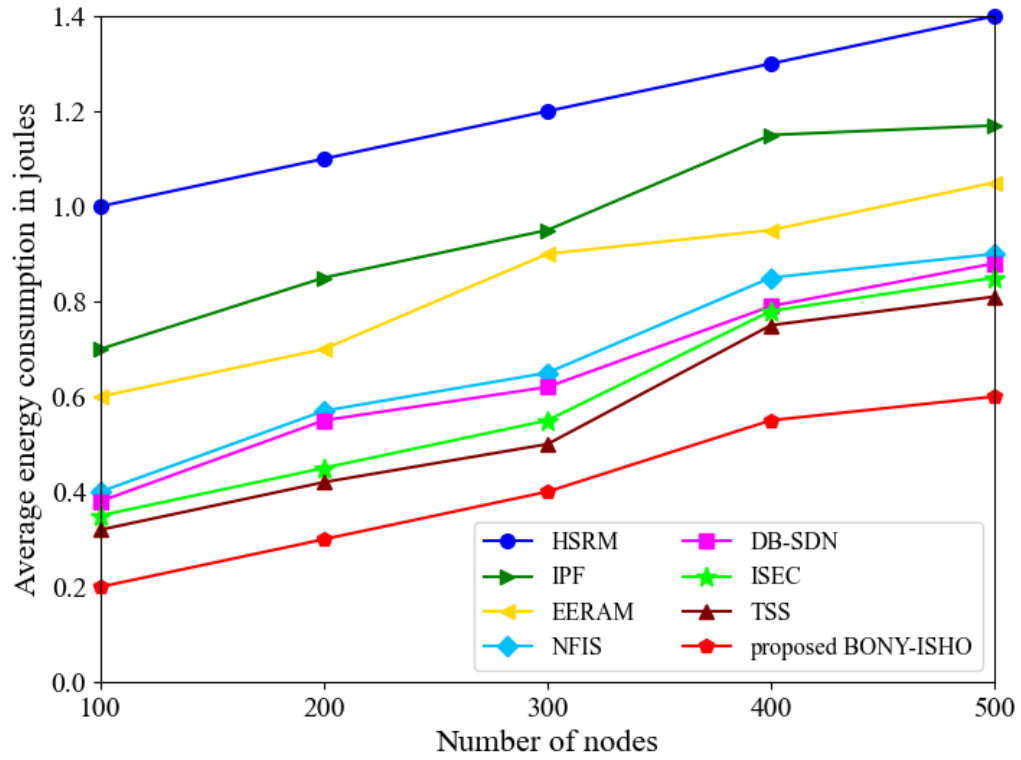


Figure 4.5: Energy consumption

The delay analysis of the proposed BONY-ISHO protocol is shown in Fig. 4.6 for different node counts. The proposed BONY-ISHO protocol has shown to have a minimum delay of 0.068 sec under a node count of 500, whereas the HSRM [166], IPF [167], EERAM [101], and NFIS [168] algorithms have a longer delay of 0.098 sec, 0.090 sec, 0.087 sec, 0.085 sec rounds, respectively. Since the proposed BONY-ISHO protocol delivers the data packet through the shortest path available, the delay is reduced.

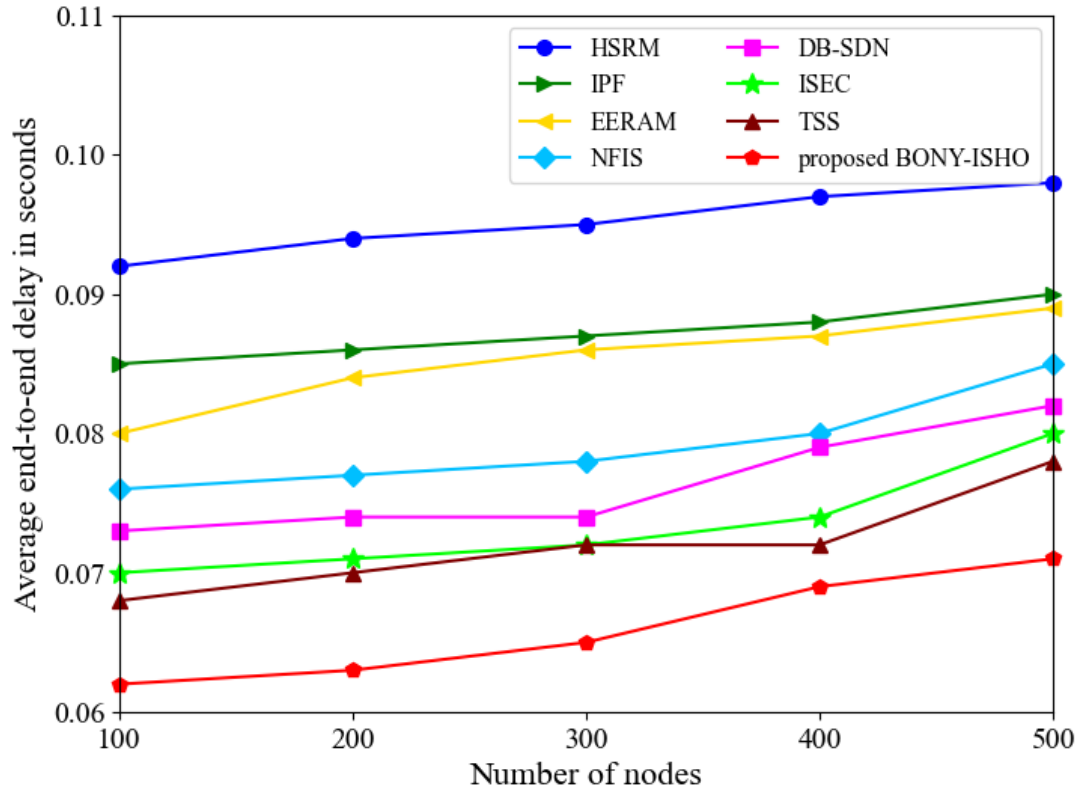


Figure 4.6: End-to-end delay

The throughput analysis of the proposed BONY-ISHO protocol under various node counts is shown in Fig. 4.7. The figure depicted that the HSRM [166] technique has the lowest throughput compared to other methods. The proposed BONY-ISHO protocol, on the other hand, outperformed all of the other protocols and achieved the highest throughput value of 45,000 kbps under a node count of 500, whereas the NFIS [168], EERAM [101], IPF [167], HSRM[166], DB-SDN [169], ISEC [170], and TSS [171] models have minimum throughput values of 41,000 kbps, 37,000 kbps, 36,500 kbps, 30,000 kbps, 41,140 kbps, 41,740 kbps, and 42,000 kbps respectively. In the proposed BONY-ISHO protocol, the data is transmitted through the nodes that have the best link quality, which increases the throughput.

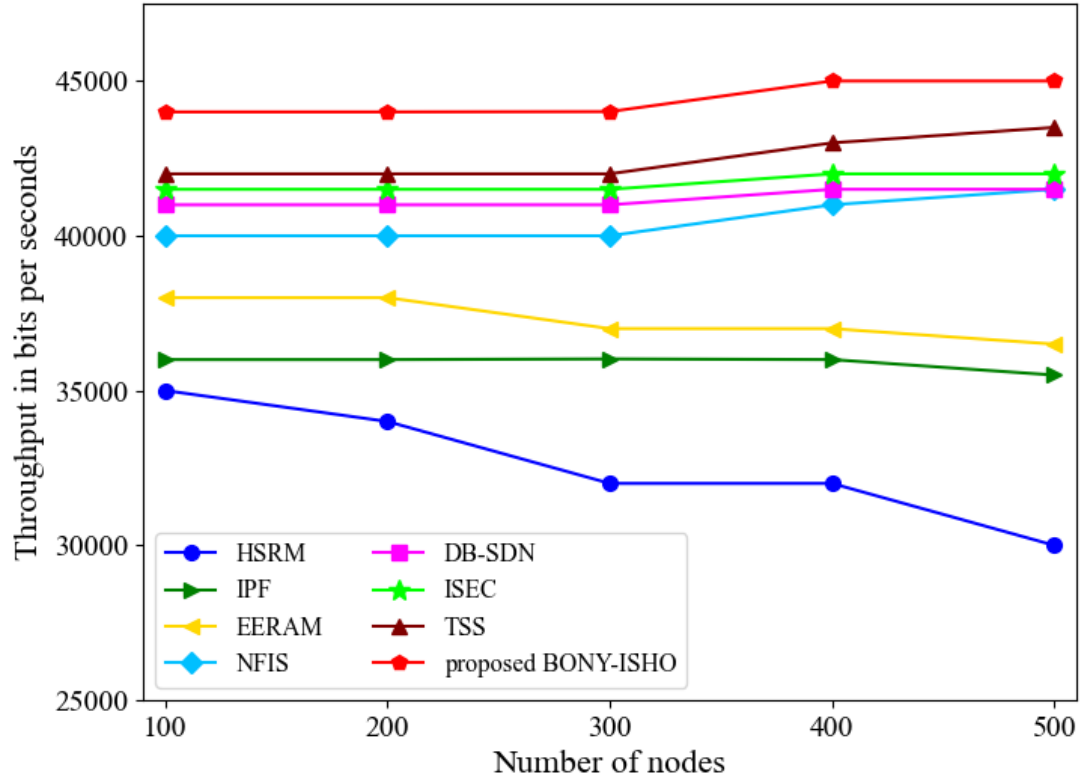


Figure 4.7: Throughput analysis

The packet delivery rate (PDR) analysis of the proposed BONY-ISHO protocol under changing node counts is shown in Fig. 4.8. According to the graph, the HSRM [166] algorithm is an ineffectual strategy with a lower PDR than other techniques. The proposed BONY-ISHO protocol surpassed all other techniques and achieved a higher PDR value. The proposed BONY-ISHO protocol has a maximum PDR of 99.8 percent at a node count of 100, whereas the HSRM [166], IPF [167], EERAM [101], NFIS [168], ISEC [170], TSS [171], and DB-SDN [169] algorithms have lower PDR of 97 percent, 97 percent, 95.8 percent, 96.75 percent, 97.8 percent, 98 percent, and 97.5 percent, respectively. Since the malicious nodes are eliminated by secure clustering, the PDR is increased for the proposed BONY-ISHO protocol.

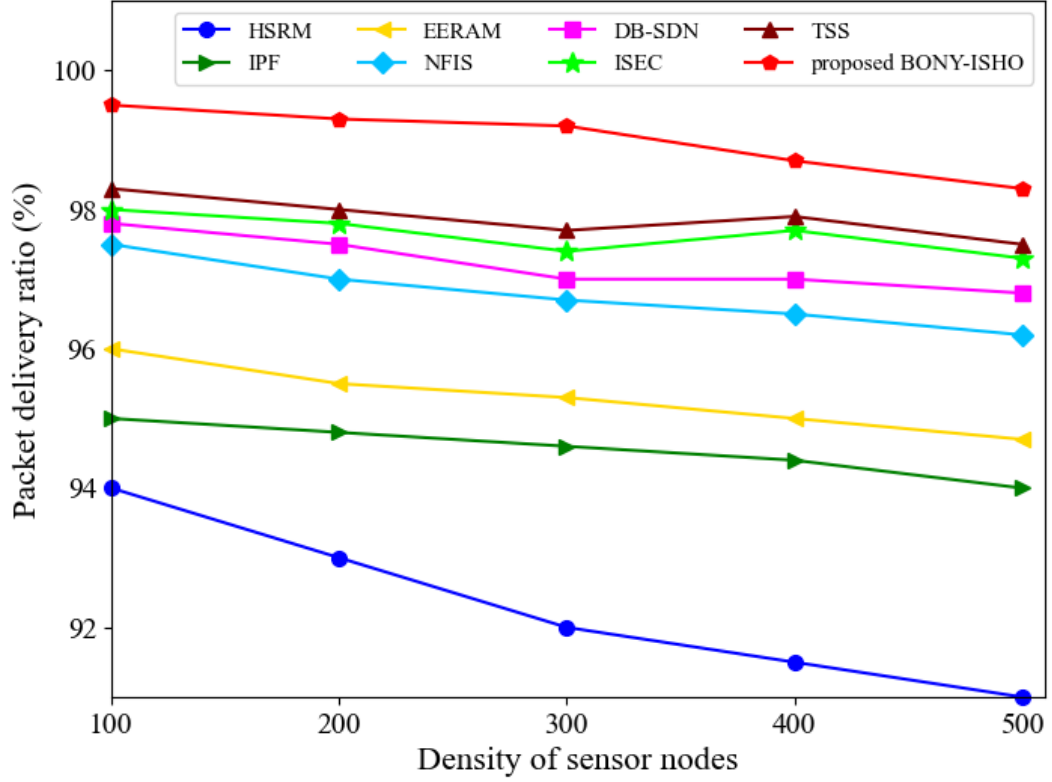


Figure 4.8: Packet Delivery Ratio analysis

According to Fig. 4.9, the proposed BONY-ISHO protocol has a greater number of living nodes than the other protocols, which had a smaller number of alive nodes. After 400 rounds, the proposed BONY-ISHO protocol has 500 alive nodes, whereas the existing methods such as NFIS [168], EERAM [101], IPF [167], HSRM [166], ISEC [170], TSS [171], and DB-SDN [169] have 480, 450, 400, 390, 490, 488 and 485 alive nodes, respectively. The proposed BONY-ISHO protocol has more alive nodes than existing techniques, even with a 3600 increase in rounds. Since the proposed BONY-ISHO protocol authenticates each node in the network by a blockchain based authentication mechanism, the network can quickly identify and eliminate malicious nodes. This increases the packet delivery ratio.

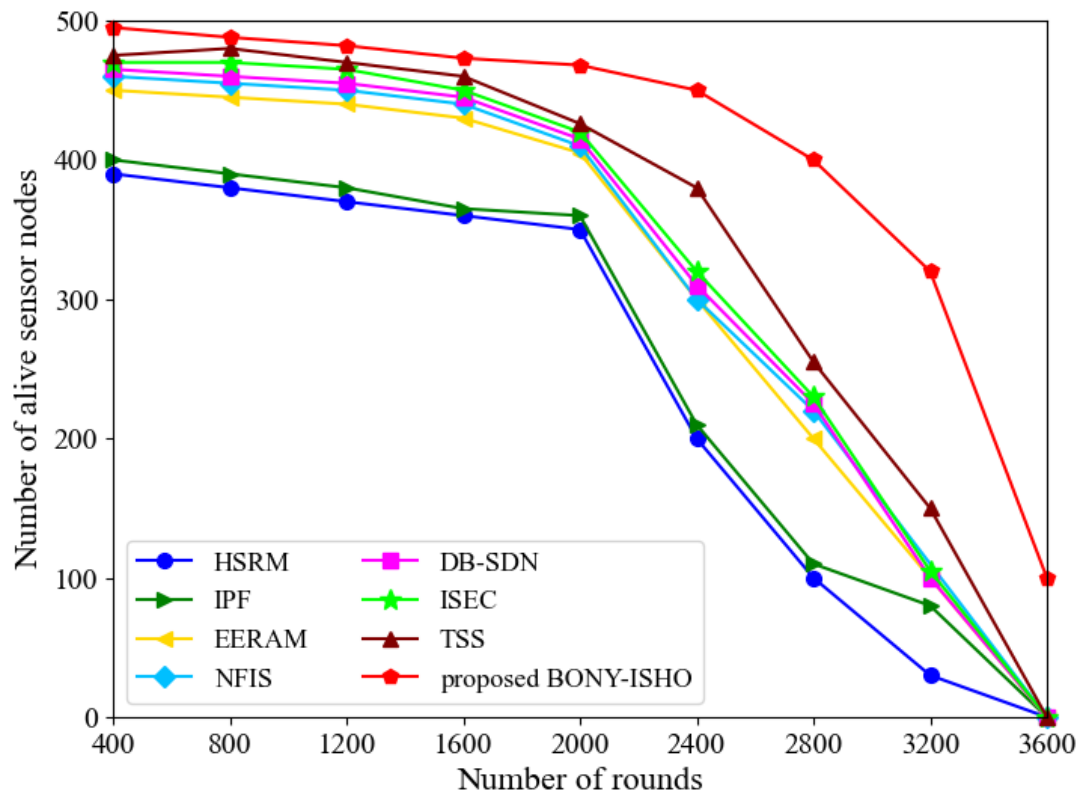


Figure 4.9: Alive sensor nodes analysis

The encryption and decryption time comparison is shown in Fig. 4.10 and Fig. 4.11, respectively.

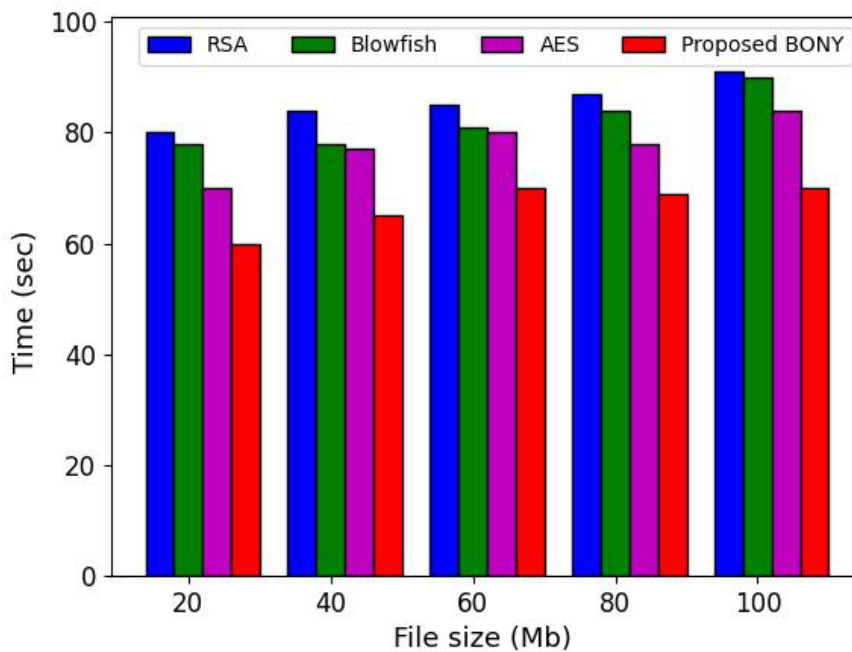


Figure 4.10: Encryption time analysis

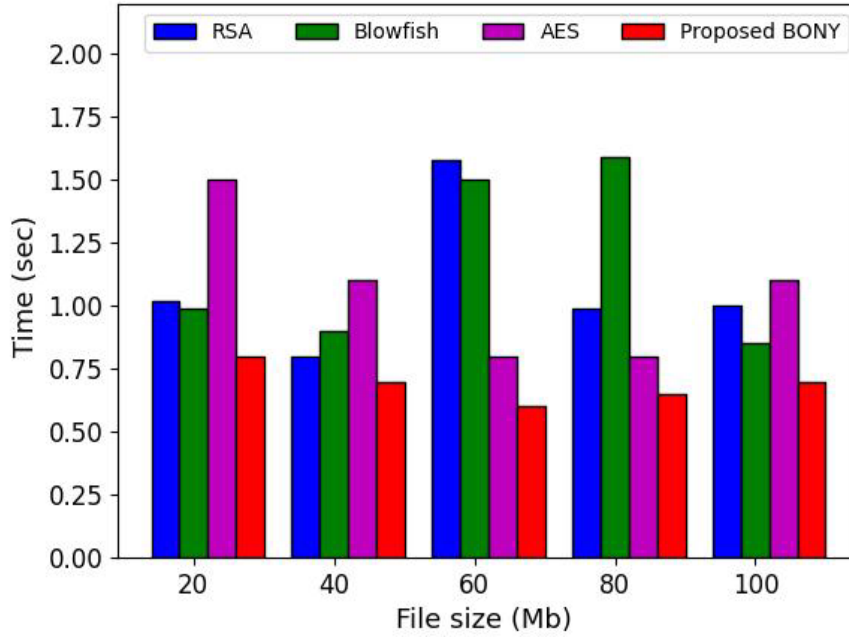


Figure 4.11: Decryption time analysis

For this comparison, the OTP technique and secure WSN-IoT are used. The key length and randomization maintain the proposed BONY algorithm's time consumption and security level. Rivest–Shamir–Adleman (RSA) is utilized for encryption in secure WSN-IoT, although it is not ideal for resource restricted environments. When the file size is 20Mb, the RSA takes 77 seconds to encrypt the data, whereas the blowfish approach takes 78 seconds, and the AES algorithm takes 70 seconds. The proposed BONY encryption algorithm technique takes just 58 seconds for the same file size. Similarly, the decryption time of the proposed BONY algorithm is also less compared to other existing algorithms.

4.3.3 Computational complexity

In the following evaluation, we present a comparative analysis of various methods, including HSRM [166], IPF [101], [167], EERAM [101], NFIS [168], ISEC [170], TSS [171], and DB-SDN [169], alongside our proposed BONY-ISHO method. We examine their performance based on key metrics such as operations (GFLOPS), model parameters, inference time, memory space, and execution time. This analysis aims to provide insights into the effectiveness and efficiency of our proposed BONY-ISHO method in comparison to existing techniques.

Table 4.2: Comparison table

Methods	Operations (GFLOPS)	Model parameters	Execution time (seconds)	Model size (MB)	Inference time (milliseconds)
HSRM [166]	974.75	62,985,137	0.634	375.47	95.82
IPF [167]	883.7	60,894,673	0.473	328.43	93.54
EERAM [101]	894.74	58,673,917	0.456	256.42	87.86
NFIS [168]	486.85	54,742,702	0.433	193.67	86.85
DB-SDN [169]	271.8	16,375,713	0.452	165.32	84.42
ISEC [170]	365.21	24,673,347	0.674	158.23	78.54
TSS [171]	276.94	34,964,643	0.643	140.76	73.2
proposed BONY- ISHO	123.65	6,347,214	0.195	30.53	19.14

The proposed protocol demonstrates outstanding performance in terms of Giga Floating Point Operations per Second (GFLOPS), achieving 123.65 GFLOPS. This exceptional performance can be attributed to the optimized utilization of the Blowfish-Honey cryptographic technique for secure data transmission and the integration of the ISHO algorithm, which intelligently routes data, minimizing computational overhead. With a model parameter count of 6,347,214, our protocol strikes a balance between complexity and efficiency, ensuring robust functionality without excessive computational burden. Its execution time is swift at 0.195 seconds, thanks to meticulous algorithm design. Furthermore, the conservative model size of 30.53 Megabytes (MB) and an inference time of 19.14 milliseconds enhance its applicability on resource-constrained IoT devices. These factors collectively position the proposed BONY-ISHO protocol as a superior choice for IoT-based Wireless Sensor Networks, excelling in computational efficiency without compromising performance.

4.4 Chapter Summary

This chapter proposed a secure and energy-efficient BONY-ISHO routing technique. BONY-ISHO protocol had four stages, namely initialization, clustering, encryption, and optimal route selection. In the first stage, known as the initialization stage, the Ethernet address of the nodes was hashed to get a unique identity for each node. The nodes are first initialized by storing the ID in the public and private blockchain. In the second stage, i.e., the clustering stage, the nodes were authenticated and clustering was made more secured by a smart contract mechanism. In order to

secure the data, the BONY encryption technique was applied in the third stage before transmitting the data. Since the advantages of the blowfish and honey algorithm were effectively utilized, any intrusion was extremely unlikely and BONY encryption effectively secured data from unwanted access. Finally, in the last stage, the data was transmitted to the destination through an optimal route by using the ISHO algorithm. This conserved the network energy and extended the lifetime of the network. The performance of the proposed protocol provides better performance than the existing techniques in terms of network lifetime, throughput, energy consumption, PDR, delay, number of alive nodes, encryption time, and decryption time.

CHAPTER 5

MULTI-LEVEL TRUST BASED SECURE AND OPTIMAL IoT-WSN ROUTING FOR ENVIRONMENTAL MONITORING APPLICATIONS

The IoT is an expanding network of devices that are connected to gather and share data in real-time. With the evolution of unified communication, there has been a significant rise in the use of devices connected to the IoT [172] [173] [174]. Wireless Sensor Networks (WSNs) use low-cost, smart sensors to improve data collection capabilities as the Internet of Things (IoT) expands. In an IoT system, sensors broadcast their data directly to the internet, but in WSNs, sensors are connected to a central node or a router [175]. To obtain data from WSN, an IoT system can establish a connection with the WSN's router. The gateway offers LoRaWAN wireless connectivity to the sensor network over a range of up to 15 kilometers. Low power, short distance radios that comply with IEEE 802.15.4 are often utilized in several IoT applications [176]. IoT applications are being developed for a wide range of fields, like the transportation sectors, smart cities, the energy and medical sectors, military, and agricultural monitoring, forest monitoring, etc. [166] [177].

5.1 Introduction

The collaboration between IoT and WSNs has brought about transformative advancements in various sectors. By connecting IoT systems to a WSN, farmers gain access to real-time data regarding soil moisture, weather conditions, and crop health, thus enabling precision irrigation and enhancing crop yields [178]. Similarly, in the healthcare sector, IoT devices integrated with WSN technology offer continuous monitoring of patients' vital signs, transmitting crucial data to healthcare providers for remote patient care, and timely medical interventions. IoT-WSN in healthcare enables wearables to send patient data to doctors, improving remote care and early diagnosis [179]. Moreover, the energy sector benefits from IoT-WSN applications for smart grid management, facilitating the monitoring and optimization of energy distribution [180]. These concrete examples illustrate how the fusion of IoT and WSN technologies drives efficiencies, optimizes resource utilization.

However, it is essential to acknowledge that this synergy between IoT and WSNs brings about new challenges, particularly in terms of security. The security of IoT-based WSNs is critical to ensure the integrity and reliability of data, as well as the proper functioning of applications. These security challenges encompass various threats, such as unauthorized access, data breaches, and network disruptions [166]. In

precision agriculture, for instance, unauthorized access to IoT-WSN systems can lead to inaccurate data collection, potentially resulting in suboptimal crop yields or even crop losses. Furthermore, in healthcare, the compromise of patient health data transmitted through IoT-WSN technology poses a significant risk to patient care and privacy. These specific examples underscore the gravity of security risks in IoT-based WSN applications, emphasizing the need for robust security measures.

Rosewood, pine, teak wood, and sandalwood are a few of the precious trees that have seen a dramatic rise in poaching as a result of man's excessive desire to satisfy his desires [181]. Moreover, Rhino poaching is a serious problem in India. The horn may range in length from around 20 centimeters to about 60 centimeters; it can cost as much as \$300,000 due to its high demand in the treatment of diseases, including cancer and hangovers. Most of the world's population of the Indian one-horned rhinoceros reside in the protected areas of Assam, particularly in Kaziranga National Park, Orang National Park, etc. The nails, skin, and horns of rhinos are all highly prized because of its usage in traditional Vietnamese and Chinese medicine [182]. Moreover, international boundaries may not have nearby settlements, patrolling troops, or barriers in challenging terrain such as forests [183] [184]. Therefore, IoT-based WSNs provide a solution for real-time environmental monitoring. However, these networks are prone to various security threats by illegal intruders that may affect the national security [185].

Recently, Intrusion Detection System (IDS) based methods [186], Machine learning methods (ML) [187], Trust based methods [188], and cryptographic techniques [189] have been proposed to minimize the routing attacks. However, cryptographic methods cannot guarantee complete network security for sensor nodes. By comparing nodes' behaviors against a set of criteria, trust may be established between them. In recent years, researchers focus towards trust-based methods because of its easy implementation and less complexity [160]. Several trust formation systems have been suggested in different industries, including web-based services, e-commerce, and WSNs. In WSNs, trust is calculated regularly depending upon the number of occurrences of good and bad conduct measured within a certain span of time and using a specific mechanism [190]. Therefore, this chapter proposes a multi-level trust based optimal routing technique for IoT based WSN applications. The following are main contributions in this chapter:

- The proposed ML-HSOR uses Markov-based adaptive clustering for CH selection and dynamically adjusts attribute weights in real-time based on network conditions. This adaptive mechanism continuously evaluates sensor nodes and their attributes, ensuring that the most suitable nodes are appointed as CHs. This adaptability enhances network lifespan, performance, and CH selection, making the methodology more resilient and dynamic compared to traditional Markov models.
- ML-HSOR addresses the problem of malicious nodes that cause various attacks, such as garnishing and bad mouthing, by introducing a multi-level hierarchical trust evaluation approach. Trust is evaluated at both intra-cluster and inter-cluster levels, considering factors such as interaction trust, data trust, validation trust, transmission trust, and identity trust. This approach significantly enhances security and reliability,

a novel contribution that safeguards environmental monitoring systems.

- In the context of garnishing attacks, a time-window mechanism is introduced to observe the behavior of nodes over specific periods. This approach monitors the number of successful and unsuccessful interactions between nodes and uses this information to assess trustworthiness. The time-window mechanism contributes to effectively countering dynamic threats and maintaining system integrity, a novel feature not typically found in existing methodologies.
- To prevent replay attacks and eavesdropping, timestamps are added to each message, and data is encrypted using the Improved Blowfish Algorithm (IBFA). This ensures data integrity and confidentiality, particularly in forest areas where sensitive information needs protection.
- The congestion is reduced and the optimal path is determined using PL-COA. The performance of COA is improved by introducing polarity learning and Levy flight strategy.
- Polarity-Based Learning enable solutions to be found efficiently by examining opposite search spaces. This novel feature helps prevent local optima entrapment and enhances the algorithm's exploration capabilities, a unique contribution that is not commonly found in traditional meta-heuristic algorithms.
- Levy flights facilitating comprehensive data surveying and navigation in complex and high-dimensional environmental data spaces. This feature enables the algorithm to avoid local optima and discover better solutions, contributing to more robust environmental monitoring in dynamic and evolving conditions.

The remaining sections of the chapter are organized as follows: Section 5.2 introduces the proposed ML-HSOR protocol, Section 5.3, provides the comparative result analysis, and finally, Section 5.4 concludes with chapter summary.

5.2 Proposed ML-HSOR Methodology

In our proposed ML-HSOR methodology, we assume that the Base Station (BS) does not face resource limitations and remains secure from potential attackers. As the central command authority, the BS has the ability to eliminate malicious nodes and replace them with trustworthy ones to ensure stable operation. This assumption is grounded in practical considerations where the BS typically possesses greater computational resources and security measures compared to individual sensor nodes in WSNs. This distinction allows the BS to perform resource-intensive tasks and security functions. In practice, the BS often serves as the network's control center, enabling measures such as physical isolation, security protocols, and secure communication channels. These safeguards protect the BS and align with real-world practices to ensure its resource adequacy and security, thereby empowering it as the central command authority within our methodology to enhance network stability and security. Fig. 5.1 illustrates the architecture of the proposed ML-HSOR methodology. It consists of perception, networking, and application layer. First, all the nodes in the

network are registered, and then clustering is performed using the Markov model. The aggregated data is encrypted and transmitted via a secure and optimal path using the PL-COA algorithm. Furthermore, the assumption of the BS's security extends to its resilience against sophisticated attacks, requiring the implementation of anomaly detection systems, intrusion prevention measures, and constant vigilance. The complexity is heightened by the need for the BS to dynamically adjust its security protocols based on evolving threat landscapes, ensuring a proactive defense mechanism against emerging vulnerabilities. The proposed algorithm's complexity is further compounded by the interplay between the perception, networking, and application layers. The registration and clustering of nodes, encryption of aggregated data, and transmission through a secure path using the PL-COA algorithm demand meticulous coordination and synchronization. Achieving optimal performance while maintaining security adds layers of intricacy, necessitating a carefully crafted algorithmic framework. The BS dynamically allocates resources that perfectly complements the adaptability built into our algorithm. The ML-HSOR model excels in real-time resource optimization, intelligently responding to changing network conditions. This synergy between the assumed resource allocation capabilities of the BS and our algorithm's adaptability enhances the overall responsiveness and efficiency of the WSN.

Real-time decision-making stands as a core strength of our proposed model. The BS, as the central command authority, capitalizes on the strengths of our algorithm to make instantaneous decisions. The energy efficiency for the BS mirrors the energy-saving strategies embedded in our algorithm. The ML-HSOR model optimizes communication protocols, manages power consumption, and employs sleep-wake strategies for sensor nodes, all contributing to the sustainable operation of the network. Scalability is a prominent strength of our algorithm, and the assumption that the BS accommodates network growth seamlessly reinforces this. The ML-HSOR model scales effectively with an increasing number of nodes, maintaining optimal performance.

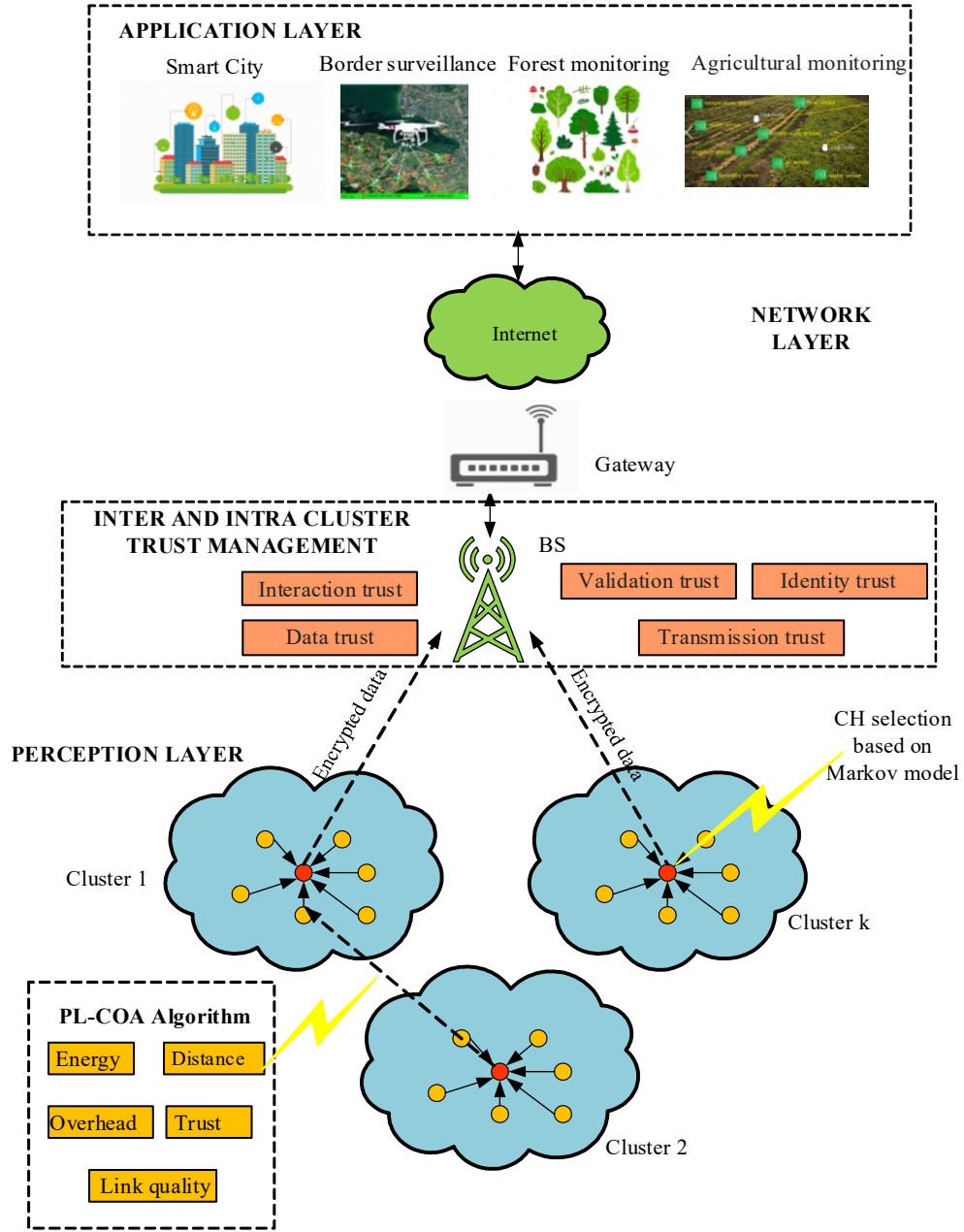


Figure 5.1: The architecture of the proposed ML-HSOR methodology

5.2.1 Registration

The new sensor node that is added to the network is first registered with the BS using a unique identity. To enhance the security of the system, each node is assigned a distinct identity such as labels or hash values using a modified SHA-1 and SHA-2 algorithm [191]. The unique identities of the sensor nodes (SN) are generated by using Eqn. (5.1).

$$U_{ID} = ((Ky + 1) \oplus N_{Rd} \| H(Ky + 1) \oplus ID \| N_{Rd}) \quad (5.1)$$

where, K_y represents the key needed for the hash function, N_{Rd} denotes the random number generated by the BS, and ID denote the serial number. By using this labeling approach, communication becomes more efficient, and the system is protected against various external attacks, such as the spoofing and whitewashing attacks.

5.2.2 Clustering

In our clustering approach, we uphold the core principles of the Markov model [192] for CH selection. This model is the fundamental of our methodology, enabling the meticulous selection of secure and energy-efficient CHs. However, what sets our approach apart is the introduction of a pioneering feature: the ability to adaptively adjust the weighting of attributes in real-time, based on the network's current conditions. This adaptive weighting mechanism performs a continuous evaluation of sensor nodes and their attributes, ensuring that the most suitable nodes are appointed as CHs. This adaptability grants us a unique advantage. For instance, if particular nodes consistently demonstrate exceptional energy efficiency, our model may assign a higher weight to energy-related attributes for CH selection. This dynamic responsiveness allows our network to swiftly adapt to variations in energy levels and node behavior, ultimately leading to more precise CH selection. The innovative essence of our method lies in this adaptability, transforming it from a static model into a dynamic and resilient solution, capable of maintaining optimal performance throughout its operational lifetime. This dynamic and adaptive CH selection mechanism significantly boosts network lifespan and performance, all while optimizing the distribution of responsibilities among CHs. By integrating this novel feature, our methodology ensures efficient operation and longevity, rendering it a robust choice for WSNs in evolving environments. While the Markov model naturally adapts to changes in energy levels and node behavior over discrete time slots, the adaptive adjustment we introduce serves to fine-tune the model's parameters continuously in real-time. This means the Markov model operates in a step-wise fashion, periodically reassessing and reselecting CHs based on discrete time slots. In contrast, our adaptive adjustment is more continuous and responsive, enabling real-time fine-tuning of model parameters to accurately mirror the immediate state of the network. The goal of introducing adaptive adjustment is to enhance CH selection responsiveness to the network's real-time condition. It accomplishes this by continually monitoring and adjusting parameters, a capability not captured by the Markov models discrete time slots. This added adaptability further optimizes CH selection and elevates overall network performance.

For example, consider a network with 100 sensor nodes. Our methodology employs the Markov model, a dynamic and adaptive approach, to continuously monitor the energy levels and behaviors of these sensor nodes in real-time. This model updates the network state as it evaluates each node for the role of CH. For instance, at the beginning of the observation period, in time slot 1, node 50 exhibits the highest energy level. The Markov model, recognizing this, selects node 50 as the CH for that specific time slot. However, as time progresses and the network's state evolves, the energy levels of the nodes are subject to change. In the next time slot, it may turn out that node 20 now has the highest energy level, surpassing node 50. In response to this shift in energy levels, the Markov model dynamically adjusts and selects node 20 as

the new CH. This process of continually updating the CH selection carries on, with the Markov model making real-time decisions based on changing energy levels and other relevant network conditions. The ability to select CHs dynamically and adapt to the network's immediate state ensures the efficient operation and longevity of the network, making our methodology a robust choice for WSNs in dynamic and evolving environments.

Every CH is responsible for managing the Member Nodes (MN) within its cluster and maintaining a record of their IDs and locations. The CH also maintains and updates the energy values of each node with the arrival of each message in a matrix, as given in Eqn. (5.2).

$$Mat_{MN} = \begin{bmatrix} ID_1 & PO_1 & En_1 \\ ID_2 & PO_2 & En_2 \\ \vdots & \vdots & \vdots \\ ID_{k-1} & PO_{k-1} & En_{k-1} \end{bmatrix} \quad (5.2)$$

where, ID , PO , and En denotes the identity, position, and energy of the MNs respectively. Similarly, the BS keeps a record of IDs, positions, and energy of the CHs as given in Eqn. (5.3).

$$Mat_{CH} = \begin{bmatrix} ID'_1 & PO'_1 & En'_1 \\ ID'_2 & PO'_2 & En'_2 \\ \vdots & \vdots & \vdots \\ ID'_{l-1} & PO'_{l-1} & En'_{l-1} \end{bmatrix} \quad (5.3)$$

where, ID' , PO' , and En' denotes the identity, position, and energy of the CHs respectively.

5.2.3 Authentication by multi-level hierarchical trust evaluation

Wireless sensors used in environmental monitoring applications can play a role in detecting illegal poaching in wildlife reserves and protected areas. For example, the sensors can detect any movement and trigger an alert, while infrared cameras can provide real-time images of the poacher, and sound recorders can pick up any sounds of vehicles or weapons. When a poacher enters the protected area, the wireless sensors are activated and send an alert to the park rangers. With this information, park rangers can quickly respond to the situation and prevent the illegal activity from taking place. This helps to protect wildlife and their habitats and supports conservation efforts. However, if the sensor network is disabled or otherwise compromised by the attackers, it becomes much easier for poachers to engage in illegal activities. To overcome these issues, the proposed method detects the malicious nodes that cause various attacks such as garnishing, bad mouthing, grey-hole etc., using a multi-level hierarchical trust evaluation approach as provided in Fig. 5.2. The trust of the nodes is determined at inter as well as intra cluster level. Intra-cluster trust refers to the level of trust evaluated between Cluster Members (CM) and between CH and CM, while inter-cluster trust

pertains to the trust evaluated between CH and CH, as well as between BS and CH. The proposed method computes the interaction trust, data trust, validation trust, transmission trust, and identity trust of a node directly or based on the feedback (indirect) trust obtained from the neighboring nodes.

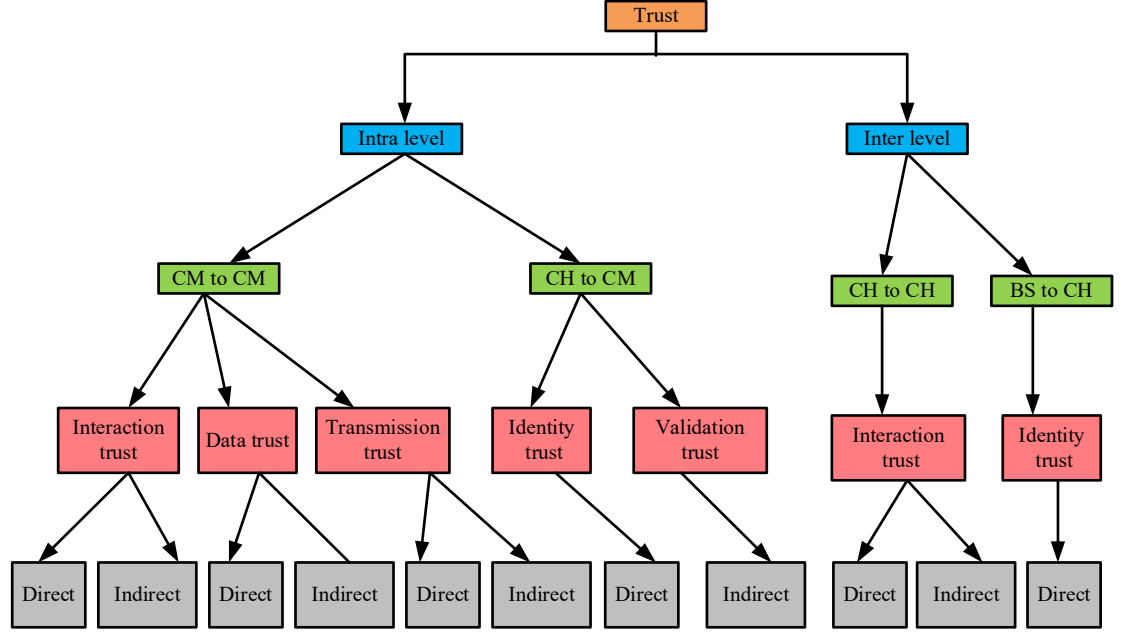


Figure 5.2: Multi-level hierarchical trust evaluation

The process depicted in Fig. 5.3 involves the interaction between CM-CM and CM-CH for cluster head selection and information transfer to the BS. In this process, the trust level of CMs is assessed using three distinct types: interaction trust, data trust, and transmission trust. Subsequently, the most secure CMs undergo identity and validation trust to pinpoint the CM with a high trust level, who then transmits information to the CH. This process involves both direct and indirect nodes. Additionally, the information gathered from various cluster heads undergoes identity trust once again to identify the reliable cluster head in the network. Once the secure CH is determined, the information is sent to the BS, which encrypts the data before forwarding it.

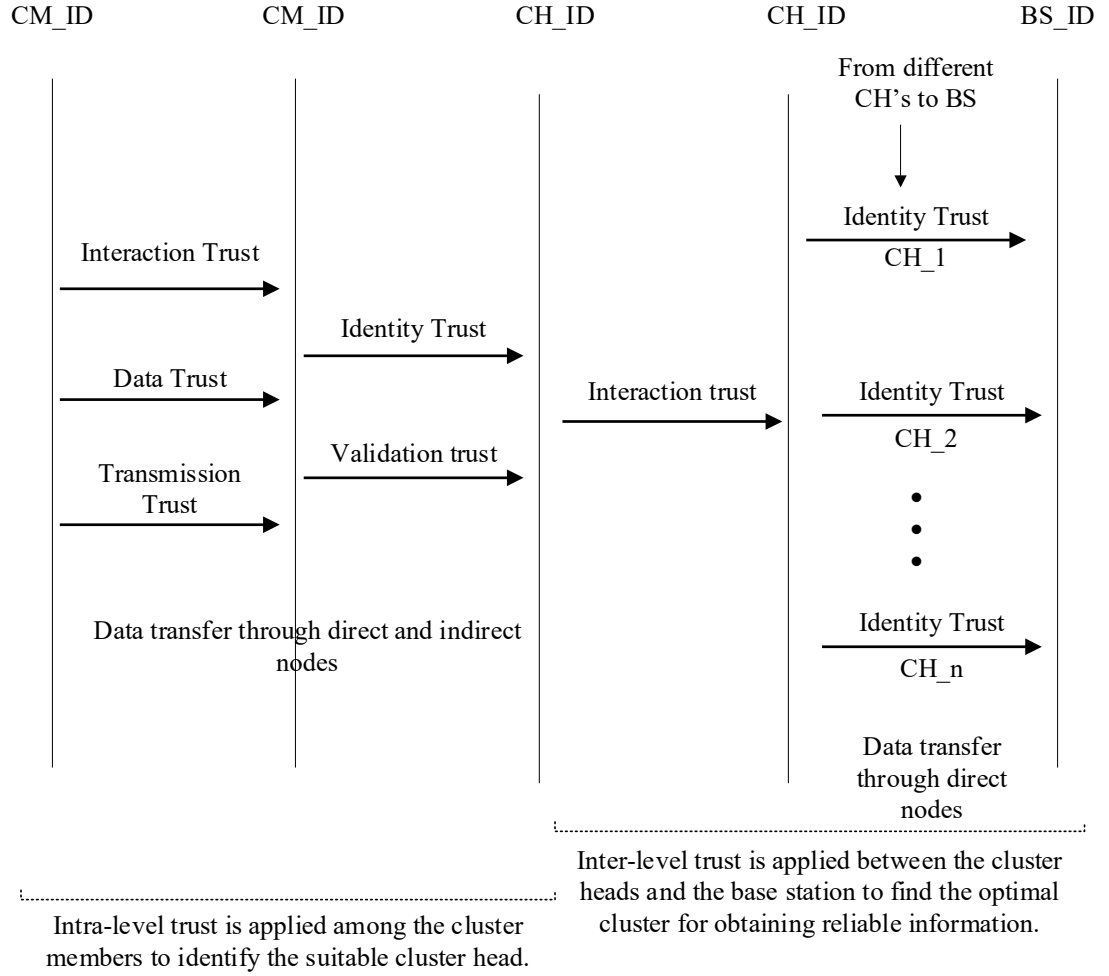


Figure 5.3: Interactions between the different nodes in the Trust evaluation

In a garnishing attack, the malicious nodes alternate their behavior between trustworthy and malicious, which aims to harm the network while remaining undetected. To address this issue, trust is computed in which, a time window mechanism, as illustrated in Fig. 5.4, is used to observe the behavior of nodes over a certain time frame.

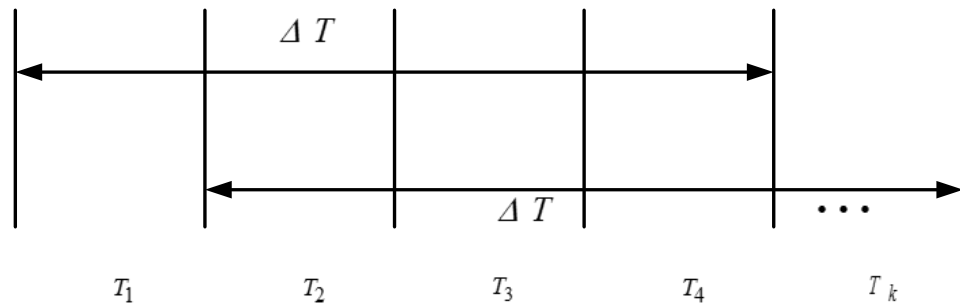


Figure 5.4: Illustration of time window

(i) Time window

Trust computation takes advantage of a time window mechanism each lasting a fixed amount of time interval. Interactions between nodes are periodically tracked and recorded for successful and unsuccessful outcomes. The time window then moves to the right and includes new interactions while discarding the oldest ones. This way, the time window provides an overview of a node's behavior over a certain period, taking into account both past and current actions.

For instance, if node A transmitted 5 successful readings and 2 unsuccessful readings in the first interval, this information is recorded. After each 5-minute interval, the time window slides to the right and includes new interactions while discarding the oldest ones. If the BS notices that a particular node has a high rate of unsuccessful transmissions during this time interval, the BS gives a penalty to reduce trust or replaces the node to ensure the reliability and security of the environmental monitoring system.

(ii) Time stamp

A replay attack in WSNs can occur when an attacker intercepts a message and later retransmits it. This kind of attack has the potential to interfere with the WSN's regular operations or grant the attacker unauthorized access. To prevent this, a timestamp is added to each message before transmission. The recipient can check whether the message is new and hasn't been replayed by looking at the timestamp, which shows the date and time the message was sent. The receiver has the option to delete messages that are too old. By discarding a message with an outdated timestamp, the receiver prevents an attacker from intercepting and later retransmitting it. Nevertheless, an attacker could alter the timestamp to an earlier time, deceiving the recipient into believing that the message was transmitted earlier than it actually was.

Moreover, eavesdropping attacks in forest areas can have various motives for attackers. They may aim to access sensitive or confidential information about environmental or conservation activities in the forest. By eavesdropping on the communication signals, they can obtain sensitive details regarding the location of valuable resources such as timber or minerals and execute their theft more efficiently. To address this issue, the aggregated data and the timestamp are encrypted before transmitting them to the receiving node using the Improved Blowfish Algorithm (IBFA) [103]. Upon reception, the recipient can decode the message and timestamp. Since the attacker does not possess the decryption key, an attacker who intercepts the communication cannot read or alter the message or timestamp. Encryption helps secure the integrity of the timestamp and aggregated data.

5.2.3.1 Intra cluster trust

The intra cluster trust is computed in two levels: CM-CM and CH-CM.

5.2.3.1.1 CM-CM level

The trust between CM-CM relies on factors such as interaction trust, data trust, and transmission trust.

(i) Interaction trust (IT)

The direct interaction trust of node A on node B is computed using Eqn. (5.4).

$$I_{A,B}(\Delta T) = \left[N_{TW} \times \left(\frac{S_{A,B}(\Delta T)}{S_{A,B}(\Delta T) + F_{A,B}(\Delta T)} * \frac{1}{\phi} \left(\frac{S_{A,B}(\Delta T)}{S_{A,B}(\Delta T) + 1} \right)^\beta \right) \right] \quad (5.4)$$

where, N_{TW} denotes the number of units in the time window, $S_{A,B}(\Delta T)$ denotes the sum of successful interaction of node A with node B during the time window ΔT , $F_{A,B}(\Delta T)$ represents the sum of failed interaction of node A with node B, ϕ and β signifies the penalty and pay factor that changes the trust value. The values of ϕ and β are tuned based on the application requirement.

Consider the case of node A which has never communicated with node B before but wants to determine whether node B can be trusted to forward a packet. In this case, the IT of node B is computed on the basis of scores given by other neighboring nodes (indirect trust) of the node B. The trust estimation accuracy is improved by utilizing feedback from neighboring nodes to determine the node B trust level. Thus, the indirect interaction trust is computed using Eqn. (5.5).

$$I'_{A,B}(\Delta T) = \frac{1}{n} \sum_{k=1}^n I_{N,B}(\Delta T) \quad (5.5)$$

where, n denotes total number of neighbors of node B, $I_{N,B}(\Delta T)$ represents the trust calculated directly by neighboring node N on node B which can be computed similar to Eqn. (5.4).

(ii) Data trust (DT)

The concept of data trust is utilized to detect the presence of malicious nodes that cause grey-hole attacks, in which the intruder drops only certain kinds of data packets to stay as hidden as possible. An example to illustrate a grey-hole attack in a WSN could be monitoring the temperature in a forest for early fire detection. A malicious node, acting as a grey-hole attacker, might drop data indicating high temperatures, making it appear as if the forest is not in danger even if there is an active fire. Failure to detect the fire promptly could result in significant damage.

Direct DT of node B is computed using Eqn. (5.6).

$$D_{A,B}(\Delta T) = \frac{DP_B(\Delta T)}{ST_A(\Delta T)} \quad (5.6)$$

where, ST_A denotes the sum of data packets sent by node A, DP_B represents the sum of packets dropped by node B.

Similarly, the indirect data trust of node B is computed using Eqn. (5.7).

$$D'_{A,B}(\Delta T) = \frac{1}{n} \sum_{k=1}^n D_{N,B}(\Delta T) \quad (5.7)$$

where, n denotes total number of neighbors of node B, $D_{N,B}(\Delta T)$ represents the trust calculated directly by neighbouring node N on node B which can be computed similar to Eqn. (5.6).

(iii) Transmission trust

A DoS attack can cause a node to become overloaded and reduce the responsiveness of the system. This causes delays in detecting environmental activities or anomalies, which can be critical in real-time monitoring applications. To prevent this, the transmission trust of a node is computed using Eqn. (5.8).

$$TT_{A,B}(\Delta T) = \frac{Pt(A,B)}{t} \quad (5.8)$$

where, $Pt(A,B)$ denotes the sum of packets transmitted from node A to node B at time t .

The indirect transmission trust is computed using Eqn. (5.9).

$$TT'_{A,B}(\Delta T) = \frac{1}{n} \sum_{k=1}^n TT_{N,B}(\Delta T) \quad (5.9)$$

where, n denotes total number of neighbors of node B, $TT_{N,B}(\Delta T)$ denotes the direct transmission trust calculated by neighboring node N on node B.

5.2.3.1.2 CH-CM level

Trust between CH-CM is determined as follows:

(i) Identity trust

A Whitewashing attack is a security threat that occurs when the system detects the malicious node and disconnects it from the network, and the malicious node then attempts to reenter the network under a different identity in order to trick the system into assigning it a new reputation score. To prevent this threat, identity trust is computed, in which each CM sends its data to its corresponding CH along with its ID for trust evaluation. During trust evaluation, the identity trust of node A is computed directly by using Eqn. (5.10).

$$ID_A = \begin{cases} 1 & \text{IF } ID_A = ID(Mat_{MN}) \\ 0 & \text{otherwise} \end{cases} \quad (5.10)$$

where, ID_A represents the identity of node A, $ID(Mat_{MN})$ represents the MN identities registered in matrix Mat_{MN} by the base station. The data is forwarded to the

base station by the CH after undergoing trust verification and aggregation.

(ii) Validation trust

The CH gathers data from the MNs and makes decisions based on the information. During these processes, the CH obtains the reputation score of the target node from other nodes in the cluster. However, a malicious node can affect the reputation of legitimate nodes by giving them a low reputation score (Bad mouthing attack). To determine if the information received is accurate, it is crucial to evaluate the reliability of the recommender. To counteract such attacks, the validity of the reputation score computed by node A is determined indirectly from other nodes using Eqn. (5.11).

$$VT_{CH,A}(\Delta T) = |R_{score(A,B)} - R_{avg(B)}| \quad (5.11)$$

where, $VT_{CH,A}(\Delta T)$ denotes the validation trust score of node A computed by the CH at the time interval ΔT . $R_{score(A,B)}$ represents the reputation score of node A given to node B. $R_{score(A,B)}$ is obtained based on other trust measures such as interaction trust, data trust, transmission trust, and identity trust. The average of all the trust measures gives the reputation score and $RS_{avg(b)}$ represents the average reputation score of node B obtained from other neighboring MNs (indirect) in the cluster.

5.2.3.2 Inter cluster trust

The degree of trust built between several clusters in a WSN is referred to as inter-cluster trust. The inter cluster trust is computed in two levels: CH-CH and BS-CH.

5.2.3.2.1 CH-CH level

It demonstrates how much a CH trusts the information provided by other CHs in the network. The ability of the CHs in various clusters to acquire and convey data accurately and reliably is indicated by the amount of inter-cluster trust between them. The direct interaction trust of CH 'Q' is computed using Eqn. (5.12). Formula 5.12 calculates the direct interaction trust of CH 'Q' in CH 'P'. This trust value represents how much CH 'Q' relies on the information provided by CH 'P'. The formula considers the accuracy and reliability of data shared by CH 'P' and CH 'Q's assessment of CH 'P's overall network activity.

$$I_{P,Q}(\Delta T) = \left[N_{TW} \times \left(\frac{S_{P,Q}(\Delta T)}{S_{P,Q}(\Delta T) + F_{P,Q}(\Delta T)} * \frac{1}{\phi} \left(\frac{S_{P,Q}(\Delta T)}{S_{P,Q}(\Delta T) + 1} \right)^\beta \right) \right] \quad (5.12)$$

where, $S_{P,Q}(\Delta T)$ denotes the number of successful interaction of CH 'P' with CH 'Q' during the time window ΔT , $F_{P,Q}(\Delta T)$ represents the number of failed interaction of

CH ‘P’ with CH ‘Q’ , ϕ and β signifies the penalty and pay factor that changes the trust value depending on the CH behavior.

The indirect interaction trust of CH ‘Q’ is computed using Eqn. (5.13).

$$I'_{P,Q}(\Delta T) = \frac{1}{n} \sum_{k=1}^n I_{M,Q}(\Delta T) \quad (5.13)$$

where, n denotes total number of neighbors of CH ‘A’, $I_{M,Q}(\Delta T)$ represents the direct interaction trust calculated by neighboring CH ‘M’ on node CH ‘Q’.

5.2.3.2.2 BS-CH level

It is essential for the BS to trust the data sent by CHs since it uses this data to make critical decisions. Therefore, the identity trust of CH ‘P’ is computed by using Eqn. (5.14).

$$ID_P(\Delta T) = \begin{cases} 1 & \text{IF } ID_P = ID(Mat_{CH}) \\ 0 & \text{otherwise} \end{cases} \quad (5.14)$$

where, ID_P represents the identity of CH ‘P’, $ID(Mat_{CH})$ represents the CH identities registered in matrix Mat_{CH} by the BS.

5.2.4 Optimal route selection

The encrypted data is then sent to the destination node via a secure and optimal path using the PL-COA algorithm. Algorithm 5.1 outlines the Optimal Route Selection process using the PL-COA. This algorithm is designed to efficiently find the best route for secure data transmission in a network of nodes. The process involves population initialization, dynamic location updates, fitness evaluation, and the utilization of Levy flights for exploration. The ultimate goal is to identify the optimal path for data transmission while considering factors like residual energy, distance, link quality, overhead, and trust.

There are four categories of chimpanzees in a group named driver chimp, barrier chimp, chaser chimp, and attacker chimp. The drivers (source node) pursue the preys (destination node) but don't try to catch them. Barriers put themselves in trees to block the way of the prey's escape route (block the malicious nodes). Chasers (intermediate nodes) move quickly to catch their prey. When the target stops moving, the attackers (node closest to the destination) attack the prey and end the hunt. The chapter describes this approach to identify the most efficient route to reach the destination node. Here, the prey is considered the destination node.

In the context of environmental monitoring, our approach introduces two significant innovations. Firstly, the utilization of Levy flights for exploration enhances the algorithm's ability to comprehensively survey the environmental parameters, making it adept at navigating intricate and multidimensional data spaces. This novel feature enables the algorithm to effectively avoid local optima and discover more optimal solutions in dynamic and complex environmental conditions. Secondly, our

algorithm incorporates Polarity-Based Learning, which empowers it to dynamically adjust its monitoring strategy based on the evolving environmental conditions and the presence of critical data patterns. By preventing the algorithm from becoming trapped in local data minima and allowing it to adapt to real-time environmental changes, these innovations significantly improve its performance in environmental monitoring applications, addressing a limitation in existing algorithms that often fail to consider these aspects and, as a result, struggle to adapt to changing environmental conditions effectively.

This algorithm computes the fitness of the solution based on trust, delay, link quality, residual energy, and distance. The residual energy, distance, and link quality metrics improve the network lifetime and QoS. The trust metric helps to protect the network against various malicious attacks, while the delay metric reduces congestion. The chimp population y_i^0 is randomly initialized at the beginning.

5.2.4.1 Fitness evaluation

The optimal path is chosen based on the nodes with maximum residual energy, minimum distance, maximum link quality, maximum residual energy, and, minimum overhead. The objective is to maximize the fitness. The objective is to maximize fitness. To optimize the system, the objective is to increase the fitness value F in Eqn. (5.15).

$$F = wt_1 E_{res} + wt_2 D(A, S) + wt_3 L_{qty} + wt_4 O_{hd} + wt_5 T \quad (5.15)$$

The weights wt_1 , wt_2 , wt_3 , wt_4 , and wt_5 are used to balance the importance of each objective function, where higher weights indicate higher priority. The values $wt_1 = 0.3$, $wt_2 = 0.1$, $wt_3 = 0.2$, $wt_4 = 0.1$, and $wt_5 = 0.3$ adjusted such that $wt_1 + wt_2 + wt_3 + wt_4 + wt_5 = 1$. E_{res} , $D(A, S)$, L_{qty} , O_{hd} , and T are the residual energy, distance, link quality, overhead, and trust.

(i) Residual energy

The residual energy of a SN is computed using Eqn. (5.16).

$$E_{res} = E_{ini} - E_{T_con} \quad (5.16)$$

where, E_{ini} is the initial energy of a node and E_{T_con} denotes the total energy consumed which can be determined using Eqn. (5.17).

$$E_{T_con} = E_{tx} + E_{rx} \quad (5.17)$$

where, E_{tx} and E_{rx} represents the energy consumed during data transmission and data reception which can be determined using Eqn. (5.55) and Eqn. (5.54).

(ii) Distance

The distance between node A and sink S is computed by Eqn. (5.18).

$$D(A,S) = \sqrt{(X'_A - X'_S)^2 + (Y'_A - Y'_S)^2} \quad (5.18)$$

where, (X'_A, Y'_A) and (X'_S, Y'_S) are the coordinates of the node A and sink S respectively.

(iii) Link quality:

The link quality of two nodes A and B is computed using Eqn. (5.19).

$$L_{qty} = \begin{cases} 1, & \text{if } dist(A,B) \leq R_{com} \\ 0, & \text{otherwise} \end{cases} \quad (5.19)$$

where, $dist(A,B)$ represents the distance between node A and B , R_{com} represents the node's communication range.

(iv) Overhead

The overhead of a node A is computed by using Eqn. (5.20)

$$O_{hd} = \begin{cases} 1, & \text{if } L_{queue}(A) \leq L_{min} \\ 0, & \text{otherwise} \end{cases} \quad (5.20)$$

where, $L_{queue}(A)$ denotes the queue length of node A and L_{min} denotes the minimum queue length.

(v) Trust

During data transmission, the trust T of a node or CH is determined based on the multi-level hierarchical trust evaluation approach provided in Section 5.2.3.

5.2.4.2 Driving and chasing the prey

The current location of the chimp is denoted as given in Eqn. (5.21).

$$Z_{chimp}(t+1) = Z_{prey}(t) - d \cdot C \quad (5.21)$$

where, $Z_{prey}(t)$ denotes the location of the prey, the parameters d and C are computed using Eqn. (5.22) and Eqn. (5.23).

$$d = 2 \cdot h \cdot s_1 - h \quad (5.22)$$

$$C = |m \cdot Z_{prey}(t) - n \cdot Z_{chimp}(t)| \quad (5.23)$$

where, s_1 and s_2 represents the random number in the range $[0, 1]$, n represents chaotic map vector, and the coefficient h is decreased from 2.5 to 0 nonlinearly, $m = 2s_2$.

5.2.4.3 Exploration

Exploration involves searching the entire search space to find new solutions. It is assumed that the attacker is in the same location as the prey. The attacker's location could then be used to change the location of the driver, barrier, and chaser. The other chimp's locations are updated to the best chimp's locations as given in Eqn. (5.24) to Eqn. (5.28).

$$Z_1 = Z_A(t) - d_1 \cdot C_A \quad (5.24)$$

$$Z_2 = Z_B(t) - d_2 \cdot C_B \quad (5.25)$$

$$Z_3 = Z_C(t) - d_3 \cdot C_C \quad (5.26)$$

$$Z_4 = Z_D(t) - d_4 \cdot C_D \quad (5.27)$$

$$Z(t+1) = \frac{Z_1 + Z_2 + Z_3 + Z_4}{4} \quad (5.28)$$

where, t represents the present iteration, $Z_A(t)$, $Z_B(t)$, $Z_C(t)$, and $Z_D(t)$ represents the attacker chimp, barrier chimp, chaser chimp, and driver chimp's location respectively. The value of d for various chimps can be computed similar to Eqn. (22) and C can be computed using Eqn. (5.29) to Eqn. (5.32).

$$C_A = |m_1 \cdot Z_A(t) - n_1 \cdot Z(t)| \quad (5.29)$$

$$C_B = |m_2 \cdot Z_B(t) - n_2 \cdot Z(t)| \quad (5.30)$$

$$C_C = |m_3 \cdot Z_C(t) - n_3 \cdot Z(t)| \quad (5.31)$$

$$C_D = |m_4 \cdot Z_D(t) - n_4 \cdot Z(t)| \quad (5.32)$$

5.2.4.4 Exploitation

Utilizing information from past iterations, exploitation improves the exploration of potential solutions. In addition to the d and m parameters specified in the exploration phase; an additional parameter introduced in the exploitation phase determines how to prevent local minima trapping. If $|d|$ is higher than one, the chimpanzees are pushed to depart from the prey (preventing local optima entrapment), but if $|d|$ is less than one, the chimps are driven to converge at global optima. The following Eqn. (5.33) models the updating process in this technique.

$$Z_{chimp}(t+1) = \begin{cases} Z_{prey}(t) - d \cdot C & \mu < 0.5 \\ chaotic\ value & \mu > 0.5 \end{cases} \quad (5.33)$$

where, μ denotes random number in $[0, 1]$.

5.2.4.5 Polarity based learning

In general, when the initial solutions in meta-heuristic algorithms are closer to the optimal location, convergence occurs quickly; else, slow convergence occurs. Polarity learning finds the optimal solution by looking at the opposite search space. The polarity learning then selects the best path among all possible paths.

(i) Polarity Number

The following can be used to identify a polarity-based number. Let's say y_0 is a real number that falls within an interval $y_0 \in [p, q]$, the opposite number of y_0 is determined by Eqn. (5.34).

$$\bar{y}_0 \in p + q - y_0 \quad (5.34)$$

Similarly, the opposite point is given by Eqn. (5.35) and Eqn. (5.36).

$$y = y_1, y_2, \dots, y_D \quad (5.35)$$

$$\bar{y} = \bar{y}_1, \bar{y}_2, \dots, \bar{y}_D \quad (5.36)$$

Eqn. (5.37) is used to calculate the values of each item in \bar{y} .

$$\bar{y}_l = p_l + q_l - y_l \quad (5.37)$$

where, $l = 1, 2, 3, \dots, D$. If $f(y_0)$ is greater than $f(\bar{y}_0)$, then y_0 remains unchanged; else, $y_0 = \bar{y}_0$; thus, population solutions are updated depending upon the better value of y_0 and \bar{y}_0 .

(ii) Levy flights (LF)

LF is focused on a heavy-tailed probability distribution for random walks to explore the search space efficiently using step length and potentially find better solutions. Traditional random walks tend to get trapped in local optima, but Levy flight can make larger jumps and explore farther away regions of the search space, increasing the chance of finding better solutions. The Levy distribution computed using Eqn. (5.38) is used to identify a probability function, from which the step sizes of LF are calculated.

$$L(y_i) = |y_i|^{1-\gamma} \quad (5.38)$$

where, y_i denotes the flight length, $1 < \gamma \leq 2$ signifies the power law exponent. The Levy probability density in integral form is determined using Eqn. (5.39).

$$f_L(y; \gamma, \alpha) = \frac{1}{\pi} \int_0^{\infty} \exp(-\alpha q^\gamma) \cos(qy) dq \quad (5.39)$$

where, α denotes the scale unit, γ signifies the distribution index. When y has a large value, Eqn. (5.39) typically require series expansion as given in Eqn. (5.40).

$$f_L(y; \gamma, \alpha) \approx \frac{\alpha \Gamma(1 + \gamma) \sin\left(\frac{\pi\gamma}{2}\right)}{\pi y^{(1+\gamma)}}, y \rightarrow \infty \quad (5.40)$$

where, Γ signifies gamma function. Random numbers are produced using the Magneta technique based on the Levy distribution as given in Eqn. (5.41).

$$levy(\gamma) = 0.05 \times \frac{y}{|z|^{1/\gamma}} \quad (5.41)$$

where, y and z represents two normal distribution variables computed using Eqn. (5.42) and Eqn. (5.43) σ_y and σ_z represents the standard deviation of y and z .

$$y = Normal(0, \sigma_y^2) \quad (5.42)$$

$$z = Normal(0, \sigma_z^2) \quad (5.43)$$

$$\sigma_y = \left[\frac{\Gamma(1 + \gamma) \sin\left(\frac{\pi\gamma}{2}\right)}{\Gamma\left(\frac{1 + \gamma}{2}\right) \gamma 2^{\frac{\gamma-1}{2}}} \right]^{1/\gamma}, \sigma_z = 1, \gamma = 1.5 \quad (5.44)$$

The new location after applying LF becomes,

$$\bar{C}_A = levy(\gamma) \otimes |m_1 \cdot Z_A(t) - n_1 \cdot Z(t)| \quad (5.45)$$

$$\bar{C}_B = levy(\gamma) \otimes |m_2 \cdot Z_B(t) - n_2 \cdot Z(t)| \quad (5.46)$$

$$\bar{C}_C = levy(\gamma) \otimes |m_3 \cdot Z_C(t) - n_3 \cdot Z(t)| \quad (5.47)$$

$$\bar{C}_D = levy(\gamma) \otimes |m_4 \cdot Z_D(t) - n_4 \cdot Z(t)| \quad (5.48)$$

$$\bar{Z}_1 = Z_A(t) - d_1 \cdot \bar{C}_A \quad (5.49)$$

$$\bar{Z}_2 = Z_B(t) - d_2 \cdot \bar{C}_B \quad (5.50)$$

$$\bar{Z}_3 = Z_C(t) - d_3 \cdot \bar{C}_C \quad (5.51)$$

$$\bar{Z}_4 = Z_D(t) - d_4 \cdot \bar{C}_D \quad (5.52)$$

The algorithm 5.1 stops the process when the optimal path is found out or when I_{max} is reached.

Algorithm 5.1: Pseudo-code of PL-COA

Input: Population size P , I_{max} , upper and lower bound, dimension

Output: Optimal path

1. Initialize the population y_i^0 randomly with dimension Dim
 2. Compute the location of each chimp
 3. Categorize the chimps randomly into Z_A , Z_B , Z_C , and Z_D
 4. Apply polarity learning on y_i^0 and store the result as OL_i
 5. **while** $I \leq I_{max}$ **do**
 6. **for** $I \leq P$ **do**
 7. Evaluate y_i using Eqn. (5.15) and store the outcome as F_i
 8. Compute the fitness of the solution OL_i using Eqn. (5.15)
 - and store the result in FOL_i
 9. **if** $F_i < FOL_i$ **then**
 10. $y_i = OL_i$
 11. **end if**
 12. **end for**
 13. Initialize h , n , d , and m
 14. **for each chimp do**
 15. Extract the group of the chimp
 16. Update h , n , and m
 17. Compute d and c
 18. **end for**
 19. **for each search chimp do**
 20. **if** $\mu < 0.5$ **then**
 21. **if** $|d| < 1$ **then**
 22. Update search agent's location by Eqn. (5.24-5.27)
 23. **else if** $|d| > 1$ **then**
 24. Choose a search agent randomly
 25. **else if** $\mu < 0.5$ **then**
 26. Apply LF using Eqn. (5.45) to Eqn. (5.48)
 27. Update present search agent's location by Eqn. (5.49-5.52)
 28. **end if**
 29. **end for**
 30. Update h , n , d , and m
 31. Update $Z_A(t)$, $Z_B(t)$, $Z_C(t)$, and $Z_D(t)$
 32. $T = t + 1$
 33. **end while**
 34. Return the optimal path
-

5.3 Simulation Results and Analysis

The proposed ML-HSOR protocol is implemented in Python 3.9 and the obtained results are compared with the recent existing techniques such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197] in terms of energy consumption, delay, detection rate, throughput, etc.

5.3.1 Simulation Setup

The proposed ML-HSOR protocol is implemented in Python 3.9. The implementation relies on NumPy for efficient numerical operations and array manipulations, Matplotlib and Seaborn for compelling data visualization, and scikit-learn's Standard Scaler module to standardize input data. The Cryptography library's Fernet module is employed for secure encryption processes, while SymPy enhances the simulation's mathematical modeling capabilities. Additionally, Python's built-in libraries, such as 'uuid' for universal identifier generation and 'random' for random number generation, contribute to the robustness of the simulation setup. Table 5.1 gives the parameter settings used for implementing the proposed ML-HSOR method.

Table 5.1: Parameter setup

Parameters	Value
Sensing area	$800m \times 800m$
Number of sensors	100-500
BS location	(500,800)
Malicious nodes	5-25
Initial energy of a node	1 J
Size of data packet	512 bytes
Node's communication range	30m
Receiving and Transmission energy	0.01 J
ϵ_{fs}	10 pJ/bit/m^2
E_{elec}	100 nJ/bit
ϵ_{mp}	$0.0013 \text{ pJ/bit/m}^2$
Population	50
Maximum iteration	100
Simulation time	1000 sec
Number of units in the time window	4

5.3.2 Evaluation metrics

The metrics used for evaluating the proposed ML-HSOR method are defined in this section.

(i) Energy Consumption

The energy consumed in the network can be computed by using Eqn. (5.53).

$$E_{con} = (E_{rx} \times N_T) + E_{tx} \quad (5.53)$$

where, N_T denotes the total number of nodes, E_{rx} and E_{tx} represents the energy consumed during reception and transmission respectively.

$$E_{rx} = M_{bits} E_{elec} \quad (5.54)$$

where, M_{bits} signifies the number of bits in the message, the energy needed to activate electronic circuits during reception is expressed as E_{elec} .

The required energy to run the transmitter is given by Eqn. (5.55).

$$E_{tx} = \begin{cases} M_{bits} E_{elec} + M_{bits} \varepsilon_{fs} d^2, & d < d_0 \\ M_{bits} E_{elec} + M_{bits} \varepsilon_{amp} d^4, & d \geq d_0 \end{cases} \quad (5.55)$$

where $d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{amp}}}$, d indicates the distance between the source and the destination node, ε_{amp} and ε_{fs} indicates the dissipated energy in the transmit amplifier and the energy lost in free space respectively.

(ii) Delay

It refers to the time it takes for data to be transmitted from one node to another. It can be computed using Eqn. (5.56).

$$Dly = \sum_{i,j=1}^n \frac{At_j - St_i}{N_{con}} \quad (5.56)$$

where, St_i is the time at which the packet is sent from node i, At_j is the time at which the packet is received at node j, N_{con} indicates number of connections that is made between the source and destination.

(iii) PDR

It is a metric used to measure the percentage of data packets transmitted by a sender that are successfully received by the intended recipient. It is computed by using Eqn. (5.57).

$$PDR = \frac{P_{DN}}{P_{SN}} \times 100 \quad (5.57)$$

where, P_{DN} denotes the data packets received by the destination node (DN) and P_{SN} denotes the data packets sent by the sender node.

(iv) Throughput

It is the volume of data that can be sent across a communication link in a specific amount of time. Typically, it is expressed in bits per second (bps) and can be estimated using Eqn. (5.58).

$$Throughput = \frac{P_d(DN)}{T_{sim}} \times 100 \quad (5.58)$$

where, T_{sim} indicates the simulation time and $P_d(DN)$ represents the packets delivered to the destination.

(v) Detection Rate

It is used to calculate the proportion of events or occurrences observed by the network's sensor nodes like environmental tracking, spying, etc. Detection rate is computed using Eqn. (5.59).

$$D_r = \frac{T_{det}}{T_{act}} \times 100 \quad (5.59)$$

where, T_{det} signifies the sum of malicious nodes that are detected and T_{act} denotes the actual number of malicious nodes.

(vi) Network Lifetime

It the amount of time the network can operate effectively before the batteries of the sensor nodes run out or the nodes begin to fail or malfunction.

5.3.3 Comparative Analysis

The reduction in energy consumption and delay holds paramount importance in the context of Quality of Service (QoS) for IoT wireless sensor networks (WSNs), particularly in applications like environmental monitoring. It directly affects the network's efficiency, reliability, and overall performance. IoT WSNs are often deployed in remote or inaccessible areas, where replacing sensor node batteries is impractical. Minimizing energy consumption is essential for prolonging network lifetime. This extended network lifespan, as demonstrated by the ML-HSOR protocol, reduces maintenance and operational costs, making it highly practical for long-term deployments. Real-time monitoring applications, such as environmental sensing, surveillance, and industrial control, rely on low-latency data transmission. High end-to-end delays can result in missed critical events, affecting the timeliness and effectiveness of decision-making processes. ML-HSOR's reduced delay ensures that

data reaches the application layer promptly, making it suitable for time-sensitive applications.

In addition to energy consumption and delay, other performance indicators also play pivotal roles in evaluating the effectiveness of the ML-HSOR protocol. High throughput, as achieved by the ML-HSOR protocol, is essential for IoT applications requiring real-time data transmission. In environmental monitoring, for example, rapid data transfer ensures that environmental changes are reported promptly, enabling timely response to critical events. The high throughput of ML-HSOR enhances its practicality in these applications. Maintaining a high Packet delivery ratio (PDR) is crucial in IoT WSNs, especially in applications like surveillance and environmental monitoring, where data accuracy is paramount. By enhancing PDR through the prevention of grey-hole attacks, ML-HSOR ensures that the collected data is dependable, making it highly suitable for applications demanding data integrity and trustworthiness. A longer network lifetime, as demonstrated by ML-HSOR, is of immense practical value in remote and challenging deployment scenarios. Reduced maintenance and prolonged operational lifespans reduce the costs and logistical challenges associated with battery replacement, making it a cost-effective solution for applications in remote or hostile environments. High detection rates are essential for ensuring the security of IoT WSNs. By improving the detection rate of malicious nodes, the ML-HSOR protocol enhances network security and reliability. This is of paramount importance in critical infrastructure protection and surveillance applications, where the detection of malicious events is imperative.

The proposed ML-HSOR protocol is evaluated by comparing its performance to established techniques, such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197]. By altering the number of nodes, Fig. 5.5 provides an analysis of energy consumption across different methods.

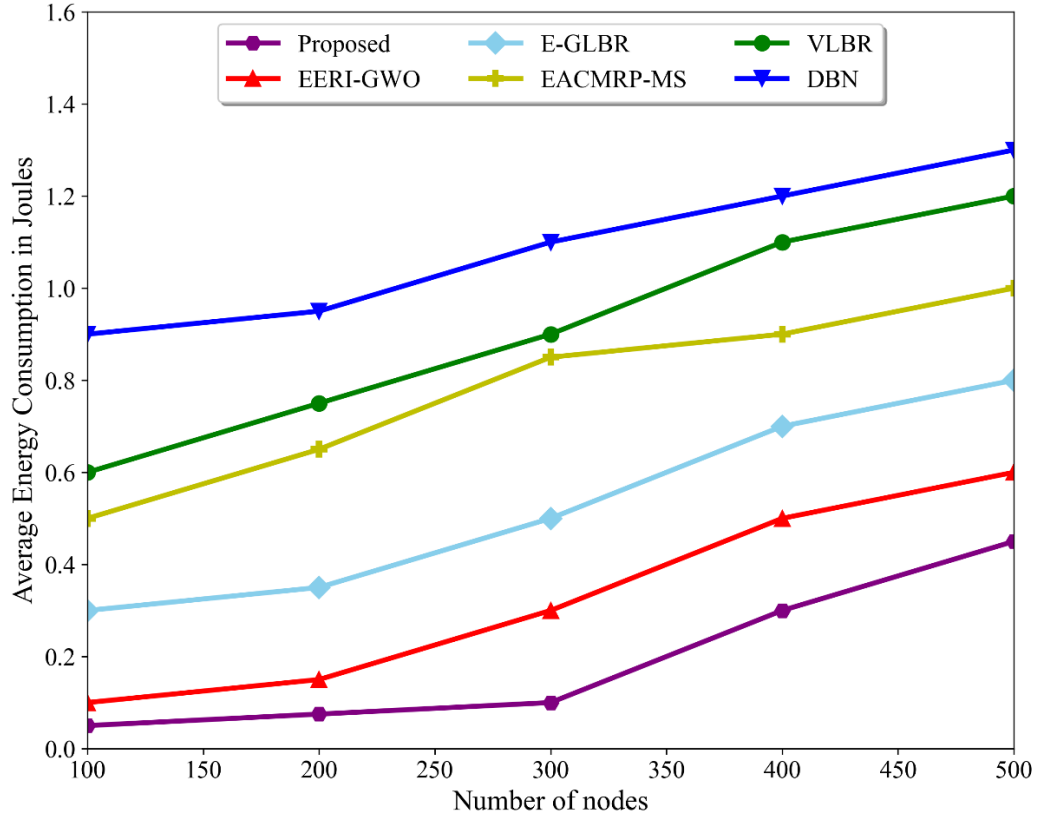


Figure 5.5: Energy consumption analysis

It is observed that the proposed ML-HSOR method has consumed less energy compared to the existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197]. The consumption of energy by the existing EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197] methods are 1.0 J, 1.2 J, 1.1 J, 0.5 J, and 0.7 J when the number of nodes is raised to 500 whereas ML-HSOR has the lowest energy consumption, at 0.39 J. Energy efficient CH selection and routing contribute to minimizing the energy consumption of the proposed ML-HSOR protocol. ML-HSOR exhibits dynamic adaptability by adjusting energy consumption strategies based on real-time variations in network conditions and node characteristics. The comparative analysis explores the protocol's effective handling of varying traffic levels, showcasing its prowess in optimizing energy consumption under fluctuating communication demands. Additionally, the theoretical framework of ML-HSOR is scrutinized for resilience to changes in network topology, emphasizing its capacity to maintain energy efficiency even in dynamic Wireless Sensor Network (WSN) environments. The protocol's robustness extends to fault tolerance, with an in-depth exploration of its capabilities for fault recovery and energy optimization in the face of node failures or network disturbances.

ML-HSOR's scalability is a notable feature, ensuring sustained energy efficiency as the network scales, making it applicable across both small and large-scale deployments. The protocol's adaptability to varying transmission ranges is

investigated theoretically, shedding light on its effectiveness in optimizing energy consumption for different communication distances. Furthermore, the analysis delves into latency considerations, investigating how ML-HSOR minimizes delays, contributing to its overall energy-efficient operation. A nuanced understanding of trade-offs within ML-HSOR, striking a balance between energy efficiency and communication reliability, adds depth to the assessment of its performance characteristics.

The ML-HSOR protocol significantly minimizes energy consumption, making it well-suited for long-term, remote deployments common in environmental monitoring. This is achieved through a dynamic Markov model for CH selection, which intelligently considers real-time energy levels and network conditions. By ensuring judicious selection of CHs and employing energy-efficient routing mechanisms, the protocol optimizes energy consumption. Techniques such as data aggregation and intelligent path selection play a vital role in reducing the energy footprint of the protocol. This reduction in energy consumption has direct practical implications in environmental monitoring. IoT sensors deployed in remote areas, such as rainforests, wildlife reserves, or Arctic ecosystems, can operate for extended periods without the need for frequent battery replacements. The cost savings and logistical benefits of this approach are particularly relevant in environmental conservation and biodiversity studies.

One of the key factors that have a significant impact on the quality of service (QoS) of IoT-based wireless sensor networks is a delay. For real-time monitoring applications, it is required to maintain less delay. High delays occur due to factors such as network congestion or high processing time. When the network has 500 nodes, the delay for the existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197] are 0.08 s, 0.09 s, 0.084 s, 0.065 s, and 0.078 s.

The outcomes from Fig. 5.6, show that the ML-HSOR protocol has less delay of 0.058s compared to the existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197]. The delay is minimized by transmitting the packet via the path that has less overload/congestion.

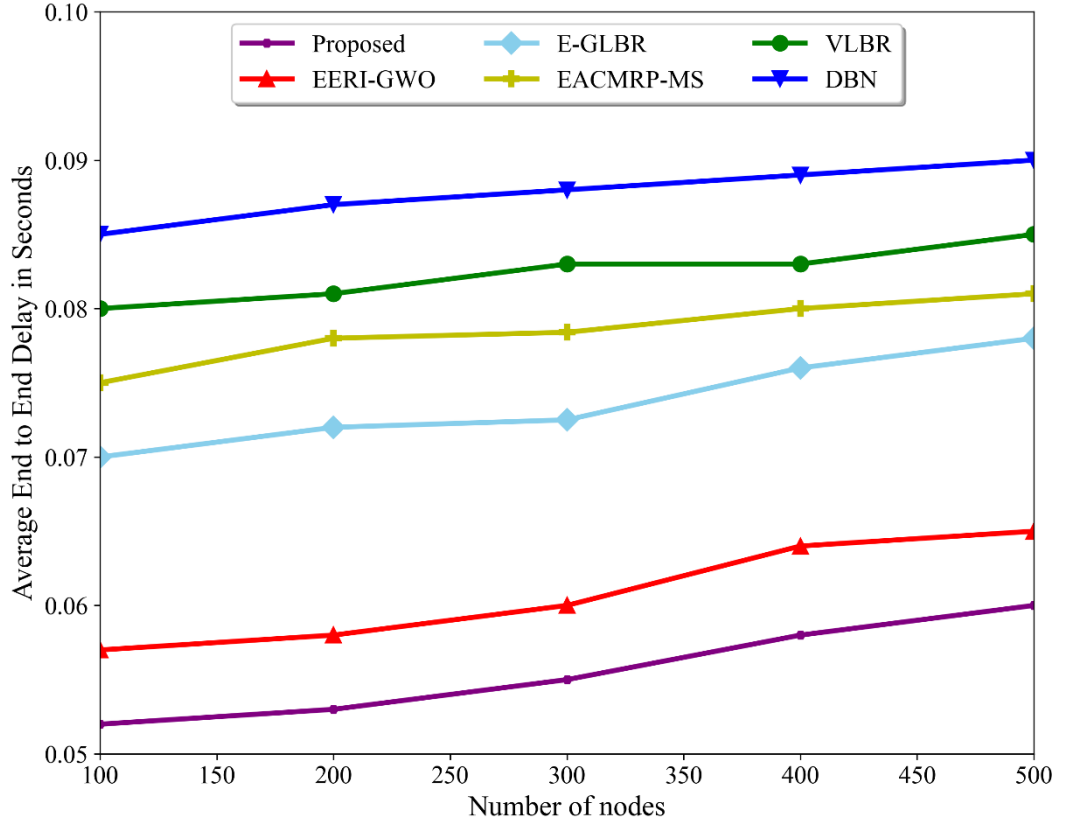


Figure 5.6: End-to-end delay analysis

Low-latency data transmission, as demonstrated by the ML-HSOR protocol, is essential for real-time environmental monitoring. Environmental changes and critical events can occur at any time, and minimizing end-to-end delay is crucial for timely response. By dynamically evaluating network conditions and selecting paths with minimal congestion, the protocol reduces delay. This is particularly important in applications like wildfire detection, where timely data transmission can help mitigate ecological damage and protect human lives. The ML-HSOR protocol's ability to minimize delays directly enhances its practicality in environmental monitoring scenarios.

Throughput is a crucial metric in WSNs because it shows how well the network can transmit data. It depends on a number of variables, including the size of the network, the type of transmission medium, and the data-transfer routing protocol. High throughput is preferred because it guarantees that data can be transmitted quickly and effectively, allowing applications to receive data on time. For instance, high throughput is crucial in applications like environmental monitoring to ensure that the data is transmitted in real-time and can be analyzed quickly to make informed decisions. From Fig. 5.7, it can be observed that the proposed ML-HSOR protocol has a high throughput of 48,000 bps compared to the existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197].

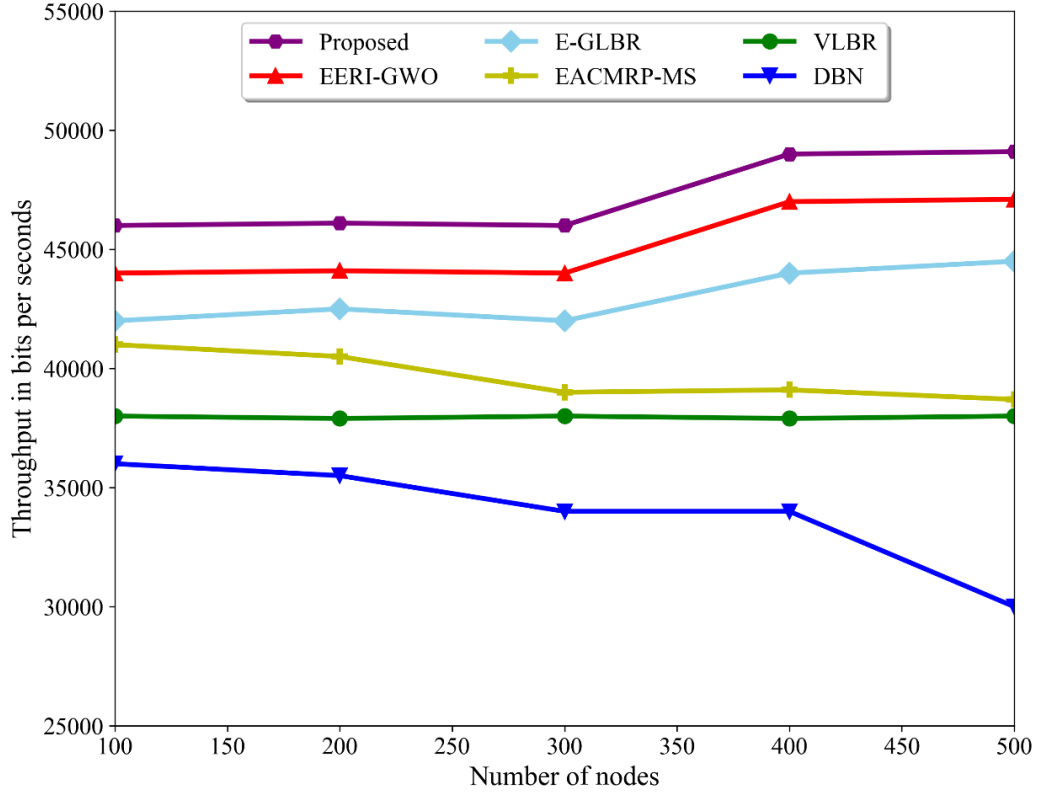


Figure 5.7: Throughput analysis

Based on the analysis, DBN has the least throughput of 30,000 bps followed by VLBR (38,000 bps), EACMRP-MS (39,000 bps), E-GLBR (45,000 bps), and EERI-GWO (47,000 bps). Since the PL-COA algorithm converges quickly by using polarity learning and levy flight, the throughput is increased. The PL-COA algorithm, incorporating polarity learning and Levy flight, enhances the protocol's search for optimal data transmission paths. This results in faster and more efficient data transfer, crucial for applications like environmental monitoring that require real-time data analysis. High throughput, which the ML-HSOR protocol excels at, is indispensable in applications like environmental monitoring. The ability to transfer data quickly is essential for real-time data analysis. For example, in the context of tracking climate parameters in rapidly changing environments, high throughput is crucial for ensuring that data reaches the application layer promptly. The protocol's enhanced throughput is particularly relevant for applications where timely decision-making is imperative. PDR measures the dependability of the communication link between a sender and a receiver in a WSN.

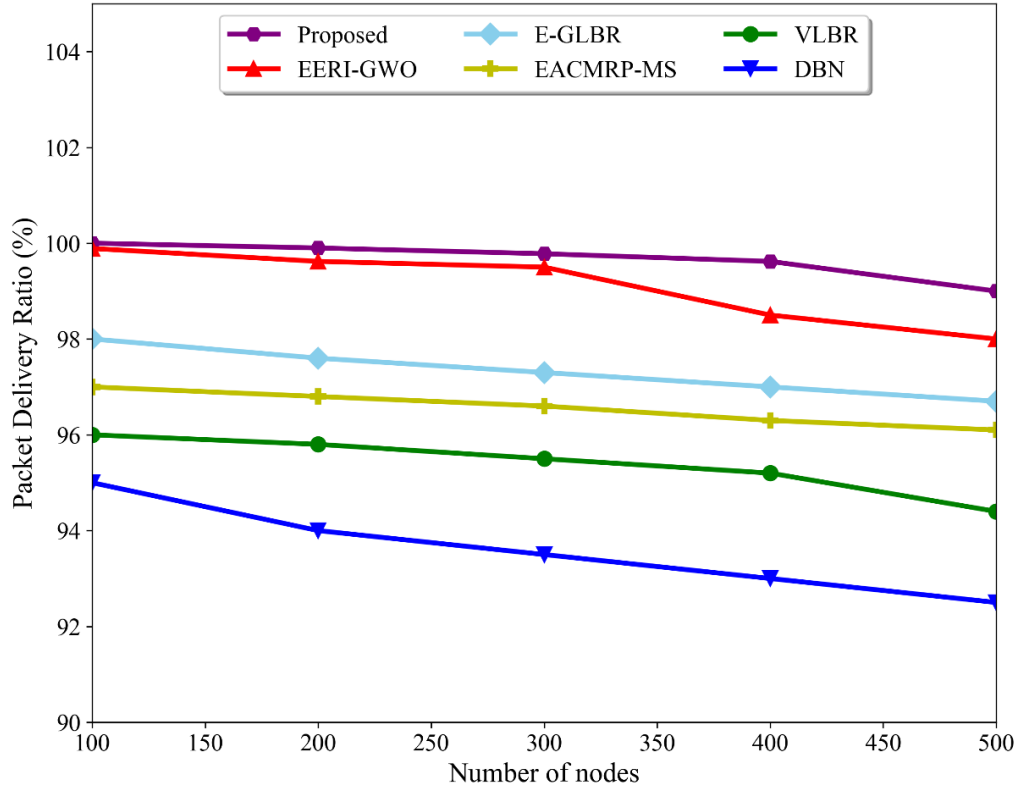


Figure 5.8: PDR analysis

A high PDR indicates that the majority of packets are received successfully, whereas a low PDR indicates that a significant number of packets are lost during transmission. PDR is especially important for applications requiring reliable data delivery, such as surveillance and environmental monitoring, because a high PDR ensures that the collected data is accurate and complete. From Fig. 5.8, it can be seen that the PDR of the proposed ML-HSOR protocol is higher (99.8% for 500 nodes) than the existing methods such as EACMRP-MS (96.5%), DBN (93.0%), VLBR (95.0%), EERIGWO (98.9 %), and E-GLBR (97.5 %). The PDR is increased by preventing grey-hole attacks using the data trust metric. ML-HSOR employs a data trust metric and a multi-level hierarchical trust evaluation, which significantly enhance the PDR. These measures efficiently detect and mitigate grey-hole attacks, ensuring reliable data delivery. A high PDR, as achieved by the ML-HSOR protocol, is vital for applications like environmental sensing and surveillance. In these scenarios, data accuracy and completeness are critical. By efficiently detecting and preventing grey-hole attacks, the protocol ensures that data collected for environmental analysis is reliable and trustworthy. This trustworthiness is vital for making informed decisions based on sensor data, such as those required for pollution control or wildlife conservation.

Network lifetime refers to the time duration for which the network can operate effectively before the batteries of the sensor nodes are depleted or the nodes start to malfunction or fail. The network lifetime is a crucial metric in WSNs, especially in applications where the nodes are deployed in remote locations, and battery replacement

or maintenance is difficult or impossible. A longer network lifetime is needed in such applications to minimize the cost required to operate the network. From Fig. 5.9, it is observed that the proposed ML-HSOR method has 250 alive sensors at the end of 36,000 rounds and EERI-GWO method has 200 alive sensors at the end of 36,000 rounds. On the other hand, all the other existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], and E-GLBR [197] have no alive sensors at the end of 36,000 rounds. The analysis from Fig. 5.9 shows that the network lifetime of the proposed ML-HSOR method is high compared to the existing methods such as EACMRP-MS [193], DBN [194], VLBR [195], EERI-GWO [196], and E-GLBR [197]. The dynamic change in CH by the Markov model helps to increase the network lifetime.

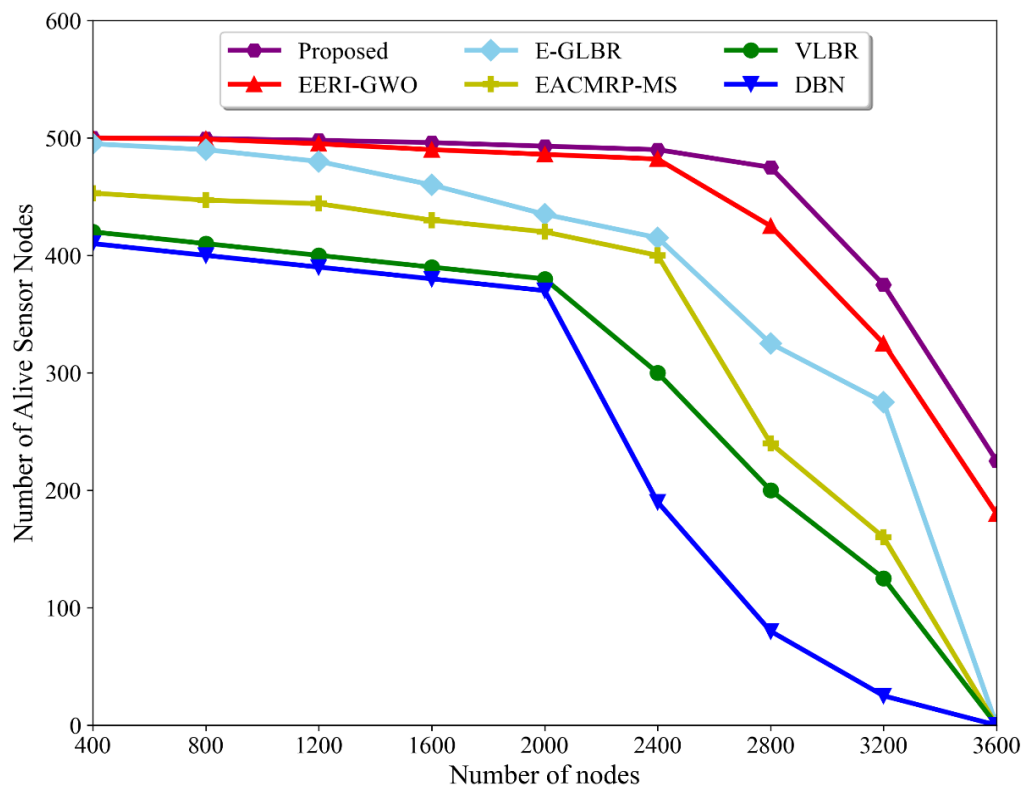


Figure 5.9: Network lifetime analysis

The extended network lifetime achieved by ML-HSOR is crucial for remote and challenging deployment scenarios. The dynamic change in CHs, guided by the Markov model, effectively optimizes energy usage and node lifespan. This is particularly vital in remote locations where battery replacement or maintenance is challenging. Deployments in remote or hostile environments, such as ocean monitoring or desert climate studies, benefit from prolonged network lifespans. Reduced maintenance requirements and operational longevity are cost-effective and logistical advantages in scenarios where sensor nodes are challenging to access.

In an IoT based WSN, high detection rate relies on the security level of the protocol used. A high detection rate indicates that the sensor nodes are efficient at

detecting malicious events, whereas a low detection rate suggests that some malicious events are missed or not noticed by the nodes. When 25 malicious nodes are deployed in the network, the proposed ML-HSOR protocol has the highest detection rate of 95% and DBN has the lowest detection rate of 45 %. The outcomes obtained from Fig. 5.10 show that, compared to existing methods, the proposed ML-HSOR protocol demonstrates a high detection rate. This is because, the multi-level hierarchical trust evaluation in the proposed ML-HSOR protocol increased the detection rate of the malicious nodes.

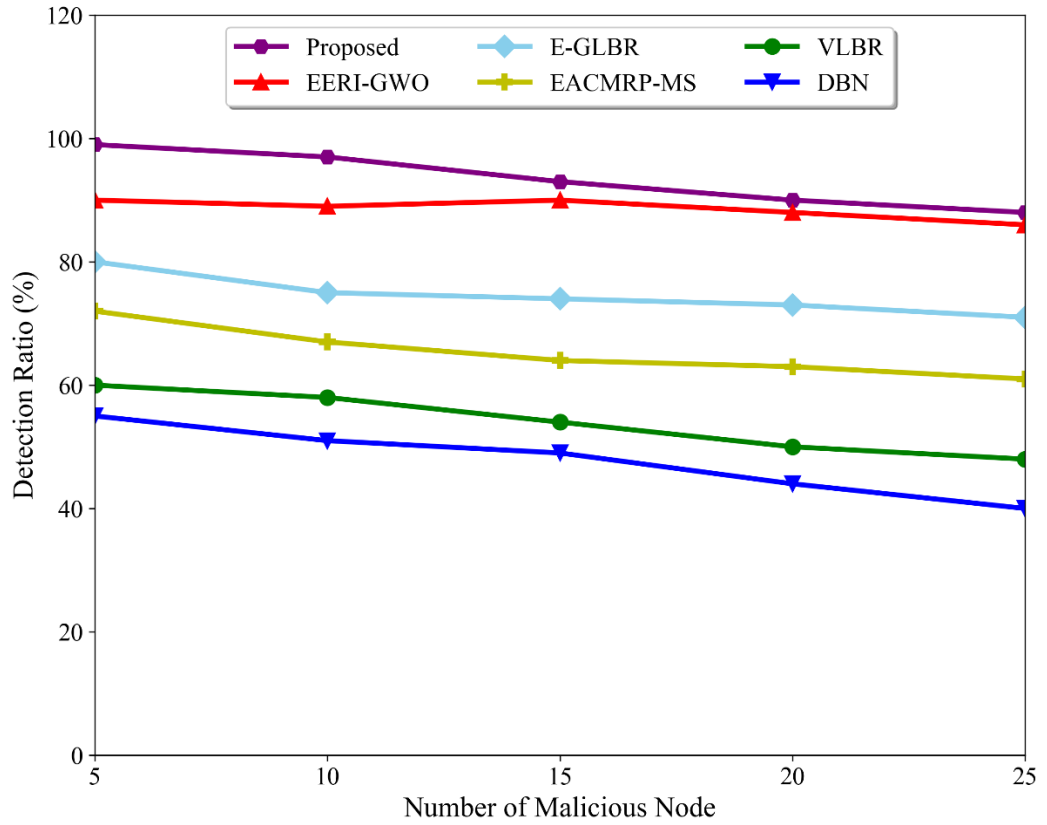


Figure 5.10: Detection rate analysis

ML-HSOR implements a multi-level hierarchical trust evaluation that elevates the detection rate of malicious nodes. This trust-based approach ensures enhanced security and reliability in IoT-based WSNs. In environmental monitoring, maintaining a high detection rate is critical. Malicious events or sensor malfunctions can have adverse ecological consequences. By employing a multi-level hierarchical trust evaluation, the ML-HSOR protocol enhances the detection rate of malicious nodes, ensuring the security and reliability of the monitoring network. This is invaluable in applications like forest fire detection and pollution tracking, where the identification of anomalies is imperative for environmental protection.

The ML-HSOR model distinguishes itself through a myriad of distinctive features, setting it apart from the other models under comparison. A key highlight is its adaptive clustering approach, dynamically adjusting attribute weighting in real-time

during Cluster Head (CH) selection—an aspect that optimizes network lifespan and performance. Furthermore, ML-HSOR confronts the challenge posed by malicious nodes by introducing a unique multi-level hierarchical trust evaluation approach. Factors like interaction trust and identity trust are considered, significantly bolstering network security.

To counteract garnishing attacks, ML-HSOR introduces an innovative time-window mechanism, systematically observing node behavior over specific periods to conduct effective trustworthiness assessments. Security enhancements in ML-HSOR extend to the use of timestamps and the application of the Improved Blowfish Algorithm (IBFA) for encryption. These advanced security measures ensure data integrity and confidentiality, providing an additional layer of protection. Noteworthy are ML-HSOR's optimization strategies involving the PL-COA algorithm, Polarity-Based Learning, and Levy flights. This combination presents a unique approach for efficient exploration and data surveying in complex environmental spaces, demonstrating greater superiority when compared to other models. Thus, ML-HSOR emerges as a comprehensive and advanced solution, adeptly addressing specific challenges in wireless sensor networks through its dynamic adaptability, enhanced security measures, and unique optimization techniques.

In conclusion, the ML-HSOR protocol's superior performance in energy consumption, delay reduction, and other evaluation metrics carries substantial practical implications for environmental monitoring. The reduction in energy consumption enhances the protocol's suitability for long-term deployments in remote areas, reducing operational costs and logistical challenges. Its low-latency data transmission capabilities make it suitable for real-time monitoring, ensuring timely response to critical environmental events. These features position the protocol for applications such as air quality monitoring, wildlife conservation, climate change research, and natural disaster monitoring. The high PDR, throughput, network lifetime, and detection rate further enhance its applicability in environmental monitoring, providing reliable, timely, and cost-effective data collection and analysis. Overall, the ML-HSOR protocol's remarkable performance metrics directly translate into its relevance and significance in real-world scenarios across various industries and applications, with a strong emphasis on environmental monitoring.

5.4 Chapter Summary

In summary, the secure and optimal routing method in IoT based environmental monitoring WSN that employed the improved chimp optimization algorithm is a promising approach for achieving high performance and reliability in WSNs. By computing the fitness of the nodes based on residual energy, distance to the destination, link quality, delay, and trust, this method ensures a high PDR, high throughput, low delay, a high detection rate, and low energy consumption. Moreover, the multi-level trust evaluation approach helps to detect and avoid malicious nodes, making the network more secure and reliable. The improved chimp optimization algorithm and the trust evaluation approach provide an effective solution for addressing the challenges of routing in WSNs, making it suitable for applications like environmental monitoring, smart agriculture, and industrial automation. ML-HSOR significantly reduces energy

consumption through intelligent CH selection, energy-efficient routing, data aggregation, and path optimization. This translates into extended network lifespans, making it an ideal choice for long-term deployments in remote and inaccessible areas where frequent battery replacements are impractical. ML-HSOR's ability to minimize end-to-end delay ensures that data reaches the application layer promptly. This is critical for real-time environmental monitoring, enabling timely responses to crucial events like wildfire detection. The protocol's high throughput, achieved through the PL-COA algorithm, ensures that data can be transmitted quickly and efficiently. This is particularly relevant for applications like environmental monitoring, where real-time data analysis is essential for informed decision-making. Also, ML-HSOR excels in PDR by employing data trust metrics and multi-level hierarchical trust evaluations. This enhances data accuracy and completeness, vital for applications requiring reliable data delivery, such as surveillance and environmental monitoring. The protocol effectively prolongs network lifespans by optimizing energy usage and node lifespan through dynamic CH selection guided by the Markov model. This is a crucial benefit in remote or challenging deployment scenarios. ML-HSOR's high detection rate of malicious nodes, achieved through a multi-level hierarchical trust evaluation, ensures the security and reliability of environmental monitoring networks. This is indispensable for identifying anomalies that could have adverse ecological consequences. Overall, this method is a promising approach for achieving efficient and secure routing in IoT-WSNs.

REFERENCES

- [1] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, “A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications,” *IEEE Internet Things J*, vol. 4, no. 5, pp. 1125–1142, Oct. 2017, doi: 10.1109/JIOT.2017.2683200.
- [2] M. Sain, Y. J. Kang, and H. J. Lee, “Survey on security in Internet of Things: State of the art and challenges,” in *2017 19th International Conference on Advanced Communication Technology (ICACT)*, IEEE, 2017, pp. 699–704. doi: 10.23919/ICACT.2017.7890183.
- [3] A. Tripathi, A. K. Singh, P. Choudhary, P. C. Vashist, and K. K. Mishra, “Significance of Wireless Technology in Internet of Things (IoT),” in *Machine Learning and Cognitive Computing for Mobile Communications and Wireless Networks*, Wiley, 2020, pp. 131–154. doi: 10.1002/9781119640554.ch6.
- [4] B. Suresh and G. Shyama Chandra Prasad, “An Energy Efficient Secure routing Scheme using LEACH protocol in WSN for IoT networks,” *Measurement: Sensors*, vol. 30, p. 100883, Dec. 2023, doi: 10.1016/j.measen.2023.100883.
- [5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013, doi: 10.1016/j.future.2013.01.010.
- [6] M. Vogler, J. Schleicher, C. Inzinger, S. Nastic, S. Sehic, and S. Dustdar, “LEONORE -- Large-Scale Provisioning of Resource-Constrained IoT Deployments,” in *2015 IEEE Symposium on Service-Oriented System Engineering*, IEEE, Mar. 2015, pp. 78–87. doi: 10.1109/SOSE.2015.23.
- [7] J. Granjal, E. Monteiro, and J. S. Silva, “Security in the integration of low-power Wireless Sensor Networks with the Internet: A survey,” *Ad Hoc Networks*, vol. 24, pp. 264–287, Jan. 2015, doi: 10.1016/j.adhoc.2014.08.001.
- [8] Y. Shoshitaishvili *et al.*, “SOK: (State of) The Art of War: Offensive Techniques in Binary Analysis,” in *2016 IEEE Symposium on Security and Privacy (SP)*, IEEE, May 2016, pp. 138–157. doi: 10.1109/SP.2016.17.
- [9] J. W. Hui and D. E. Culler, “Extending IP to Low-Power, Wireless Personal Area Networks,” *IEEE Internet Comput*, vol. 12, no. 4, pp. 37–45, Jul. 2008, doi: 10.1109/MIC.2008.79.
- [10] M. Cäsar, T. Pawelke, J. Steffan, and G. Terhorst, “A survey on Bluetooth Low Energy security and privacy,” *Computer Networks*, vol. 205, p. 108712, Mar. 2022, doi: 10.1016/j.comnet.2021.108712.
- [11] P. Baronti, P. Pillai, V. W. C. Chook, S. Chessa, A. Gotta, and Y. F. Hu, “Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards,”

Comput Commun, vol. 30, no. 7, pp. 1655–1695, May 2007, doi: 10.1016/j.comcom.2006.12.020.

- [12] L. M. Nicolas, T. Eirich, T. Kramp, and O. S. Hersent, “Lorawan Specification, LoRa alliance,” 2015. Accessed: Dec. 27, 2024. [Online]. Available: https://loralliance.org/wp-content/uploads/2020/11/2015_-_lorawan_specification_lr0_611_1.pdf
- [13] J. Olsson, “6LoWPAN demystified,” 2014.
- [14] A. Brandt *et al.*, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” Mar. 2012. doi: 10.17487/rfc6550.
- [15] I. Ud Din, M. Guizani, B.-S. Kim, S. Hassan, and M. Khurram Khan, “Trust Management Techniques for the Internet of Things: A Survey,” *IEEE Access*, vol. 7, pp. 29763–29787, 2019, doi: 10.1109/ACCESS.2018.2880838.
- [16] J.-H. Cho, A. Swami, and I.-R. Chen, “A Survey on Trust Management for Mobile Ad Hoc Networks,” *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 562–583, 2011, doi: 10.1109/SURV.2011.092110.00088.
- [17] P. P. Jadhav and S. D. Joshi, “Atom search sunflower optimization for trust-based routing in internet of things,” *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, vol. 34, no. 3, May 2021, doi: 10.1002/jnm.2845.
- [18] K. Zhang, X. Liang, R. Lu, and X. Shen, “Sybil Attacks and Their Defenses in the Internet of Things,” *IEEE Internet Things J*, vol. 1, no. 5, pp. 372–383, Oct. 2014, doi: 10.1109/JIOT.2014.2344013.
- [19] E. E. Tatar and M. Dener, “Wormhole Attacks in IoT Based Networks,” in *2021 6th International Conference on Computer Science and Engineering (UBMK)*, IEEE, Sep. 2021, pp. 478–482. doi: 10.1109/UBMK52708.2021.9558996.
- [20] A. A. R. Al-chikh Omar, B. Soudan, and Ala’ Altaweel, “A comprehensive survey on detection of sinkhole attack in routing over low power and Lossy network for internet of things,” *Internet of Things*, vol. 22, p. 100750, Jul. 2023, doi: 10.1016/j.iot.2023.100750.
- [21] S. Ali, M. A. Khan, J. Ahmad, A. W. Malik, and A. ur Rehman, “Detection and prevention of Black Hole Attacks in IOT & WSN,” in *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, IEEE, Apr. 2018, pp. 217–226. doi: 10.1109/FMEC.2018.8364068.
- [22] Q. Ye, Y. Wang, M. Xi, and Y. Tang, “Recognition of grey hole attacks in wireless sensor networks using fuzzy logic in IoT,” *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 12, Dec. 2020, doi: 10.1002/ett.3873.
- [23] J. Guo, I.-R. Chen, and J. J. P. Tsai, “A survey of trust computation models for service management in internet of things systems,” *Comput Commun*, vol. 97, pp. 1–14, Jan. 2017, doi: 10.1016/j.comcom.2016.10.012.

- [24] X. Li, F. Zhou, and J. Du, "LDTS: A Lightweight and Dependable Trust System for Clustered Wireless Sensor Networks," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 924–935, Jun. 2013, doi: 10.1109/TIFS.2013.2240299.
- [25] J. Wang, M. Wang, Z. Zhang, and H. Zhu, "Toward a Trust Evaluation Framework Against Malicious Behaviors of Industrial IoT," *IEEE Internet Things J*, vol. 9, no. 21, pp. 21260–21277, Nov. 2022, doi: 10.1109/JIOT.2022.3179428.
- [26] L. Wallgren, S. Raza, and T. Voigt, "Routing Attacks and Countermeasures in the RPL-Based Internet of Things," *Int J Distrib Sens Netw*, vol. 9, no. 8, p. 794326, Aug. 2013, doi: 10.1155/2013/794326.
- [27] Y. Al-Hadhrami and F. K. Hussain, "DDoS attacks in IoT networks: a comprehensive systematic literature review," *World Wide Web*, vol. 24, no. 3, pp. 971–1001, May 2021, doi: 10.1007/s11280-020-00855-2.
- [28] M. Alyami, I. Alharbi, C. Zou, Y. Solihin, and K. Ackerman, "WiFi-based IoT Devices Profiling Attack based on Eavesdropping of Encrypted WiFi Traffic," in *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, IEEE, Jan. 2022, pp. 385–392. doi: 10.1109/CCNC49033.2022.9700674.
- [29] H. Aldabbas and R. Amin, "A novel mechanism to handle address spoofing attacks in SDN based IoT," *Cluster Comput*, vol. 24, no. 4, pp. 3011–3026, Dec. 2021, doi: 10.1007/s10586-021-03309-0.
- [30] A. A. Elsaedy, A. Jamalipour, and K. S. Munasinghe, "A Hybrid Deep Learning Approach for Replay and DDoS Attack Detection in a Smart City," *IEEE Access*, vol. 9, pp. 154864–154875, 2021, doi: 10.1109/ACCESS.2021.3128701.
- [31] A. Le, J. Loo, A. Lasebae, A. Vinel, Y. Chen, and M. Chai, "The Impact of Rank Attack on Network Topology of Routing Protocol for Low-Power and Lossy Networks," *IEEE Sens J*, vol. 13, no. 10, pp. 3685–3692, Oct. 2013, doi: 10.1109/JSEN.2013.2266399.
- [32] S. D. Mali and K. Govinda, "A study on network routing attacks in IoT," *Mater Today Proc*, vol. 80, pp. 2997–3002, 2023, doi: 10.1016/j.matpr.2021.07.092.
- [33] P. Kumari and A. K. Jain, "A comprehensive study of DDoS attacks over IoT network and their countermeasures," *Comput Secur*, vol. 127, p. 103096, Apr. 2023, doi: 10.1016/j.cose.2023.103096.
- [34] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8–27, Feb. 2018, doi: 10.1016/j.jisa.2017.11.002.
- [35] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of Things applications: A systematic review," *Computer Networks*, vol. 148, pp. 241–261, Jan. 2019, doi: 10.1016/j.comnet.2018.12.008.
- [36] S. Sahraoui and N. Henni, "SAMP-RPL: secure and adaptive multipath RPL for enhanced security and reliability in heterogeneous IoT-connected low power and lossy networks," *J Ambient Intell Humaniz Comput*, vol. 14, no. 1, pp. 409–429, Jan. 2023, doi: 10.1007/s12652-021-03303-9.

- [37] R. Rani, V. Kashyap, and M. Khurana, "Role of IoT-Cloud Ecosystem in Smart Cities : Review and Challenges," *Mater Today Proc*, vol. 49, pp. 2994–2998, 2022, doi: 10.1016/j.matpr.2020.10.054.
- [38] T. A. Al-Amiedy *et al.*, "A systematic literature review on attacks defense mechanisms in RPL-based 6LoWPAN of Internet of Things," *Internet of Things*, vol. 22, p. 100741, Jul. 2023, doi: 10.1016/j.iot.2023.100741.
- [39] R. Coulter and L. Pan, "Intelligent agents defending for an IoT world: A review," *Comput Secur*, vol. 73, pp. 439–458, Mar. 2018, doi: 10.1016/j.cose.2017.11.014.
- [40] A. O. Bang, U. P. Rao, P. Kaliyar, and M. Conti, "Assessment of Routing Attacks and Mitigation Techniques with RPL Control Messages: A Survey," *ACM Comput Surv*, vol. 55, no. 2, pp. 1–36, Feb. 2023, doi: 10.1145/3494524.
- [41] D. Airehrour, J. Gutierrez, and S. K. Ray, "Secure routing for internet of things: A survey," *Journal of Network and Computer Applications*, vol. 66, pp. 198–213, May 2016, doi: 10.1016/j.jnca.2016.03.006.
- [42] S. M. Muzammal, R. K. Murugesan, and N. Z. Jhanjhi, "A Comprehensive Review on Secure Routing in Internet of Things: Mitigation Methods and Trust-Based Approaches," *IEEE Internet Things J*, vol. 8, no. 6, pp. 4186–4210, Mar. 2021, doi: 10.1109/JIOT.2020.3031162.
- [43] T. A. Ahanger, A. Aljumah, and M. Atiquzzaman, "State-of-the-art survey of artificial intelligent techniques for IoT security," *Computer Networks*, vol. 206, p. 108771, Apr. 2022, doi: 10.1016/j.comnet.2022.108771.
- [44] R. Yugha and S. Chithra, "A survey on technologies and security protocols: Reference for future generation IoT," *Journal of Network and Computer Applications*, vol. 169, p. 102763, Nov. 2020, doi: 10.1016/j.jnca.2020.102763.
- [45] W. Ejaz, M. Basharat, S. Saadat, A. M. Khattak, M. Naeem, and A. Anpalagan, "Learning paradigms for communication and computing technologies in IoT systems," *Comput Commun*, vol. 153, pp. 11–25, Mar. 2020, doi: 10.1016/j.comcom.2020.01.043.
- [46] A. Altaf, H. Abbas, F. Iqbal, and A. Derhab, "Trust models of internet of smart things: A survey, open issues, and future directions," *Journal of Network and Computer Applications*, vol. 137, pp. 93–111, Jul. 2019, doi: 10.1016/j.jnca.2019.02.024.
- [47] M. binti Mohamad Noor and W. H. Hassan, "Current research on Internet of Things (IoT) security: A survey," *Computer Networks*, vol. 148, pp. 283–294, Jan. 2019, doi: 10.1016/j.comnet.2018.11.025.
- [48] I. Souissi, N. Ben Azzouna, and L. Ben Said, "A multi-level study of information trust models in WSN-assisted IoT," *Computer Networks*, vol. 151, pp. 12–30, Mar. 2019, doi: 10.1016/j.comnet.2019.01.010.
- [49] S. K. Mousavi, A. Ghaffari, S. Besharat, and H. Afshari, "Security of internet of things based on cryptographic algorithms: a survey," *Wireless Networks*, vol. 27, no. 2, pp. 1515–1555, Feb. 2021, doi: 10.1007/s11276-020-02535-5.

- [50] U. Panahi and C. Bayılmış, “Enabling secure data transmission for wireless sensor networks based IoT applications,” *Ain Shams Engineering Journal*, vol. 14, no. 2, p. 101866, Mar. 2023, doi: 10.1016/j.asej.2022.101866.
- [51] M. Stute, P. Agarwal, A. Kumar, A. Asadi, and M. Hollick, “LIDOR: A Lightweight DoS-Resilient Communication Protocol for Safety-Critical IoT Systems,” *IEEE Internet Things J*, vol. 7, no. 8, pp. 6802–6816, Aug. 2020, doi: 10.1109/JIOT.2020.2985044.
- [52] M. T. Hammi, E. Livolant, P. Bellot, A. Serhrouchni, and P. Minet, “A lightweight IoT security protocol,” in *2017 1st Cyber Security in Networking Conference (CSNet)*, IEEE, Oct. 2017, pp. 1–8. doi: 10.1109/CSNET.2017.8242001.
- [53] G. Liu *et al.*, “Softwarized IoT Network Immunity Against Eavesdropping With Programmable Data Planes,” *IEEE Internet Things J*, vol. 8, no. 8, pp. 6578–6590, Apr. 2021, doi: 10.1109/JIOT.2020.3048842.
- [54] M. Kim and T. Suh, “Eavesdropping Vulnerability and Countermeasure in Infrared Communication for IoT Devices,” *Sensors*, vol. 21, no. 24, p. 8207, Dec. 2021, doi: 10.3390/s21248207.
- [55] E. Garcia Ribera, B. Martinez Alvarez, C. Samuel, P. P. Ioulianos, and V. G. Vassilakis, “An Intrusion Detection System for RPL-Based IoT Networks,” *Electronics (Basel)*, vol. 11, no. 23, p. 4041, Dec. 2022, doi: 10.3390/electronics11234041.
- [56] C. Pu and K.-K. R. Choo, “Lightweight Sybil Attack Detection in IoT based on Bloom Filter and Physical Unclonable Function,” *Comput Secur*, vol. 113, p. 102541, Feb. 2022, doi: 10.1016/j.cose.2021.102541.
- [57] S. Ankam and Dr. N. S. Reddy, “A mechanism to detecting flooding attacks in quantum enabled cloud-based lowpower and lossy networks,” *Theor Comput Sci*, vol. 941, pp. 29–38, Jan. 2023, doi: 10.1016/j.tcs.2022.08.018.
- [58] F. Farha and H. Ning, “Enhanced Timestamp Scheme for Mitigating Replay Attacks in Secure ZigBee Networks,” in *2019 IEEE International Conference on Smart Internet of Things (SmartIoT)*, IEEE, Aug. 2019, pp. 469–473. doi: 10.1109/SmartIoT.2019.00085.
- [59] SeungJae Na, DongYeop Hwang, WoonSeob Shin, and Ki-Hyung Kim, “Scenario and countermeasure for replay attack using join request messages in LoRaWAN,” in *2017 International Conference on Information Networking (ICOIN)*, IEEE, 2017, pp. 718–720. doi: 10.1109/ICOIN.2017.7899580.
- [60] F. Farha, H. Ning, shunkun yang, J. xu, W. Zhang, and K.-K. R. Choo, “Timestamp Scheme to Mitigate Replay Attacks in Secure ZigBee Networks,” *IEEE Trans Mob Comput*, pp. 1–1, 2020, doi: 10.1109/TMC.2020.3006905.
- [61] R. Yin, F. Zhang, Y. Xu, L. Liu, and X. Li, “A security routing algorithm against selective forwarding attacks in scale-free networks,” *Procedia Comput Sci*, vol. 174, pp. 543–548, 2020, doi: 10.1016/j.procs.2020.06.151.

- [62] S. Madria and J. Yin, “SeRWA: A secure routing protocol against wormhole attacks in sensor networks,” *Ad Hoc Networks*, vol. 7, no. 6, pp. 1051–1063, Aug. 2009, doi: 10.1016/j.adhoc.2008.09.005.
- [63] P. Deepavathi, B. S. Srisivasubramanyan, and C. Mala, “ESWSIoT: Enhanced Secure Communication by Detecting and Removing Wormhole Attack using Schnorr Digital Signature in IoT Networks,” in *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, Mar. 2023, pp. 2008–2013. doi: 10.1109/ICACCS57279.2023.10113078.
- [64] P. Kaliyar, W. Ben Jaballah, M. Conti, and C. Lal, “LiDL: Localization with early detection of sybil and wormhole attacks in IoT Networks,” *Comput Secur*, vol. 94, p. 101849, Jul. 2020, doi: 10.1016/j.cose.2020.101849.
- [65] D. K. Sharma, S. K. Dhurandher, S. Kumaram, K. Datta Gupta, and P. K. Sharma, “Mitigation of black hole attacks in 6LoWPAN RPL-based Wireless sensor network for cyber physical systems,” *Comput Commun*, vol. 189, pp. 182–192, May 2022, doi: 10.1016/j.comcom.2022.04.003.
- [66] D. J. Jakubisin, C. McPeak, J. Sloop, and B. Davis, “Securing Route Discovery for the Underwater Internet of Things,” in *OCEANS 2022, Hampton Roads*, IEEE, Oct. 2022, pp. 1–10. doi: 10.1109/OCEANS47191.2022.9977183.
- [67] Z. Li, D. Pu, W. Wang, and A. Wyglinski, “Forced collision: Detecting wormhole attacks with physical layer network coding,” *Tsinghua Sci Technol*, vol. 16, no. 5, pp. 505–519, Oct. 2011, doi: 10.1016/S1007-0214(11)70069-4.
- [68] X. Li, K. Huang, S. Wang, and X. Xu, “A physical layer authentication mechanism for IoT devices,” *China Communications*, vol. 19, no. 5, pp. 129–140, May 2022, doi: 10.23919/JCC.2021.00.014.
- [69] D. Chulerttiyawong and A. Jamalipour, “Sybil Attack Detection in Internet of Flying Things-IoFT: A Machine Learning Approach,” *IEEE Internet Things J*, vol. 10, no. 14, pp. 12854–12866, Jul. 2023, doi: 10.1109/JIOT.2023.3257848.
- [70] Y. Wu, T. Jing, Q. Gao, Y. Wu, and Y. Huo, “Game-theoretic physical layer authentication for spoofing detection in internet of things,” *Digital Communications and Networks*, vol. 10, no. 5, pp. 1394–1404, Oct. 2024, doi: 10.1016/j.dcan.2022.12.016.
- [71] J. H. Anajemba, C. Iwendi, I. Razzak, J. A. Ansere, and I. M. Okpalaoguchi, “A Counter-Eavesdropping Technique for Optimized Privacy of Wireless Industrial IoT Communications,” *IEEE Trans Industr Inform*, vol. 18, no. 9, pp. 6445–6454, Sep. 2022, doi: 10.1109/TII.2021.3140109.
- [72] S. Saxena, A. Pandey, and S. Kumar, “RSS based multistage statistical method for attack detection and localization in IoT networks,” *Pervasive Mob Comput*, vol. 85, p. 101648, Sep. 2022, doi: 10.1016/j.pmcj.2022.101648.
- [73] T. Aditya Sai Srinivas and S. S. Manivannan, “Prevention of Hello Flood Attack in IoT using combination of Deep Learning with Improved Rider Optimization Algorithm,”

Comput Commun, vol. 163, pp. 162–175, Nov. 2020, doi: 10.1016/j.comcom.2020.03.031.

- [74] S. Deshmukh-Bhosale and S. S. Sonavane, “A Real-Time Intrusion Detection System for Wormhole Attack in the RPL based Internet of Things,” *Procedia Manuf*, vol. 32, pp. 840–847, 2019, doi: 10.1016/j.promfg.2019.02.292.
- [75] M. Ghahramani, R. Javidan, M. Shojafar, R. Taheri, M. Alazab, and R. Tafazolli, “RSS: An Energy-Efficient Approach for Securing IoT Service Protocols Against the DoS Attack,” *IEEE Internet Things J*, vol. 8, no. 5, pp. 3619–3635, Mar. 2021, doi: 10.1109/JIOT.2020.3023102.
- [76] D. D. N. Nguyen, K. Sood, Y. Xiang, L. Gao, and L. Chi, “Impersonation Attack Detection in IoT Networks,” in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, IEEE, Dec. 2022, pp. 6061–6066. doi: 10.1109/GLOBECOM48099.2022.10001392.
- [77] F. Hendaoui, H. Eltaief, and H. Youssef, “UAP: A unified authentication platform for IoT environment,” *Computer Networks*, vol. 188, p. 107811, Apr. 2021, doi: 10.1016/j.comnet.2021.107811.
- [78] C. Pu, “Sybil Attack in RPL-Based Internet of Things: Analysis and Defenses,” *IEEE Internet Things J*, vol. 7, no. 6, pp. 4937–4949, Jun. 2020, doi: 10.1109/JIOT.2020.2971463.
- [79] J. J. Kponyo, J. O. Agyemang, G. S. Klogo, and J. O. Boateng, “Lightweight and host-based denial of service (DoS) detection and defense mechanism for resource-constrained IoT devices,” *Internet of Things*, vol. 12, p. 100319, Dec. 2020, doi: 10.1016/j.iot.2020.100319.
- [80] M. R. Nosouhi, K. Sood, M. Grobler, and R. Doss, “Towards Spoofing Resistant Next Generation IoT Networks,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1669–1683, 2022, doi: 10.1109/TIFS.2022.3170276.
- [81] B. Huber and F. Kandah, “Beast: Behavior as a Service for Trust Management in Iot Devices,” *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4071348.
- [82] A. Patel and D. Jinwala, “A reputation-based RPL protocol to detect selective forwarding attack in Internet of Things,” *International Journal of Communication Systems*, vol. 35, no. 1, Jan. 2022, doi: 10.1002/dac.5007.
- [83] B. Kumar and B. Bhuyan, “Game Theoretical Defense Mechanism Against Reputation Based Sybil Attacks,” *Procedia Comput Sci*, vol. 167, pp. 2465–2477, 2020, doi: 10.1016/j.procs.2020.03.299.
- [84] B. B. Gupta, P. Chaudhary, X. Chang, and N. Nadjah, “Smart defense against distributed Denial of service attack in IoT networks using supervised learning classifiers,” *Computers & Electrical Engineering*, vol. 98, p. 107726, Mar. 2022, doi: 10.1016/j.compeleceng.2022.107726.
- [85] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, “IoT DoS and DDoS Attack Detection using ResNet,” in *2020 IEEE 23rd International*

- Multitopic Conference (INMIC)*, IEEE, Nov. 2020, pp. 1–6. doi: 10.1109/INMIC50486.2020.9318216.
- [86] A. Mihoub, O. Ben Fredj, O. Cheikhrouhou, A. Derhab, and M. Krichen, “Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques,” *Computers & Electrical Engineering*, vol. 98, p. 107716, Mar. 2022, doi: 10.1016/j.compeleceng.2022.107716.
 - [87] W. Choukri, H. Lamaazi, and N. Benamar, “A Novel Deep Learning-based Framework for Blackhole Attack Detection in Unsecured RPL Networks,” in *2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, IEEE, Nov. 2022, pp. 457–462. doi: 10.1109/3ICT56508.2022.9990664.
 - [88] K. Prathapchandran and T. Janani, “A trust aware security mechanism to detect sinkhole attack in RPL-based IoT environment using random forest – RFTRUST,” *Computer Networks*, vol. 198, p. 108413, Oct. 2021, doi: 10.1016/j.comnet.2021.108413.
 - [89] T. Khan *et al.*, “ETERS: A comprehensive energy aware trust-based efficient routing scheme for adversarial WSNs,” *Future Generation Computer Systems*, vol. 125, pp. 921–943, Dec. 2021, doi: 10.1016/j.future.2021.06.049.
 - [90] A. Altaf, H. Abbas, F. Iqbal, M. M. Z. M. Khan, A. Rauf, and T. Kanwal, “Mitigating service-oriented attacks using context-based trust for smart cities in IoT networks,” *Journal of Systems Architecture*, vol. 115, p. 102028, May 2021, doi: 10.1016/j.sysarc.2021.102028.
 - [91] K. Ahmadi and R. Javidan, “Trust Based IOT Routing Attacks Detection Using Recurrent Neural Networks,” in *2022 Sixth International Conference on Smart Cities, Internet of Things and Applications (SCIoT)*, IEEE, Sep. 2022, pp. 1–7. doi: 10.1109/SCIoT56583.2022.9953707.
 - [92] V. B. Reddy, A. Negi, S. Venkataraman, and V. R. Venkataraman, “A Similarity based Trust Model to Mitigate Badmouthing Attacks in Internet of Things (IoT),” in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, IEEE, Apr. 2019, pp. 278–282. doi: 10.1109/WF-IoT.2019.8767170.
 - [93] P. Srividya and L. N. Devi, “An optimal cluster and trusted path for routing formation and classification of intrusion using the machine learning classification approach in WSN,” *Global Transitions Proceedings*, vol. 3, no. 1, pp. 317–325, Jun. 2022, doi: 10.1016/j.gltp.2022.03.018.
 - [94] S. Chinnaswamy and A. K., “Trust aggregation authentication protocol using machine learning for IoT wireless sensor networks,” *Computers & Electrical Engineering*, vol. 91, p. 107130, May 2021, doi: 10.1016/j.compeleceng.2021.107130.
 - [95] T. Khan, K. Singh, M. Manjul, M. N. Ahmad, A. M. Zain, and A. Ahmadian, “A Temperature-Aware Trusted Routing Scheme for Sensor Networks: Security Approach,” *Computers & Electrical Engineering*, vol. 98, p. 107735, Mar. 2022, doi: 10.1016/j.compeleceng.2022.107735.

- [96] K. Kalkan and K. Rasmussen, "TruSD: Trust framework for service discovery among IoT devices," *Computer Networks*, vol. 178, p. 107318, Sep. 2020, doi: 10.1016/j.comnet.2020.107318.
- [97] H. HaddadPajouh, A. Dehghantanha, R. M. Parizi, M. Aledhari, and H. Karimipour, "A survey on internet of things security: Requirements, challenges, and solutions," *Internet of Things*, vol. 14, p. 100129, Jun. 2021, doi: 10.1016/j.iot.2019.100129.
- [98] A. Tzanetos and G. Dounias, "Nature inspired optimization algorithms or simply variations of metaheuristics?," *Artif Intell Rev*, vol. 54, no. 3, pp. 1841–1862, Mar. 2021, doi: 10.1007/s10462-020-09893-8.
- [99] V. Sharma and A. K. Tripathi, "A systematic review of meta-heuristic algorithms in IoT based application," *Array*, vol. 14, p. 100164, Jul. 2022, doi: 10.1016/j.array.2022.100164.
- [100] A. E. Ezugwu *et al.*, "Metaheuristics: a comprehensive overview and classification along with bibliometric analysis," *Artif Intell Rev*, vol. 54, no. 6, pp. 4237–4316, Aug. 2021, doi: 10.1007/s10462-020-09952-0.
- [101] J. K. Jain, "Secure and Energy-Efficient Route Adjustment Model for Internet of Things," *Wirel Pers Commun*, vol. 108, no. 1, pp. 633–657, Sep. 2019, doi: 10.1007/s11277-019-06422-x.
- [102] J. Ji, G. Wu, J. Shuai, Z. Zhang, Z. Wang, and Y. Ren, "Heuristic Approaches for Enhancing the Privacy of the Leader in IoT Networks," *Sensors*, vol. 19, no. 18, p. 3886, Sep. 2019, doi: 10.3390/s19183886.
- [103] M. Alotaibi, "Improved Blowfish Algorithm-Based Secure Routing Technique in IoT-Based WSN," *IEEE Access*, vol. 9, pp. 159187–159197, 2021, doi: 10.1109/ACCESS.2021.3130005.
- [104] A. Fatani, M. Abd Elaziz, A. Dahou, M. A. A. Al-Qaness, and S. Lu, "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization," *IEEE Access*, vol. 9, pp. 123448–123464, 2021, doi: 10.1109/ACCESS.2021.3109081.
- [105] Z. A. Dagdeviren, "Weighted Connected Vertex Cover Based Energy-Efficient Link Monitoring for Wireless Sensor Networks Towards Secure Internet of Things," *IEEE Access*, vol. 9, pp. 10107–10119, 2021, doi: 10.1109/ACCESS.2021.3050930.
- [106] A. Salim, W. Osamy, A. M. Khedr, A. Aziz, and M. Abdel-Mageed, "A Secure Data Gathering Scheme Based on Properties of Primes and Compressive Sensing for IoT-Based WSNs," *IEEE Sens J*, vol. 21, no. 4, pp. 5553–5571, Feb. 2021, doi: 10.1109/JSEN.2020.3032585.
- [107] C. Anusha, A. Sravani, J. Anusha, C. Lakshmi, and G. S. Kumari, "Intrusion Detection System in IoT Network by using Metaheuristic Algorithm with Machine Learning Dimensional Reduction Technique," in *2022 3rd International Conference on Computing, Analytics and Networks (ICAN)*, IEEE, Nov. 2022, pp. 1–6. doi: 10.1109/ICAN56228.2022.10007341.

- [108] A. K. Jain, D. S. Ross, M. K. Babu, Dharamvir, D. Uike, and D. Gangodkar, "Cloud Computing Applications For Protecting the Information of Healthcare Department Using Smart Internet of Things Appliance," in *2022 5th International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, Dec. 2022, pp. 229–234. doi: 10.1109/IC3I56241.2022.10072938.
- [109] Z. E. Ahmed *et al.*, "Optimization Procedure for Intelligent Internet of Things Applications," in *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*, IEEE, Feb. 2022, pp. 1–6. doi: 10.1109/ICBATS54253.2022.9759065.
- [110] S. Singh, A. S. Nandan, G. Sikka, A. Malik, and A. Vidyarthi, "A secure energy-efficient routing protocol for disease data transmission using IoMT," *Computers and Electrical Engineering*, vol. 101, p. 108113, Jul. 2022, doi: 10.1016/j.compeleceng.2022.108113.
- [111] L. Almuqren, H. Alqahtani, S. S. Aljameel, A. S. Salama, I. Yaseen, and A. A. Alneil, "Hybrid Metaheuristics With Machine Learning Based Botnet Detection in Cloud Assisted Internet of Things Environment," *IEEE Access*, vol. 11, pp. 115668–115676, 2023, doi: 10.1109/ACCESS.2023.3322369.
- [112] M. Biradar and B. Mathapathi, "Energy, Reliability, and Trust-Based Security Framework for Clustering-Based Routing Model in WSN," *International Journal of Information Security and Privacy*, vol. 17, no. 1, pp. 1–18, Jan. 2023, doi: 10.4018/IJISP.315817.
- [113] N. Hijazi, M. Aloqaily, B. Ouni, F. Karray, and M. Debbah, "Harris Hawks Feature Selection in Distributed Machine Learning for Secure IoT Environments," in *ICC 2023 - IEEE International Conference on Communications*, IEEE, May 2023, pp. 3169–3174. doi: 10.1109/ICC45041.2023.10279042.
- [114] K. H. V. Prasad and S. Periyasamy, "Secure-Energy Efficient Bio-Inspired Clustering and Deep Learning-Based Routing Using Blockchain for Edge Assisted WSN Environment," *IEEE Access*, vol. 11, pp. 145421–145440, 2023, doi: 10.1109/ACCESS.2023.3345218.
- [115] M. Hosseinzadeh *et al.*, "A cluster-based trusted routing method using fire hawk optimizer (FHO) in wireless sensor networks (WSNs)," *Sci Rep*, vol. 13, no. 1, p. 13046, Aug. 2023, doi: 10.1038/s41598-023-40273-8.
- [116] A. K. Dey, G. P. Gupta, and S. P. Sahu, "A metaheuristic-based ensemble feature selection framework for cyber threat detection in IoT-enabled networks," *Decision Analytics Journal*, vol. 7, p. 100206, Jun. 2023, doi: 10.1016/j.dajour.2023.100206.
- [117] V. Sharma, R. Beniwal, and V. Kumar, "Towards secure IoT system from a smart city perspective: An optimized algorithm and implementation," *Transactions on Emerging Telecommunications Technologies*, vol. 35, no. 4, Apr. 2024, doi: 10.1002/ett.4883.
- [118] V. Sharma, R. Beniwal, and V. Kumar, "Multi-level trust-based secure and optimal IoT-WSN routing for environmental monitoring applications," *J Supercomput*, vol. 80, no. 8, pp. 11338–11381, May 2024, doi: 10.1007/s11227-023-05875-z.

- [119] I. A. Reshi, S. Sholla, and Z. A. Najar, "Safeguarding IoT networks: Mitigating black hole attacks with an innovative defense algorithm," *Journal of Engineering Research*, vol. 12, no. 1, pp. 133–139, Mar. 2024, doi: 10.1016/j.jer.2024.01.014.
- [120] A. Maharajan and P. Kumar, "Whale optimized routing path selection and 128 bit secured key management for maritime safety," *International Journal of Naval Architecture and Ocean Engineering*, vol. 16, p. 100584, 2024, doi: 10.1016/j.ijnaoe.2024.100584.
- [121] A. Rehman, I. Abunadi, K. Haseeb, T. Saba, and J. Lloret, "Intelligent and trusted metaheuristic optimization model for reliable agricultural network," *Comput Stand Interfaces*, vol. 87, p. 103768, Jan. 2024, doi: 10.1016/j.csi.2023.103768.
- [122] K. Kimani, V. Oduol, and K. Langat, "Cyber security challenges for IoT-based smart grid networks," *International Journal of Critical Infrastructure Protection*, vol. 25, pp. 36–49, Jun. 2019, doi: 10.1016/j.ijcip.2019.01.001.
- [123] A. A. Hussain and F. Al-Turjman, "Artificial intelligence and blockchain: A review," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 9, Sep. 2021, doi: 10.1002/ett.4268.
- [124] S. K. Devineni, S. Kathiriya, and A. Shende, "Machine Learning-Powered Anomaly Detection: Enhancing Data Security and Integrity," *Journal of Artificial Intelligence & Cloud Computing*, pp. 1–9, Jun. 2023, doi: 10.47363/JAICC/2023(2)184.
- [125] J. Ni, K. Zhang, X. Lin, and X. S. Shen, "Securing Fog Computing for Internet of Things Applications: Challenges and Solutions," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 1, pp. 601–628, Jan. 2018, doi: 10.1109/COMST.2017.2762345.
- [126] D. Gupta *et al.*, "A new approach to interdomain routing based on secure multi-party computation," in *Proceedings of the 11th ACM Workshop on Hot Topics in Networks*, New York, NY, USA: ACM, Oct. 2012, pp. 37–42. doi: 10.1145/2390231.2390238.
- [127] N. El Ioini and C. Pahl, "A Review of Distributed Ledger Technologies," 2018, pp. 277–288. doi: 10.1007/978-3-030-02671-4_16.
- [128] C. Tamizhselvan, "A novel communication-aware adaptive key management approach for ensuring security in IoT networks," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 11, Nov. 2022, doi: 10.1002/ett.4605.
- [129] P. Li, A. Shrivastava, J. Moore, and A. C. Konig, "Hashing Algorithms for Large-Scale Learning," Jun. 2011, [Online]. Available: <http://arxiv.org/abs/1106.0967>
- [130] W. by Wouter Penard and T. van Werkhoven, "On the Secure Hash Algorithm family." [Online]. Available: http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Nov07.pdf
- [131] M. Rao, T. Newe, and I. Grout, "Secure Hash Algorithm-3(SHA-3) implementation on Xilinx FPGAs, Suitable for IoT Applications," *International Journal on Smart Sensing and Intelligent Systems*, vol. 7, no. 5, pp. 1–6, Jan. 2014, doi: 10.21307/ijssis-2019-018.

- [132] S. M. S. Hussain, S. M. Farooq, and T. S. Ustun, "Analysis and Implementation of Message Authentication Code (MAC) Algorithms for GOOSE Message Security," *IEEE Access*, vol. 7, pp. 80980–80984, 2019, doi: 10.1109/ACCESS.2019.2923728.
- [133] M. Bellare, "New Proofs for NMAC and HMAC: Security without Collision Resistance," *Journal of Cryptology*, vol. 28, no. 4, pp. 844–878, Oct. 2015, doi: 10.1007/s00145-014-9185-x.
- [134] Y. Li, S. Schäge, Z. Yang, F. Kohlar, and J. Schwenk, "On the Security of the Pre-shared Key Ciphersuites of TLS," 2014, pp. 669–684. doi: 10.1007/978-3-642-54631-0_38.
- [135] Z. Drias, A. Serhrouchni, and O. Vogel, "Identity-based cryptography (IBC) based key management system (KMS) for industrial control systems (ICS)," in *2017 1st Cyber Security in Networking Conference (CSNet)*, IEEE, Oct. 2017, pp. 1–10. doi: 10.1109/CSNET.2017.8242008.
- [136] U. Banerjee, C. Juvekar, S. H. Fuller, and A. P. Chandrakasan, "eeDTLS: Energy-Efficient Datagram Transport Layer Security for the Internet of Things," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, IEEE, Dec. 2017, pp. 1–6. doi: 10.1109/GLOCOM.2017.8255053.
- [137] R. C. Merkle, "Secure communications over insecure channels," *Commun ACM*, vol. 21, no. 4, pp. 294–299, Apr. 1978, doi: 10.1145/359460.359473.
- [138] F. He, W. Feng, Y. Zhang, and J. Liu, "An Improved Byzantine Fault-Tolerant Algorithm Based on Reputation Model," *Electronics (Basel)*, vol. 12, no. 9, p. 2049, Apr. 2023, doi: 10.3390/electronics12092049.
- [139] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovský, "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems," *Knowl Based Syst*, vol. 259, p. 110011, Jan. 2023, doi: 10.1016/j.knosys.2022.110011.
- [140] R. Hyde and P. Angelov, "A fully autonomous Data Density based Clustering technique," in *2014 IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*, IEEE, Dec. 2014, pp. 116–123. doi: 10.1109/EALS.2014.7009512.
- [141] M. K. Hasan *et al.*, "Lightweight Cryptographic Algorithms for Guessing Attack Protection in Complex Internet of Things Applications," *Complexity*, vol. 2021, no. 1, Jan. 2021, doi: 10.1155/2021/5540296.
- [142] J. Andrew, R. J. Eunice, and J. Karthikeyan, "An anonymization-based privacy-preserving data collection protocol for digital health data," *Front Public Health*, vol. 11, Mar. 2023, doi: 10.3389/fpubh.2023.1125011.
- [143] A. Biryukov and D. Khovratovich, "Related-Key Cryptanalysis of the Full AES-192 and AES-256," 2009, pp. 1–18. doi: 10.1007/978-3-642-10366-7_1.
- [144] L. Zhang, J. Xu, P. Vijayakumar, P. K. Sharma, and U. Ghosh, "Homomorphic Encryption-Based Privacy-Preserving Federated Learning in IoT-Enabled Healthcare

System,” *IEEE Trans Netw Sci Eng*, vol. 10, no. 5, pp. 2864–2880, Sep. 2023, doi: 10.1109/TNSE.2022.3185327.

- [145] L. Yang, S. X. Yang, Y. Li, Y. Lu, and T. Guo, “Generative Adversarial Learning for Trusted and Secure Clustering in Industrial Wireless Sensor Networks,” *IEEE Transactions on Industrial Electronics*, vol. 70, no. 8, pp. 8377–8387, Aug. 2023, doi: 10.1109/TIE.2022.3212378.
- [146] M. Arazzi, S. Nicolazzo, and A. Nocera, “A novel IoT trust model leveraging fully distributed behavioral fingerprinting and secure delegation,” *Pervasive Mob Comput*, vol. 99, p. 101889, Apr. 2024, doi: 10.1016/j.pmcj.2024.101889.
- [147] S. Anand and D. A. Sharma, “AgroKy: An approach for enhancing security services in precision agriculture,” *Measurement: Sensors*, vol. 24, p. 100449, Dec. 2022, doi: 10.1016/j.measen.2022.100449.
- [148] M. J. Baucas, P. Spachos, and K. N. Plataniotis, “Federated Learning and Blockchain-Enabled Fog-IoT Platform for Wearables in Predictive Healthcare,” *IEEE Trans Comput Soc Syst*, vol. 10, no. 4, pp. 1732–1741, Aug. 2023, doi: 10.1109/TCSS.2023.3235950.
- [149] M. Al-Hawawreh, N. Moustafa, and J. Slay, “A threat intelligence framework for protecting smart satellite-based healthcare networks,” *Neural Comput Appl*, vol. 36, no. 1, pp. 15–35, Jan. 2024, doi: 10.1007/s00521-021-06441-5.
- [150] Y. Liu *et al.*, “A Blockchain-Based Decentralized, Fair and Authenticated Information Sharing Scheme in Zero Trust Internet-of-Things,” *IEEE Transactions on Computers*, vol. 72, no. 2, pp. 501–512, Feb. 2023, doi: 10.1109/TC.2022.3157996.
- [151] H. Bodur and I. F. T. Al Yaseen, “An Improved blockchain-based secure medical record sharing scheme,” *Cluster Comput*, vol. 27, no. 6, pp. 7981–8000, Sep. 2024, doi: 10.1007/s10586-024-04414-6.
- [152] O. B. J. Rabie, S. Selvarajan, T. Hasanin, G. B. Mohammed, A. M. Alshareef, and M. Uddin, “A full privacy-preserving distributed batch-based certificate-less aggregate signature authentication scheme for healthcare wearable wireless medical sensor networks (HWMSNs),” *Int J Inf Secur*, vol. 23, no. 1, pp. 51–80, Feb. 2024, doi: 10.1007/s10207-023-00748-1.
- [153] H. Li, K. Ota, and M. Dong, “Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing,” *IEEE Netw*, vol. 32, no. 1, pp. 96–101, Jan. 2018, doi: 10.1109/MNET.2018.1700202.
- [154] M. A. Khan and K. Salah, “IoT security: Review, blockchain solutions, and open challenges,” *Future Generation Computer Systems*, vol. 82, pp. 395–411, May 2018, doi: 10.1016/j.future.2017.11.022.
- [155] A. Ullah, G. Said, M. Sher, and H. Ning, “Fog-assisted secure healthcare data aggregation scheme in IoT-enabled WSN,” *Peer Peer Netw Appl*, vol. 13, no. 1, pp. 163–174, Jan. 2020, doi: 10.1007/s12083-019-00745-z.

- [156] A. Prasanth and S. Jayachitra, "A novel multi-objective optimization strategy for enhancing quality of service in IoT-enabled WSN applications," *Peer Peer Netw Appl*, vol. 13, no. 6, pp. 1905–1920, Nov. 2020, doi: 10.1007/s12083-020-00945-y.
- [157] J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, "Security and Privacy for Cloud-Based IoT: Challenges," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 26–33, Jan. 2017, doi: 10.1109/MCOM.2017.1600363CM.
- [158] Y. Lv, Y. Liu, and J. Hua, "A Study on the Application of WSN Positioning Technology to Unattended Areas," *IEEE Access*, vol. 7, pp. 38085–38099, 2019, doi: 10.1109/ACCESS.2019.2903820.
- [159] M. Razzaq, D. Devi Ningombam, and S. Shin, "Energy efficient K-means clustering-based routing protocol for WSN using optimal packet size," in *2018 International Conference on Information Networking (ICOIN)*, IEEE, Jan. 2018, pp. 632–635. doi: 10.1109/ICOIN.2018.8343195.
- [160] T. Gaber, S. Abdelwahab, M. Elhoseny, and A. E. Hassanien, "Trust-based secure clustering in WSN-based intelligent transportation systems," *Computer Networks*, vol. 146, pp. 151–158, Dec. 2018, doi: 10.1016/j.comnet.2018.09.015.
- [161] Z. Liu, H. Seo, A. Castiglione, K.-K. R. Choo, and H. Kim, "Memory-Efficient Implementation of Elliptic Curve Cryptography for the Internet-of-Things," *IEEE Trans Dependable Secure Comput*, vol. 16, no. 3, pp. 521–529, May 2019, doi: 10.1109/TDSC.2018.2825449.
- [162] M. D. Alshehri and F. K. Hussain, "A fuzzy security protocol for trust management in the internet of things (Fuzzy-IoT)," *Computing*, vol. 101, no. 7, pp. 791–818, Jul. 2019, doi: 10.1007/s00607-018-0685-7.
- [163] M. Gupta and A. Sinha, "Enhanced-AES encryption mechanism with S-box splitting for wireless sensor networks," *International Journal of Information Technology*, vol. 13, no. 3, pp. 933–941, Jun. 2021, doi: 10.1007/s41870-021-00626-w.
- [164] K. N. Prasetyo, Y. Purwanto, and D. Darlis, "An implementation of data encryption for Internet of Things using blowfish algorithm on FPGA," in *2014 2nd International Conference on Information and Communication Technology (ICoICT)*, IEEE, May 2014, pp. 75–79. doi: 10.1109/ICoICT.2014.6914043.
- [165] G. Han and L. Zhang, "WPO-EECRP: Energy-Efficient Clustering Routing Protocol Based on Weighting and Parameter Optimization in WSN," *Wirel Pers Commun*, vol. 98, no. 1, pp. 1171–1205, Jan. 2018, doi: 10.1007/s11277-017-4914-8.
- [166] D. B.D. and F. Al-Turjman, "A hybrid secure routing and monitoring mechanism in IoT-based wireless sensor networks," *Ad Hoc Networks*, vol. 97, p. 102022, Feb. 2020, doi: 10.1016/j.adhoc.2019.102022.
- [167] K. Haseeb, N. Islam, A. Almogren, and I. Ud Din, "Intrusion Prevention Framework for Secure Routing in WSN-Based Mobile Internet of Things," *IEEE Access*, vol. 7, pp. 185496–185505, 2019, doi: 10.1109/ACCESS.2019.2960633.

- [168] K. Thangaramya, K. Kulothungan, R. Logambigai, M. Selvi, S. Ganapathy, and A. Kannan, "Energy aware cluster and neuro-fuzzy based routing algorithm for wireless sensor networks in IoT," *Computer Networks*, vol. 151, pp. 211–223, Mar. 2019, doi: 10.1016/j.comnet.2019.01.024.
- [169] Md. J. Islam *et al.*, "Blockchain-SDN-Based Energy-Aware and Distributed Secure Architecture for IoT in Smart Cities," *IEEE Internet Things J*, vol. 9, no. 5, pp. 3850–3864, Mar. 2022, doi: 10.1109/JIOT.2021.3100797.
- [170] K. Haseeb, I. Ud Din, A. Almogren, I. Ahmed, and M. Guizani, "Intelligent and secure edge-enabled computing model for sustainable cities using green internet of things," *Sustain Cities Soc*, vol. 68, p. 102779, May 2021, doi: 10.1016/j.scs.2021.102779.
- [171] W. Fang, N. Cui, W. Chen, W. Zhang, and Y. Chen, "A Trust-Based Security System for Data Collection in Smart City," *IEEE Trans Industr Inform*, vol. 17, no. 6, pp. 4131–4140, Jun. 2021, doi: 10.1109/TII.2020.3006137.
- [172] T. M. Behera, S. K. Mohapatra, U. C. Samal, M. S. Khan, M. Daneshmand, and A. H. Gandomi, "I-SEP: An Improved Routing Protocol for Heterogeneous WSN for IoT-Based Environmental Monitoring," *IEEE Internet Things J*, vol. 7, no. 1, pp. 710–717, Jan. 2020, doi: 10.1109/JIOT.2019.2940988.
- [173] K. Haseeb, N. Islam, A. Almogren, I. Ud Din, H. N. Almajed, and N. Guizani, "Secret Sharing-Based Energy-Aware and Multi-Hop Routing Protocol for IoT Based WSNs," *IEEE Access*, vol. 7, pp. 79980–79988, 2019, doi: 10.1109/ACCESS.2019.2922971.
- [174] D. Gopika and R. Panjanathan, "Energy efficient routing protocols for WSN based IoT applications: A review," *Mater Today Proc*, Nov. 2020, doi: 10.1016/j.matpr.2020.10.137.
- [175] R. Manchanda and K. Sharma, "Energy efficient compression sensing-based clustering framework for IoT-based heterogeneous WSN," *Telecommun Syst*, vol. 74, no. 3, pp. 311–330, Jul. 2020, doi: 10.1007/s11235-020-00652-2.
- [176] I. Kala, S. Karthik, and S. K., "Advanced hybrid secure multipath optimized routing in Internet of Things (IoT)-based WSN," *International Journal of Communication Systems*, vol. 34, no. 8, May 2021, doi: 10.1002/dac.4782.
- [177] A. Banerjee, A. Mitra, and A. Biswas, "An Integrated Application of IoT-Based WSN in the Field of Indian Agriculture System Using Hybrid Optimization Technique and Machine Learning," in *Agricultural Informatics*, Wiley, 2021, pp. 171–187. doi: 10.1002/9781119769231.ch9.
- [178] B. Keswani *et al.*, "Adapting weather conditions based IoT enabled smart irrigation technique in precision agriculture mechanisms," *Neural Comput Appl*, vol. 31, no. S1, pp. 277–292, Jan. 2019, doi: 10.1007/s00521-018-3737-1.
- [179] P. Chanak and I. Banerjee, "Congestion Free Routing Mechanism for IoT-Enabled Wireless Sensor Networks for Smart Healthcare Applications," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 3, pp. 223–232, Aug. 2020, doi: 10.1109/TCE.2020.2987433.

- [180] A. H. Bagdadee, M. Z. Hoque, and L. Zhang, "IoT Based Wireless Sensor Network for Power Quality Control in Smart Grid," *Procedia Comput Sci*, vol. 167, pp. 1148–1160, 2020, doi: 10.1016/j.procs.2020.03.417.
- [181] P. G. Salunkhe and P. U. Chaudhari, "Design WSN Node for Protection of Forest Trees Against Poaching Based MSP430," in *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*, IEEE, Feb. 2018, pp. 520–523. doi: 10.1109/ICACCT.2018.8529377.
- [182] T. Nath and M. Azharuddin, "Application of wireless sensor networks for Rhino protection against poachers in Kaziranga National Park," *AEU - International Journal of Electronics and Communications*, vol. 111, p. 152882, Nov. 2019, doi: 10.1016/j.aeue.2019.152882.
- [183] K. Ghosh, S. Neogy, P. K. Das, and M. Mehta, "Intrusion Detection at International Borders and Large Military Barracks with Multi-sink Wireless Sensor Networks: An Energy Efficient Solution," *Wirel Pers Commun*, vol. 98, no. 1, pp. 1083–1101, Jan. 2018, doi: 10.1007/s11277-017-4909-5.
- [184] A. Ali, Y. K. Jadoon, S. A. Changazi, and M. Qasim, "Military Operations: Wireless Sensor Networks based Applications to Reinforce Future Battlefield Command System," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, IEEE, Nov. 2020, pp. 1–6. doi: 10.1109/INMIC50486.2020.9318168.
- [185] Z. Huanan, X. Suping, and W. Jiannan, "Security and application of wireless sensor network," *Procedia Comput Sci*, vol. 183, pp. 486–492, 2021, doi: 10.1016/j.procs.2021.02.088.
- [186] K. Selvakumar *et al.*, "Intelligent temporal classification and fuzzy rough set-based feature selection algorithm for intrusion detection system in WSNs," *Inf Sci (N Y)*, vol. 497, pp. 77–90, Sep. 2019, doi: 10.1016/j.ins.2019.05.040.
- [187] D. Puri and B. Bhushan, "Enhancement of security and energy efficiency in WSNs: Machine Learning to the rescue," in *2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, IEEE, Oct. 2019, pp. 120–125. doi: 10.1109/ICCCIS48478.2019.8974465.
- [188] M. S. Abdalzaher, L. Samy, and O. Muta, "Non-zero-sum game-based trust model to enhance wireless sensor networks security for IoT applications," *IET Wireless Sensor Systems*, vol. 9, no. 4, pp. 218–226, Aug. 2019, doi: 10.1049/iet-wss.2018.5114.
- [189] A. Shah and M. Engineer, "A Survey of Lightweight Cryptographic Algorithms for IoT-Based Applications," 2019, pp. 283–293. doi: 10.1007/978-981-13-2414-7_27.
- [190] B. A. Ali, H. M. Abdulsalam, and A. AlGhemlas, "Trust Based Scheme for IoT Enabled Wireless Sensor Networks," *Wirel Pers Commun*, vol. 99, no. 2, pp. 1061–1080, Mar. 2018, doi: 10.1007/s11277-017-5166-3.
- [191] A. Amuthan and A. Arulmurugan, "Semi-Markov inspired hybrid trust prediction scheme for prolonging lifetime through reliable cluster head selection in WSNs," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 8, pp. 936–946, Oct. 2021, doi: 10.1016/j.jksuci.2018.07.006.

- [192] Z. A. Al-Odat, S. U. Khan, and E. Al-Qtiemat, "A modified secure hash design to circumvent collision and length extension attacks," *Journal of Information Security and Applications*, vol. 71, p. 103376, Dec. 2022, doi: 10.1016/j.jisa.2022.103376.
- [193] N. Subramani, S. K. Perumal, J. S. Kallimani, S. Ulaganathan, S. Bhargava, and S. Meckanizi, "Controlling energy aware clustering and multihop routing protocol for <scp>IoT</scp> assisted wireless sensor networks," *Concurr Comput*, vol. 34, no. 21, Sep. 2022, doi: 10.1002/cpe.7106.
- [194] G. Arya, A. Bagwari, and D. S. Chauhan, "Performance Analysis of Deep Learning-Based Routing Protocol for an Efficient Data Transmission in 5G WSN Communication," *IEEE Access*, vol. 10, pp. 9340–9356, 2022, doi: 10.1109/ACCESS.2022.3142082.
- [195] E. Hajian, M. R. Khayyambashi, and N. Movahhedinia, "A Mechanism for Load Balancing Routing and Virtualization Based on SDWSN for IoT Applications," *IEEE Access*, vol. 10, pp. 37457–37476, 2022, doi: 10.1109/ACCESS.2022.3164693.
- [196] A. Seyyedabbasi, F. Kiani, T. Allahviranloo, U. Fernandez-Gamiz, and S. Noeiaghdam, "Optimal data transmission and pathfinding for WSN and decentralized IoT systems using I-GWO and Ex-GWO algorithms," *Alexandria Engineering Journal*, vol. 63, pp. 339–357, Jan. 2023, doi: 10.1016/j.aej.2022.08.009.
- [197] A. Benelhouri, H. Idrissi-Saba, and J. Antari, "An Evolutionary Routing Protocol for Load Balancing and QoS Enhancement in IoT Enabled Heterogeneous WSNs," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4135169.

LIST OF PUBLICATIONS

Papers Published in International Journals

1. Beniwal, R., Kumar, V., & Sharma, V. (2025). A Multi-cluster Security Framework for Healthcare IoT: The Synergy of Redundant Byzantine Fault Tolerance with Extensions and Coati-Based Network. *Transactions on Emerging Telecommunications Technologies*, 36(3), e70098. doi:10.1002/ett.70098
2. Sharma, V., Beniwal, R., & Kumar, V. (2024). Multi-level trust-based secure and optimal IoT-WSN routing for environmental monitoring applications. *The Journal of Supercomputing*, 80(8), 11338–11381. doi:10.1007/s11227-023-05875-z
3. Sharma, V., Beniwal, R., & Kumar, V. (2024). Towards secure IoT system from a smart city perspective: An optimized algorithm and implementation. *Transactions on Emerging Telecommunications Technologies*, 35(4), e4883. doi:10.1002/ett.4883

Papers Presented/Published in International Conferences

4. R. Beniwal, V. Kumar and V. Sharma, "Metaheuristics Approaches Towards Secure and Optimized Routing in IoT: A Systematic Literature Review," 2024 International Conference on Electrical Electronics and Computing Technologies (ICEECT), Greater Noida, India, 2024, pp. 1-6, doi: 10.1109/ICEECT61758.2024.10739076.
5. R. Beniwal, V. Kumar and V. Sharma, "Recent Advances in Routing Techniques for IoT-Based Networks: A Comparative Analysis," International Conference on Emerging Trends in Microelectronics, Communication and Intelligent Systems, Pune, India, 2024

DESIGN AND DEVELOPMENT OF SECURE ROUTING TECHNIQUES FOR IOT BASED NETWORKS

**A Thesis Submitted
in Partial Fulfillment of the Requirements for the
Degree of**

DOCTOR OF PHILOSOPHY

**in
Computer Science & Engineering**

**by
Vishal Sharma
(2K20/PHDCO/506)**

Under the Supervision of

**Dr. Rohit Beniwal
(Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University

**Prof. Vinod Kumar
(Co-Supervisor)**
Department of Computer
Science & Engineering
Delhi Technological University



Department of Computer Science & Engineering

**DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India**

June, 2025

CHAPTER 6

CONCLUSION, FUTURE SCOPE AND SOCIAL IMPACT

This chapter presents a comprehensive overview of the framework for cluster security in IoT based networks, and techniques for optimal routing and secure communication. Section 6.1 provides an overview of the research contributions, while Section 6.2 discusses the limitations of the proposed techniques. Section 6.3 explores the future prospects and societal implications of the proposed work.

6.1 Research Summary

Our work introduces new secure routing techniques for IoT based networks. This thesis about design and development of secure routing techniques presents my work on improving foolproof authentication and authorization framework providing security at intra-cluster and inter-cluster levels. This thesis also includes methods combining cryptographic techniques, optimization algorithms, and multi-level trust systems to improve the security and performance of wireless sensor networks in IoT environments. As per the literature review, various studies have studied unauthorized access to data and attacks that can harm the reliability and availability of IoT-based systems. The biggest challenge with IoT-based networks is prolonging network's lifetime and reducing energy consumption while maintaining data transmission security. Although various secure routing techniques and frameworks have been proposed, real-time environmental monitoring systems are particularly prone to security threats, which can further reduce the network's lifespan.

In order to achieve the RO1, i.e., to perform the systematic literature review on secure routing techniques used in IoT based Networks, we explore routing security and performance attributes associated with metaheuristics algorithms and provides an exhaustive review of secure routing across different heuristics techniques for the robust operation of the IoT network. By analysing existing studies and research, the review seeks to identify common metaheuristics techniques used and the corresponding algorithms employed to counteract IoT routing security threats. This study will thoroughly examine these metaheuristics, delving into the particular algorithm employed, their classification, and the security service they present. This study primarily focuses on nature and non-nature inspired metaheuristics. Chapter 2 goes over the whole review of the secure routing techniques used in IoT based Networks in detail.

In order to attain RO2, two solutions have been proposed. In the first solution, blockchain approach is used in proposed Blowfish Honey-Improved Spotted Hyena Optimization (BONY-ISHO) protocol to improve privacy and security in IoT-based WSNs. This blockchain consists of a smart contract (SC), which is used to verify and authenticate the nodes and the data transferred in the network. A node can join the network if its ID is the same as one of those in the blockchain. In this way, the clustering is made more secure, and thus, it becomes impossible for malicious nodes to enter the network. Another solution is based on the multi-cluster Security framework for healthcare IoT, designed to overcome existing limitations in security and scalability. The framework combines Redundant Byzantine Fault Tolerance with Extensions (RB-BFT X) and CoatiNet, leveraging lightweight cryptographic techniques, and role-based access control. RB-BFT X enhances intra-cluster security through fault tolerance and anomaly detection, while CoatiNet optimizes inter-cluster communication using adaptive routing and self-recovery mechanisms inspired by coatis' natural behavior. Additionally, trust-based multi-level authentication is incorporated to detect malicious nodes and mitigate attacks like gray-hole, eavesdropping, and data forgery, ensuring the reliability of environmental monitoring and smart city applications.

Similarly, for achieving RO3, we proposed the enhancement of the spotted hyena optimization (SHO) algorithm to create ISHO, optimizing data transmission routes based on factors like distance, link quality, and node energy levels. ISHO extends the network's lifetime and conserves energy resources, improving energy efficiency. Secondly, to choose optimal data transmission path, we also proposed polarity learning-based chimp optimization algorithm (PL-COA). This approach computed node fitness based on residual energy, distance, link quality, delay, and trust, ensuring high packet delivery ratio, high throughput, less delay, high detection rate, and less energy consumption.

The proposed approaches distinguish itself through a myriad of distinctive features, setting them apart from the other approaches under comparison. Therefore, to achieve RO4, the proposed techniques are tested against previous research in terms of packet delivery ratio, throughput, energy usage, and latency etc. We examined their performance based on key metrics such as operations (GFLOPS), model parameters, inference time, memory space, and execution time. This analysis aims to provide insights into the effectiveness and efficiency of our proposed techniques in comparison to existing techniques.

The contributions of this thesis are presented at the intersection of IoT-based networks, security, energy efficiency, and routing. The main contribution of this thesis is the novel secure routing framework for IoT-based networks. Furthermore, the research optimizes data transmission through an improved bio-inspired routing algorithm, which enhances energy efficiency and prolongs network lifetime. This comprehensive approach establishes a robust foundation for secure and efficient IoT-based WSNs, significantly contributing to the advancement of secure routing techniques. The proposed techniques are validated through extensive simulations and performance evaluations, demonstrating superior results in packet delivery ratio,

energy consumption, throughput, and security robustness compared to existing methodologies.

6.2 Limitations of the Work

No one is perfect in the world, and every study has certain limits and constraints. This work is also subject to the following limitations.

- Security framework's dependency on accurate anomaly detection poses a challenge, as insufficient or imbalanced training data may result in false positives or negatives, which can either overload the system or allow threats to bypass detection.
- Although proposed routing techniques optimizes communication paths, extreme network loads or a high percentage of malicious nodes can still lead to increased latency due to frequent rerouting and security checks.
- The autonomous clustering and configuration processes also require careful tuning and optimization, which may introduce complexity and delay deployment in large-scale networks.
- Furthermore, while the proposed techniques are designed to be scalable, their performance may diminish when scaled to extremely large networks without further optimization or the integration of external resources such as edge computing.

6.3 Social Impact and Future Scope

The unprecedented growth of connected devices and systems have improved lifestyle of individuals providing seamless automation and convenience in everyday tasks allowing more time for other activities. With the use of IoT technology, household users may monitor, control, and reduce their energy consumption, allowing people to conserve even more energy. By means of the dynamic management of connected devices, IoT can assist users in becoming aware of the environmental impact of their technology use. Smart home security systems have completely changed the way homeowners think about security. IoT devices give homeowners peace of mind by enabling real-time doorstep management, even while they are not home. Another application that relies on IoT is elderly care; thus, it is essential for society to prioritize the security of these systems.

To address these limitations and enhance the proposed techniques capabilities, several future improvements are suggested. Incorporating advanced machine learning models, such as federated learning, can improve anomaly detection accuracy while preserving data privacy. Hybrid cryptographic approaches combining lightweight and quantum-resistant techniques can further secure the system against future threats. The integration of edge computing could alleviate computational burdens on IoT devices, reducing latency and enhancing real-time performance. Advanced optimization algorithms could also be employed to improve workload distribution across clusters, addressing imbalances and ensuring consistent performance under heavy loads. Additionally, incorporating adaptive threat intelligence feeds would allow the

framework to detect emerging threats and dynamically adjust its security mechanisms. The methodology will be rigorously validated in diverse real-world environments and industry settings, ensuring its adaptability and robustness across various use cases. This comprehensive approach aligns with the evolving demands of secure and efficient data transmission in modern sensor networks and the dynamic landscape of emerging security threats and network scales.

6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.





Filtered from the Report

- Bibliography
- Small Matches (less than 8 words)




Exclusions

- 24 Excluded Matches

Match Groups

-  **83 Not Cited or Quoted 6%**
Matches with neither in-text citation nor quotation marks
-  **2 Missing Quotations 0%**
Matches that are still very similar to source material
-  **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 2%  Internet sources
- 4%  Publications
- 3%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.