Review Article

A Review of Intrusion Detection Systems in RPL Routing Protocol Based on Machine Learning for Internet of Things Applications

Ali Seyfollahi and Ali Ghaffari

Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

Correspondence should be addressed to Ali Ghaffari; a.ghaffari@iaut.ac.ir

Received 9 May 2021; Revised 12 July 2021; Accepted 21 July 2021; Published 10 August 2021

Academic Editor: Zhaolong Ning

Copyright © 2021 Ali Seyfollahi and Ali Ghaffari. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IPv6 routing protocol for low-power and lossy networks (RPL) has been developed as a routing agent in low-power and lossy networks (LLN), where nodes' resource constraint nature is challenging. This protocol operates at the network layer and can create routing and optimally distribute routing information between nodes. RPL is a low-power, high-throughput IPv6 routing protocol that uses distance vectors. Each sensor-to-wire network router has a collection of fixed parents and a preferred parent on the path to the Destination-oriented directed acyclic graph (DODAG) graph's root in steady-state. Each router part of the graph sends DODAG information object (DIO) control messages and specifies its rank within the graph, indicating its position within the network relative to the root. When a node receives a DIO message, it determines its network rank, which must be higher than all its parents' rank, and then continues sending DIO messages using the trickle timer. As a result, DODAG begins at the root and eventually extends to encompass the whole network. This paper is the first review to study intrusion detection systems in the RPL protocol based on machine learning (ML) techniques to the best of our knowledge. The complexity of the new attack models identified for RPL and the efficiency of ML in intelligent and collaborative threats detection, and the issues of deploying ML in challenging LLN environments underscore the importance of research in this area. The analysis is done using research sources of "Google Scholar," "Crossref," "Scopus," and "Web of Science" resources. The evaluations are assessed for studies from 2016 to 2021. The results are illustrated with tables and figures.

1. Introduction

The Internet of Things (IoT) concept is a new and old concept introduced in 1999 by Kevin Ashton [1, 2]. He described a world where everything, including inanimate objects, has a digital identity of its own, and computers can organize and manage them [3, 4]. When this concept was first introduced, Ashton probably only had in mind the use of radio frequency-based identification chips. Research and transformation in IoT, which encompasses all aspects of human society and simplifies communications through interconnecting billions of ubiquitous objects, provides access and extracts accurate information from the massive volume of data delivered [5]. With the increasing development and use of intelligent equipment, this idea is getting closer to implementation day by day. Forecasts show that between 2009 and 2021, intel-

ligent and interconnected devices will grow by 30 percent to more than 26 billion units. IoT tries to connect all the devices that can process and communicate through the IPv6 protocol. The RPL [6] routing protocol for LLN, also known as 6LoWPAN (IPv6 over low-power wireless personal area networks) [7, 8], was recently proposed to standardize connectivity.

The Internet Engineering Task Force (IETF) organized the routing over low-power and lossy (ROLL) network working group to introduce LLN networks. High packet losses, low bit rate, throughput and delivery ratio, poor stability, constrained resources, and the ability to work in harsh and challenging environments for a long time are the main features of LLNs [9, 10]. The LLN design commenced from the concept such: "the Internet Protocol could and should be applied even to the smallest devices, and that low-power devices with limited processing capabilities should be able

to participate in the Internet of Things. An LLN network is a sensor network that communicates using the RPL protocol and the physical data link layer [11]. In IoT, unlike sensor networks or other wireless communication networks, there are fewer restrictions on wireless equipment connection, and any equipment can be connected to any equipment with any processing and communication capabilities. For this reason, IoT systems are vulnerable to many attacks, and their security strongly influences the security needs of these devices [12]. One of the most important aspects of secure communication in an IoT system is that it is entirely secure. In other words, the relationship between confidentiality and authenticity must be ensured. Establishing a secure connection and ensuring confidentiality and accuracy, and using open authentication makes it impossible to ensure the system's security and communications. The system is still vulnerable to a wide range of attacks. As a de facto routing protocol for IoT infrastructures, RPL has experienced many security threats at the data traffic, network topology, and node resource levels [13, 14].

2. Motivation

Today, one of the most critical and widespread threats is in wireless networks, especially intrusion detection for wireless sensor networks (WSN) [15]. The intrusion action threatens the security of the collection and access to information so that the intruder can use it to advance his sinister goals. Network intrusion is defined as any unauthorized attempt to access, distort, alter, or corrupt information to make a system unreliable. Today, in almost all large-scale information technology infrastructures, we need an efficient intrusion detection system (IDS) to protect our networks from existing and future attacks. Despite advances in protection and detection mechanisms, it is still wholly impossible to protect computer networks. For competing in an arms race against very complex and different types of network intrusions, traditional intrusion prevention methods such as firewalls, access control, or encryption are insufficient to protect the networks [16, 17] entirely. An IDS tries to classify the activity of connections into two categories: normal and abnormal. In more advanced systems, the type of deviant behavior, also called intrusion, is sometimes identified. In an IDS, each connection is described based on a set of features, and decisions about whether the connection is normal or abnormal are made using those features. Intrusion detection systems allow detecting abnormal behavior in the network where an intruder intends to gain irregular access to the network after passing through a network security system. IDS is responsible for detecting and exposing intruders who detect unauthorized activities through network traffic monitoring or user activity reports [15, 16, 18].

It is not possible to prevent the intrusion entirely. Only by taking measures can the dimensions of the intrusion be reduced. Vulnerabilities of software and protocols and the network's structural vulnerabilities provide intruders' conditions to exploit [19]. Software vulnerabilities are caused by poor implementation and programming language. Structural vulnerabilities are network configuration problems in con-

trolling access to critical network points. Intrusion detection is generally divided. Host-based intrusion detection: these systems are installed on the host computers and inspect the system's activities and files. Distributed intrusion detection: a system that collects, processes, and analyzes network packets using several agents on different network parts. These systems operate in a distributed manner to receive information from other parts of the network and distribute the analysis load.

So far, numerous studies have been presented on the design of a robust yet functional IDS for various IoT applications. These IDSs are centralized, hybrid, or distributed, depending on their deployment. The mechanism for intrusion detection, i.e., based on specification, anomaly, signature, misuse-based, or hybrid, can detect, classify, and countermeasure attacks. Some of the work done is designed for the RPL protocol and can detect RPL insider attacks or attacks threatening the network structure and isolating attackers. Using ML and DL algorithms to detect attacks and counter threats intelligently is a new approach that has been considered in recent papers for RPL-based IDS design. Due to the various challenges posed by common scenario-based methods and intrusion detection datasets, the focus on learning-based models to detect and deal with complex and unknown attacks has become increasingly popular with researchers. Several review papers have focused on the LLN network and RPL security in recent years, examining the challenges in designing IDS for LLN-based IoT infrastructures. However, there is a lack of research that comprehensively studies intrusion detection methods for RPL, focusing on ML techniques. Therefore, ML's important role in providing an innovative and efficient intrusion detection strategy for complex and cooperative attacks in RPL is our primary motivation for this work.

3. Contribution and Organization

This paper provides an overview of IoT intrusion detection systems researches. Then, penetration detection techniques, deployment strategy, architectures, attack types, and validation in the proposed solutions are examined. The RPL protocol faces several security challenges as the most critical solution for routing many IoT LLN devices' data and forwarding traffic toward the Internet. Identifying RPL features and classifying the different types of known attacks completes our knowledge to design a robust and efficient IDS. This study, after considering ML and classification methods, as part of the data mining concept, will present the new mechanisms for designing IDS based on these solutions for RPL. The primary purpose of this review is to transfer useful information from RPL-based IoT intrusion detection sources to interested researchers. So, they can identify existing challenges and discover ideas for efficient IDS design for challenging environments facilitating complex and unknown attacks. The critical contribution of this study based on the presented motivations is as follows:

- (i) IoT intrusion detection autopsy including placement strategy, architecture, detection methods, security threats, and assessment methods

- (ii) An overview of the design and system features of RPL, control packets' structure, the construction process, and the techniques for returning to stable conditions
- (iii) Classification of insider and routing attacks in RPL protocol
- (iv) Familiarity with the materials, methods, and technologies provided to penetration identification into the IoT ecosystem
- (v) An overview of data mining, ML and classification concepts, and popular presented models
- (vi) Acquaintance with the equipped facilities by data mining for IDS
- (vii) Introducing the various ML mechanisms proposed to intrusion detection in RPL
- (viii) Providing open issues and challenges and in line with the current research review in IDS design for IoT and LLN

The roadmap of the paper is organized as follows. Section 4 provides a comprehensive literature review on the IDS types designed for the 6LoWPAN and WSN infrastructures in IoT, the standard attacks and threats to IoT applications, intrusion detection systems focusing on objective functions designed for RPL, ML-based solutions for designing IDS for RPL-based IoT, and the types of datasets used by ML-based systems to detect and countermeasure RPL and IoT attacks. Section 5 represents a taxonomy of IoT intrusion detection solutions and describes these categories of IDS placement, architectures, and validation strategies. The RPL routing protocol, design features, control messages' structure, DODAG graph construction process, fixing mechanisms, and trickle timer algorithm are discussed in Section 6. Next, Section 7 introduces and categorizes RPL insider attacks and attacks that threaten the RPL routing process. In Section 8, the data mining concept, machine learning and classification methods, and data mining aspects in IDS design will be presented. Section 9 proposes machine learning-based methods for IDSs used for RPL-based IoT. In Section 10, we will statistically analyze the current review. Sections 11 and 12 represent the discussion and explain the challenges of designing an efficient IDS for RPL-based IoT and future research directions in the present study for potential researchers. Finally, Section 13 provides the conclusion. Table 1 also represents a list of abbreviations and their descriptions in this research work for the reader.

4. Literature Review

In the paper [20], a framework for real-time intrusion detection called SVELTE was proposed and implemented in the Contiki/Cooja system. This method uses three elements to detect intrusion in real-time. The first element collects traffic information on the network. The second element detects the presence or absence of intrusion in the network based on the data collected and analyzed. The third element is a small distributed firewall to prevent the attack's spread and block the

distributed attacks. Pongle and Chavan [21] propose a Denial of Service (DoS) attack detection architecture for 6LoWPAN networks, a standard protocol designed to transfer data between small Internet-connected devices. The proposed architecture is an IDS integrated with the *ebbits* framework. The purpose of presenting such an architecture is to detect and counteract DoS attacks by rejecting 6LoWPAN networks. To evaluate the performance of the proposed architecture, the authors performed the experiments in real-time using penetration testing systems, which showed an improvement in the detection rate of attacks. Besides, with the development of IDS, more attacks can be detected. In the paper [22], a profile-based IDS is presented to detect an attack on IoT routing services. The purpose of this paper is to prevent, detect, and isolate the effect of routing attacks. This method's mechanism identifies and detects the attacker by analyzing the behavior of nodes in the network. The servers collect the behavior and transmission of the nodes and send them to the server. This method has a high accuracy of diagnosis. Bostani and Sheikhan [23] presented a real-time combined method to detect internal intrusion that may occur in the 6LoWPAN network. In this model, the MapReduce method is used to detect distributed intrusion. The proposed model uses real-time anomaly-based and Misuse-based methods to detect intrusion. In other words, this method has used supervised and unsupervised methods to detect intrusion. The main focus of the proposed method is on detecting distributed attacks such as distributed denial of service (DDoS).

Combining various technology, services, and standards to allow IoT solutions necessitates the use of multiple technologies, services, and standards, each with its own set of security and privacy criteria. As a result, it is reasonable to believe that the IoT model, including mobile communications networks (for example, WSN), cloud systems, and the Internet, has security concerns. As previously said, standard protection and privacy controls are inextricably linked to three major factors: IoT components' minimal computing capacity, the vast number of interconnected devices, and data exchange between objects and users cannot be seen for IoT technology. An example of how IoT devices are vulnerable is described in [24]. In this paper, the authors examine three IoT devices' activities (Philips light bulb, Belkin WeMo socket, and Nest smoke alarm) and demonstrate how these devices' protection and privacy can be jeopardized. The authors found a flaw in the answer request message shared between the bridge and the Philips light bulb (a wireless router and the Philips Hue application). The intruder will discover the contact bridge's registered usernames and P addresses by communicating in plain text with them. Using the developers' Python code, the intruder can also take complete control of the communication bridge system via *HTTP PUT* requests. According to the paper [25], IoT technologies' rapid creation may ignore security and privacy threats.

Several security vulnerabilities have been identified by creating popular commercial standard services and products. This paper's researchers put together an intelligent irrigation device that includes a section that provides environmental readings, a module for carrying out user decisions, and a unit that connects the user to the rest of the architecture. They

TABLE 1: Acronyms and their descriptions.

Acronym	Stand for	Acronym	Stand for	Acronym	Stand for	Acronym	Stand for
6BR	6LoWPAN border router	6LoWPAN	IPv6 over low-power wireless personal area networks	ADWA	Acknowledgment-based technique for detection of the wormhole attack	ANN	Artificial neural network
BH	Blackhole attack	CAOF	Context-aware objective function	CARF	Context-aware routing factor	CFS	Correlation-based function selection
CGA	Chaotic genetic algorithm	CH	Cluster head	CHA-IDS	Compression header analysis intrusion detection system	CHlist	The list of children
ClamAV	Clam antivirus	CloneID	Identity clone attack	CLRPL	Context-aware method for load balancing in RPL	CoA-OF	Congestion-aware objective function
CoAP	The Constrained Application Protocol	CoAR	Congestion-aware routing protocol	COOJA	Contiki OS Java simulator	DA	Destination advertisement
DAG	Directed acyclic graph	DAO	Destination advertisement object	DAO-ACK	Destination advertisement object acknowledgment	DARPA	Defense Advanced Research Projects Agency
DDoS	Distributed denial of service	DIO	DODAG information object	DIS	DODAG information solicitation	DLANN	Deep learning-based artificial neural network
DM	Data mining	DNS	Domain name system	DODAG	Destination-oriented directed acyclic graph	DoS	Denial of service
DTSN	Destination advertisement trigger sequence number	E-RPL	Enhanced RPL protocol	E2ED	End-to-end delay	E2V	Energy-based verification and validation
EC	Energy consumption	ETA	Encrypted traffic analytics	ETX	Expected transmission count	FS	Feature selection
HC	Hop count	HF	Hello flooding attack	HTTP	Hypertext transfer protocol	IANA	Internet assigned numbers authority
ICMP	Internet control message protocol	ICMPv6	Internet control message protocol version 6	IDS	Intrusion detection system	IEEE	Institute of electrical and electronics engineers
IETF	The Internet Engineering Task Force	INTI	Intrusion detection of sinkhole attacks on 6lowpan for internet of things	IoT	Internet of things	IP	Internet protocol
IPv6	Internet protocol version 6	KDD 1999	Knowledge Discovery and Data Mining 1999	LB	Load balancing	LBR	Low-power border router
LDA	Linear discriminant analysis	LLN	Low-power and lossy networks	LQE	RL-based link quality estimation	LR	Local repair attack
MAB	Multi-arm bandit	MAC	Medium access control	MCDM	Multicriteria decision-making	ML	Machine learning
MLP	Multilayer perceptron	MLTKNN	The machine learning technique based on K-nearest neighbor algorithm	MOP	Mode of operation	MP2P	Multipoint to point
MQTT	The message queuing telemetry transport	MRHOF	Minimum Rank with Hysteresis Objective Function	NB	Naïve Bayes	NN	Neural networks
OF	Objective function	OF-EC	Energy consumption aware objective function	OF-FL	QoS-aware fuzzy logic objective function	OFQS	Quality of service objective function

TABLE 1: Continued.

Acronym	Stand for	Acronym	Stand for	Acronym	Stand for	Acronym	Stand for
OOP-OF	Opportunistic objective function based on fuzzy logic	P2MP	Point to multipoint	P2P	Point-to-point	PDR	Packet delivery ratio
PITM	Person-in-the-middle	Prf	The route preference	QU	Queue utilization	RAOF	Rank attacks acting as OF
RFC	Request for comment	RInA	Rank inconsistency attack	RL	Reinforcement learning	RPL	IPv6 routing protocol for low-power and lossy networks
RSSI	Received signal strength indicator	SF	Selective forwarding attack	SG	Smart grid	SH	Sinkhole attack
SoS	Securing internet of things against sinkhole attack	SQL	Structured query language	SSH	Secure Shell	TCP	Transmission control protocol
TOPSIS	Technique for order choice mechanism by similarity to ideal solution	UCI	University of California, Irvine	UDP	User datagram protocol	VeRA	Version number and rank authentication
VN	Version number attack	WH	Wormhole attack	WSN	Wireless sensor networks	XSS	Cross-site scripting

have used an Arduino Uno single-board device to execute all sensing and activation functions and the web application. Web server vulnerability, SQL injection threats, infiltration into XSS, and wireless communications are only a couple of the breakthroughs described by the authors. The authors, for instance, listed the following attacks: as it happens in the real world, an intruder potency constructs a Software Powered Access Point alongside the identical Service Set Identifier without authentication. Via all transmissions of bogus authentication packets, it will then temporarily shut down all IoT applications. At this stage, IoT devices can attempt to reconnect to the same app connection point with the same identifier and best signal. According to the authors of the paper, sophisticated operating systems may deter attacks. Still, the operating systems of many IoT devices that lack proper functionality may not be able to say the difference. They will link to the attacker's forged Software Allowed Access Point. As shown in Table 2, an attacker could eavesdrop on network traffic and send remote requests to IoT devices.

IoT customers need to release patch codes for software and hardware vulnerabilities in their products. Also, the development of new IoT products should protect the interaction between IoT entities as a concern. The stability of IoT networks will be improved as a result of these steps. Additional protections, such as intrusion detection systems, are also needed because attackers may attempt to detect new vulnerabilities by merging known vulnerabilities that have not been appropriately secured. Item copying, malicious object replacement, firmware removal, extraction of security parameters, eavesdropping, PITM, routing attack, and denial of service were all listed as security risks involving IoT organizations in the paper [41]. Table 2 organizes the suggested intrusion detection device prototypes for the IoT into groups based on the kinds of threats they will track (according to the authors) [42]. Security risks associated with traditional tech-

nologies and interfaces used to create the IoT ecosystem can extend to the IoT systems, as noted in [42]. For example, insecure communication over HTTP and malicious code injection. These types of attacks are considered regular attacks.

The proposed IoT detection systems can be split into two classes, as seen in Table 3. Detection techniques for routing threats and denial of service attacks. Other attacks listed in the analysis include PITM and regular attacks. In this segment, five classes of related scientific journals are analyzed and discussed. IoT procedures, attacks attributed to Minimum Rank with Hysteresis Objective Function (MRHOF) [43] and OF0 [44] objective functions (OF), IDS procedures and feature collections, datasets and classifiers related to ML, and preprocess and load balancing (LB) methods were the five crucial issues. We have chosen two papers from each group to address here due to page restrictions. Table 3 includes more similar papers from each category and a distance overview for our novel approach. Practical and simulation studies, expert judgment, assessment, interpretation, and opposing viewpoints were used to pick publications, claims, and literature. Researchers used scholarly literature search engines, archives, and newspapers to find research's benefits, shortcomings, and holes.

Propose a survey about IoT-based IDS and employing ML, anomaly-driven approaches, energy-efficient intrusion detection, along with objective function behavior study about IoT methodologies [18], all of which are important to our research in this paper. Centered on mesh-under and route-over [45] systems, researchers address energy usage as a criterion employed to evaluate typical behavior profiles to detect malicious behavior. Each node must check energy utilization at predetermined sampling ratios also disclose any deviations indistinction to planned amounts. Malicious behavior is described as deviations from predicted values, and the sensing node record is cleared in the routing table

TABLE 2: Summary of intrusion detection systems for the Internet of Things.

Proposals	Deployment strategy	Detection method	Security threats	Validation solution
[26]	Centralized	Based on anomalies	Person-in-the-middle (PITM)	Simulation
[27]	—	Based on the signature	—	—
[28]	Hybrid	Based on specifications	Routing attacks	—
[29]	—	Based on specifications	DoS	Simulation
[30]	Centralized	Based on the signature	DoS	Empirical
[31]	Centralized	—	Routing attacks	Simulation
[20]	Hybrid	Hybrid	Routing attacks	Simulation
[32]	—	Based on anomalies	—	—
[33]	Centralized	Based on the signature	—	Hypothetical
[34]	Hybrid	Based on specifications	—	Empirical
[35]	Distributed	Based on the signature	Common attacks (Snort and ClamAV databases)	Empirical
[36]	Distributed	Based on anomalies	DoS	Simulation
[37]	—	Hybrid	Routing attacks and PITM	Simulation
[38]	Distributed	Hybrid	Routing attacks	Simulation
[39]	—	Based on anomalies	Traditional	Empirical
[40]	Hybrid	Based on anomalies	—	—
[21]	Hybrid	Based on anomalies	Routing attacks	Simulation

TABLE 3: Intrusion detection system proposals—security threats.

Recommended system	Detected attacks	Classification
[48]	RPL topology attacks - routing attack and local repair (LR) attack	Routing attack
[42]	Blackhole (BH) attacks and selective forwarding (SF)	Routing attack
[49, 50]	Sinkhole (SH), BH, Sybil, CloneID, SF, hello flooding (HF), and local repair attacks	Routing attack
[51]	BH attacks	Routing attack
[52]	Wormhole (WH) attacks	Routing attack
[53]	RPL-rank topology attacks, BH, neighbors, LR, and DIS (DODAG information solicitation) attacks	Routing attack
[54]	To emulate PITM attacks, simple path attacks (replay, erase, and insert) and invert bits combine byte and field changes with a path	Routing attack and PITM attack
[55]	Some common network attacks include SQL code injection, worm replicas, tunneling systems, and scroll attacks	Conventional attacks
[56]	Botnet attacks in 6LoWPAN networks	PITM attack
[57]	SoS distributed	DoS
[58]	IPv6 UDP (user datagram protocol) flood attack	DoS
[59]	DoS detection using a power consumption model	DoS
[60]	DODAG information solicitation attack in RPL	Routing attack
[61]	SH attack in RPL	—
[62]	Version number (VN) attack	—

accordingly. The research builds against the node behavior concept, emphasizing energy consumption (EC), arguing that packet overhead and memory usage consider appropriate IoT-based IDS criteria. Even though the study presented a deep characterization of IoT-based IDS, it lacked technical scope.

Besides, the mere analysis may not be an appropriate solution to provide an objective approach to design ML-based IDS for IoT. Again, Rehman et al. [46] address the

RPL protocol's RAOF susceptibility. The attacker disservices the routing protocol metrics. Accordingly, the neighboring nodes are attracted to the malicious node and choose him as the preferred parent, which indicates the success of the attacker's objective function. Their simulation results indicate the attack's effect when contemplating a convenient place for the attacker inside the RPL DODAG. When considering RAOF, the research is essential because it discovered a connection between EC, OF, hop count (HC), and other

routing criteria. Nevertheless, it does not give any counterarguments to their strategy.

When proposing an attack on routing metrics, the relation between EC and HC metrics should be explained in more detail. The paper included a systematic review of OF vulnerabilities across RPL networks and implementing a new attack not previously seen in science. Airehrour et al. [42] propose a trust-based RPL design identifying BH and SF attacks in IoT around OF0 and MRHOF. The conscious trust design is examined in contrast to OF0 and MRHOF to see if their method is wealthy. SF attacks across the conscious trust design were increasingly and substantially diminished due to their findings. These attacks involve separating malicious nodes from the rest of the network. MRHOF and OF0, on the other hand, will not identify or cut off SF-based attacks.

Furthermore, the trust-based agreement identified and distinguished BH attacks using transmitted packet arrangement and sequence ID review, but OF0 and MRHOF do not have this feature. They also did not use highly detailed OF0 and MRHOF mechanisms to identify and separate malicious nodes. There is a lack of basic information in this regard. Airehrour et al. [47] also provide a trust-based mechanism for identifying and countermeasuring rank and Sybil attacks. The performance of Sec-Trust-based RPL contrasts with the conventional OF0 and MRHOF-based protocol. MRHOF-based RPL performs better than OF0 based on resource and network flow analysis-based metrics. MRHOF has a higher susceptibility than SecTrust-RPL when just considering Rank and Sybil Assault. Although the authors claim that their protocol is more stable than OF0 and MRHOF, they have not talked about objective function and RPL protocol-based IDS. We need this context to expand our research scope since ML combined with IDS can eliminate the need for RPL and OF to detect and isolate attacks. Experiments comparing SecTrust-RPL to the other two OFs using an appropriate IDS would have given a reasonable assessment when considering IoT protection.

Sheikhan and Bostani [63] describe a security framework for attacks detecting across IoT infrastructures created based on a distributed design in IDS methodologies and feature selection (FS). Their suggested approach focuses on using ML to identify SF and SH attacks that disrupt planned actions or cause irregular activities. According to their reports, anomaly detection effectively detects the SH and SF attack up to 80.95% and 5.92% false alarm rate. Misuse-based monitoring predicts up to 97.88% of SF and SH attacks with a 1.96% false alarm rate. Despite the higher detection probability and lower false alarm rate of SF and SH attacks by the misuse-based mechanism, this method can only detect known attacks. Although this research emphasizes the need to identify significant behavioral characteristics such as packet reception rate, average latency, packet loss rate, and maximum HC, more studies are needed to detect OF-based attacks. Napijah et al. [64] use multilayer perceptron (MLP) to detect hybrid attacks such as HF, WH, SH, and Flood in RPL and 6LoWPAN, called compression header analysis intrusion detection (CHA-IDS). This mechanism, from naïve Bayes (NB), support vector machine (SVM), MLP, Random

Forest, Logistic, and J48 algorithms for collecting and analyzing raw data, offers intrusion detection capability for 6LoWPAN based on anomaly and signature-based features. They have experimental evidence that CHA-IDS outperforms other 6LoWPAN IDS models in detecting mixed attacks. Compared to PONGLE and SVELTE, this mechanism uses compressed header data for 6LoWPAN instead of signal strength and rank indicators calculated from detection functions. Abnormal routing patterns, destination port, context ID, destination context ID, and subsequent header are used to effectively identify attacks by ML algorithms. According to their findings, J48 was the most efficient ML algorithm across all datasets, while Random Forest came in second. Studies on 6LoWPAN and RPL bugs, as well as shortcomings in existing IDS processes, are among the paper's highlights.

Buczak and Guven [70] use the publicly available datasets NetFlow, Knowledge Discovery and Data Mining (DM) 1999 (KDD 1999), Secure Shell (SSH), Domain Name System (DNS), DARPA 1998-2000, and tcpdump for ML algorithm, classification mechanisms, and DM-based intrusion detection. The training should be done on the same dataset to ensure accurate comparisons with other samples during the research process in ML algorithms. KDD 1999 is the best dataset identified since development and is constrained by attacks that may cause problems for use as a reference dataset for OF0 and MRHOF-based RPL in IoT. Eight DM algorithms, ANN (artificial neural network), deep learning-based ANN (DLANN), C4.5, C5.0, k-nearest neighbors, SVM, linear discriminant analysis (LDA), and NB, have been proposed by Alam et al. [71] for the IoT. They use three sensor datasets from the University of California, Irvine (UCI) data warehouse to test new DM algorithms required for IoT or the convenience of existing standard algorithms for IoT datasets. The results of comparisons for IoT datasets showed that C4.5, C5.0, ANN, and DLANN performed better than NB, SVM, neural networks (NN), and LDA in terms of accuracy and elapsed time criteria.

Each ANN contains neurons that can learn complex and nonlinear functions and be appropriate in various fields such as DM, machine vision, medical applications, reinforcement learning, and deep learning by emulating human brain attitude [73, 74]. The ANN and DLANN algorithms had the highest detection accuracy, but they suffered from computational cost and poor memory performance. The two algorithms, C4.5 and C5.0, had high processing speed and accuracy and low memory usage. The study covers an area that was difficult to describe in the literature review, while the KDD and DARPA datasets are primarily used for ML-based IoT analyses. Since the paper contains an original investigation in a field that has not been studied before, it provided more in-depth descriptions of the UCI datasets. Yin and Gai [72] talk about ML and the DM techniques complexities used to solve preprocessing and balancing related to enormous and new data types complexities. Classification processes, preprocessing, feature collection, and data sampling are all discussed in this paper. According to the publication, various classification algorithms are available to establish appropriately balanced, tremendous-quality datasets. The common preprocessing mechanism increases the database's accuracy by proper sampling and reducing

TABLE 4: Machine learning background.

Researcher	Gap investigation for an innovative plan
IoT methodologies	
Zarpelão et al. [18]	This research determined an EC-efficient gap utilizing ML and IDS
Rehman et al. [46]	This research determined an EC-efficient gap utilizing ML and IDS and uses HC metric for RAOF
Le et al. [28]	The combination of attack types is not taken into account. EC and dropped packet aspects are possibly considered as an innovative mechanism to anomaly-established detections
Attacks attributed to MRHOF and OF0 objective functions	
Airehrour et al. [42]	Both OF0 and MRHOF OFs will work in combining different types OF attacks in IoT and declines detecting distinctive attacks across OF0 along with MRHOF
Airehrour et al. [47]	This research determined a gap for detecting/cutting off rank and Sybil attacks using MRHOF and OF0
Mehta and Parma [31]	This work determined a gap that conceivable attacks in OF should be perceptible
<i>IDS procedures along with FS</i>	
Sheikhan and Bostani [63]	This work declined to detect obscure attacks utilizing preferred characteristics for misuse-based detections
Mayzaud et al. [65, 66]	Even though authors believe their research is an appropriate mechanism for IoT-based anomaly detection, they have no compelling reason to detect attacks beyond DODAG
Lee et al. [36]	This study does not evaluate the two objective functions OF0 and MRHOF according to EC and network flow metrics to detecting malicious actions
Sousa et al. [67]	This research explained the various routing metrics and CAO and OF-FL objective functions but did not consider them in the simulations
Napiah et al. [64]	This work has eliminated the EC metric to ensure the effectiveness of the ML algorithms used and reduced the features to 5 features of 77
Datasets and classifiers related to ML	
Haq [68]	This research surveys 49 relevant works and focuses discussions when creating ML-based IDS. The author also considers ML-based techniques, classifier mechanisms, efficient algorithms, datasets, also FS
Nannan et al. [69]	In this paper, classification mechanisms, datasets, FS, ML-based solution, and efficient algorithms are

TABLE 4: Continued.

Researcher	Gap investigation for an innovative plan
	considered, and the authors were able to detect a high false alarm rate to detect anomalies
Buczak and Guven [70]	KDD 1999 is limited by attacks that have appeared since the dataset was produced. This work also considers IoT attacks
Alam et al. [71]	This paper determined limited research into standard ML algorithms using IoT datasets. They consider choosing or making an innovative dataset focuses on IoT-based features along with attacks utilizing ML mechanism, classifier techniques, efficient algorithms, datasets, also FS
Preprocess and LB methods	
Yin and Gai [72]	This work reviews twelve datasets and focuses on considerations when creating an overloaded (imbalanced) dataset

the selected features. The authors tried to provide access to and develop preprocessing mechanisms with ML to create extreme quality datasets. They used only the C4.5 algorithm-based classifier to exclude the contradictory effects of the 12 datasets used.

The findings show that when FS is made before data sampling, a classifier's accuracy is more accurate. When data is largely imbalanced, experimental results show that under sampling rather than oversampling is preferable when considering minority classes. Other preprocessor levels may have been included in the experimental operation to increase the dataset's precision further. Our primary focus areas for the current review paper were identified by following the available literature: network-related and EC criteria, feature collection, vulnerability analysis of OF0 and MRHOF objective functions, development of an innovative dataset based on IoT attacks, and variations to IoT attacks. For example, unlike [18, 64], our paper describes the ML-IDS mechanism that detects a combination of attacks over OF0 and MRHOF based on network and power consumption metrics. Also, unlike [70, 72], the feature reduction, normalization, sampling, and preprocessing methods have been used to create a dataset based on the two mentioned objective functions' attacks. Besides, to the best of our knowledge, no one uses time series-based ML classifiers when using the new IoT dataset to detect a combination of different objective functions (such as OF0 and MRHOF) attacks focusing on network and energy usage metrics. Table 4 summarizes the analysis of the most critical new high-quality papers under review.

5. Intrusion Detection in IoT

Due to IoT's unique features that influence intrusion detection systems' development, the IoT systems' current solutions are

insufficient. First of all, the network nodes' memory and processing capacity that host the IDS is an important issue. IoT networks are made up of limited resource nodes. As a result, locating nodes that can support intrusion agents in IoT applications is more challenging. Second, network architecture's functional characteristics are essential. End systems are connected directly to individual nodes, such as switches and routers, in conventional networks responsible for transferring packets to their destination. However, IoT networks are typically multistep, and normal nodes transmit packets and serve as terminal devices. The final function has to do with network protocols. Protocols not used in standard networks, such as RPL, 6LoWPAN, IEEE 802.15.4, and CoAP (The Constrained Application Protocol) [75], are used in IoT networks [76]. On the other hand, the papers are based on intrusion detection systems for IoT-related components. However, none of them look at basic intrusion mechanisms for IoT. In the following, IDS detection strategies and detection methods are designed, and traditional threats or attacks to security in the IoT also whereby IDSs may be used. Also, the validation strategy used in intrusion detection methods for IoT will be described. In general, the classification of related works is as follows and is shown in Figure 1 [77]: (1) placement strategy, (2) architecture, (3) detection methods, (4) security threats, and (5) validation strategy.

5.1. The Strategy of Intrusion Detection System Placement. An intrusion detection device can be installed on the 6LoWPAN border router (6BR), one or more dedicated hosts, or any physical entity in an IoT network. The ability to detect intruder attempts from the Internet against artifacts in the physical layer is one of the benefits of installing an intrusion detection feature in the 6BR. As a result of the IDS's frequent queries in the network, a 6BR IDS can create a contact overhead between the LLN nodes and the 6BR. The connectivity overhead associated with network control may be minimized by deploying an intrusion system on LLN nodes. It does, however, need more time (energy, storage, processing). Due to the limited capacity of LLN nodes, this may be a challenge. Although the more extensive distribution of intrusion detection equipment on a large scale may lead to less control over network traffic and tremendous computing potential, this strategy requires more precise organization of different network domains, which seems problematic. The advantages and drawbacks of three different placement strategies for IDSs (distributed, centralized, and hybrid) are as follows:

- (i) Distributed intrusion detection system deployment strategy
- (ii) The IDS is located within any LLN network's physical entity in the distributed model [77, 78]

IDS installed in each node must be optimized, and resources are limited. The watchdog nodes (inspection nodes) monitor the activity of neighboring nodes. INTI (intrusion detection of SH attacks on 6LoWPAN for IoT) is a solution proposed by Cervantes et al. [38], which integrates the principles of trust and reputation with watchdog to monitor and minimize attacks. At first, nodes are classified as representa-

tives, linked to members, and arranged in a hierarchical system. Depending on the network reset or an attack case, each node's position will vary over time.

Consequently, each node holds an eye on a superior node by predicting its incoming and outgoing traffic. It broadcasts a message to warn other nodes and isolate the attackers when it identifies an attack node. The authors have not addressed the solution's effect on low-capacity nodes. From the traffic prediction method, we can point to Monte Carlo Q-learning [179] and multitask learning [180].

5.1.1. Placement Technique for a Centralized Intrusion Detection System. The intrusion detection system is installed in a centralized portion in the centralized model, such as the 6BR either a consecrated host. 6BR collects all LLN node data, connects it to the Internet, and requests Internet users to send it to LLN nodes. As a result, the 6BR detection device will examine all traffic passing through the LLN and the Internet. However, traffic analysis via the 6BR is insufficient to identify attacks affecting LLN nodes. A diagnostic method that can detect traffic shared between LLN nodes while avoiding this monitoring behavior on node operations must be built. Monitor with limited power. During an attack that exploits a portion of the network, the centralized intrusion detection system can have trouble tracking nodes even though this approach increases network traffic. The authors demonstrate that LLN nodes do not need extra memory to run the heart rate algorithm, and the energy overhead is negligible.

5.1.2. Placement Technique for Hybrid Intrusion Detection Systems. Hybrid mode blends oriented and distributed principles to win benefits and escape drawbacks. The first solution divides the network into clusters or domains in hybrid mode, with the intrusion detection mechanism only present on each cluster host's primary node. As a consequence, this node is in charge of keeping track of the other cluster nodes. This description tends to be consistent with Cervantes et al. [38]. The authors, on the other hand, divide the network into clusters and choose cluster leaders. Any node, whether it is a leader or not, should keep an eye on its neighbors. Only selected nodes host instances of the IDS in hybrid approaches which are always efficient. As a result, hybrid location IDSs could need more time than distributed location IDSs to build. Amaral et al. [34] suggested a hybrid solution for an IDS considering the Internet of Things. The IDS is hosted on selected nodes in the network in this approach. Through eavesdropping on packets shared in their vicinity, these chosen nodes (watchdogs) assist in detecting intrusions. Based on a series of rules, watchdog determines if a node is corrupted. Since each part of the network can behave differently, every watchdog (inspector node) includes its own rules. A 6BR, for example, receives a higher number of data messages from a common node. The dependence on permission is one of the benefits of this approach that creates various rules for every network area. Lee et al. [56] take a regional network organization approach. By creating a column of observer nodes, they use a combination technique. An observer node listens in on its neighbors' interactions and decides if a node is at risk using a limited number of surveillance nodes that

FIGURE 1: Taxonomy of intrusion detection methods for IoT.

span the whole network. This approach has the benefit of not contributing to the amount of contact needed. Because the monitoring node only listens in on transmissions between its neighbors, this is the case. Lee et al. [57] divide the network into small clusters of several related nodes in another paper.

A cluster head (CH), a directly linked node to all cluster members, is present in each cluster. Every CH has an instance of the IDS that tracks the cluster members by listening to their communications. Cluster participants must remind the CH of essential knowledge about themselves and their neighbors. According to the authors [21, 58], the CH in question could be a more robust node. To build a solution, they choose lightweight IDS. IDS modules are found in the 6BR and other network nodes in the second solution. The inclusion of a core variable distinguishes this strategy from the previous one. The 6BR's IDS module is responsible for tasks that need more resource power. In contrast, standard node intrusion detection systems are typically lightweight. Reza et al. [58] proposed SVELTE, an intrusion detection method in which the 6BR host focuses on IDS module processing. E.g., by analyzing RPL network data, this person is in charge of detecting intrusions. Network nodes handle lightweight functions such as sending RPL network data to the 6BR and notifying the 6BR about malicious traffic they receive. Pongle et al. [21] suggested a system in which network nodes are in charge of identifying changes in their neighborhood and forwarding information to centralized modules in the 6BR. Concentrated modules are in charge of saving and processing this information to track intrusions and identify potential attackers. However, the IDS's explanation may point to an architecture that takes a lot of traffic to detect intrusions. However, the results

show that energy overhead, closed overhead, and memory utilization are adequate in a limited-node environment.

5.2. Intrusion Detection System Architectures. Classification based on the detection system's architecture is independent, distributed, participatory, and hierarchical, and the moving agent is explained in detail in this section [32].

- (i) *Independent Architecture.* Each observer node collects information and detects intrusion itself. The observer node may be centralized or distributed. In a centralized observer node architecture, each network node acts as an observer node. In a distributed observer node architecture, each observer node monitors a specific area of the network. Each sensor node must be within at least one observer node. Each monitoring node has an independent intrusion detection
- (ii) *Distributed and Participatory Architecture.* IDS agents are executed on each monitoring node. All monitoring nodes cooperate in the intrusion detection procedure. IDS monitors its neighboring nodes' behavior, but the exchanged data and alerts with another monitoring node from across the network participate in the overall decision. Such a system improves diagnostic performance. This architecture is suitable for network infrastructure with a DODAG
- (iii) *Hierarchical Architecture.* It is suitable for a cluster sensor network with a hierarchical structure and includes multiple DODAGs (common Sink node). Sink nodes act as CH agents. Local agents are designed and deployed based on an IDS with an

independent architecture and cooperate in the intrusion detection process

- (iv) *Moving Agent Architecture.* Employs several mobile agents to collaborate on the intrusion detection process. The mobility of IDS agents may improve the performance of IDSs. Mobile agents, which can move from one node to another, are a unique executable code that gives particular application self-control. Agent migration means the transfer of the agent between two nodes or the transfer of data, the calculations of which are also performed during this process
- (v) *Examining Many Routing Attack Types.* Attacks such as WH, SF, HF, SH, Sybil, and identifier can be mentioned. Walgren et al. [31] provided an IDS capable of detecting SF attacks. In this regard, Reza et al. [20] can detect the two types of SH and SF threats by their proposed IDS. Cervantes et al. [38] have developed a system for detecting SH attacks. In this work, the authors address the mobility of nodes and network self-repair, which is significantly related to the work of Reza et al. [20]. Pongle et al. [21] have developed an IDS to detect WH attacks

5.3. Validation Strategy. According to Balci [79], validation entails testing the built model's behavior according to the research objectives with sufficient accuracy. Several confirmation processes, each characterized through the two sources of proof: specialists and data (information). The employment of experts, on the other hand, offers a subjective and often qualitative paradigm. For quantitative validation, data could be more relevant. This analysis aims to look at the validation technique used in IoT intrusion detection methods. Such parameters will serve as a starting point for deciding the field's maturity. The following classification of validation methods is given for this reason [79]:

- (i) *Hypothetical.* Cases of an ambiguous relationship to actual phenomena and varying degrees of realism
- (ii) *Empirical.* Experimental approaches, such as gathering systematic experimental evidence from organizational contexts, are empirical methods
- (iii) *Simulation.* Methods for simulating such IoT situations
- (iv) *Theoretical.* Formal or systematic scientific claims that justify results are referred to as theoretical
- (v) *None.* There is no form of validation included

Scientific progress is based on the completion of results. They can be objectively tested and compared in large-scale simulations. Most traditional intrusion detection system analysis is based on data from Lincoln Laboratory/DARPA tests in 1998 and 1999. This work is the most thorough review of intrusion detection analysis that has been published to date and is the most comprehensive evaluation of intrusion detection research to date. Several criticize and point

out that this is an ancient data set that cannot adapt to attacks' latest trends. Having a data set is crucial to understanding a model correctly.

6. RPL Routing Protocol

The RPL protocol is designed for routing in LLN, in which frequent connection interruptions and packet losses are inevitable. RPL has static and reactive nature with its tree-like structure. In partnership with IPv6, it provides data aggregation and interoperability between Internet-based devices. In a 6LoWPAN network, RPL is mainly used. In an RPL-based 6LoWPAN, this protocol generates a DODAG. It also supports single-way traffic to a destination-oriented graph between 6LoWPAN devices and between devices and DODAG root without a destination-oriented graph and two-way traffic (usually 6BR). The RPL protocol is Proactive and starts routing as soon as the network begins. In a network, each node has a CH that acts as a gateway for that node. If the node does not have information in its routing table to direct the packet, it redirects it to its CH node. This guidance will continue until the node reaches its destination or the Sink node with relevant information. Therefore, the header node will have a larger routing table. Route selection is one of the essential factors in RPL [80].

The OCSVM technique for anomalous detection in Supervisory control and data acquisition (SCADA) Networks based on machine learning may achieve a high detection rate, substantially lowering the false alarm and false-negative rates. The restrictions of a one-class technique on a kernel strategy are how to determine the suitable threshold and minimize the associated cost. IForest's implementation impact is not high in OCSVM, but it has accurate and can manage the advantages of important information. Hence, it is suitable for use in online learning. To meet the targets of multiclassification, you may use the supervised learning algorithm, particularly the decision tree technique, to quickly learn, construct, and store the learned intrusion rules. Although KNN's usage of feature vectors has a comparable impact to decision trees, KNN's processing cost restricts its application [181, 182]. The infiltration into the industrial control network differs from the intrusion into the Internet network. Somewhat of focusing on network communication flaws, the former threat focuses on faults associated with employing industrial control and industrial gear. ML and data analysis may be used to discover the link between normal and aberrant. During the experiment, it was discovered that using the one-class classification approach for intrusion detection can only identify abnormal, not locate abnormal classes. The use of such unsupervised learning for intrusion detection limits the experimental goal and results from interpretation; nevertheless, using a semisupervised method to the intrusion detection system is an enhanced aspect. In addition to low cost, security, and mobility support, industrial applications demand dependable connectivity with minimal latency [183]. RPL is gaining much traction in industrial applications since it meets most of the fundamental criteria and, with the existing enhancements, can be used to create a versatile, reliable, and scalable routing solution. GTM-RPL improves RPL's performance by allowing

FIGURE 2: Example of an RPL network with two instances and three DODAG.

it to handle mobile nodes and optimize throughput, making it a viable option for industrial uses [184].

6.1. Design Goals and Network Model Based on RPL. The RPL routing protocol is an over-the-top interconnection mechanism comprising MAC (medium access control) and physical IEEE 802.15.4 layers, distance vector protocol, and source routing protocol. The RPL protocol also has a tree-like structure. The nodes alternately send their sensing data to central points called the low-power border router (LBR) or 6BR, which is the cumulative point of traffic for low-power nodes. Finally, the data is routed to the Internet or a non-DODAG structure. The RPL supports point-to-point traffic as well. The RPL LLNs have two principal characteristics, in particular [81]:

- (1) Usually, the bit rate is low (lower than 250 kbps)
- (2) Correspondence has a high rate of error and, as a result, poor data throughput

A low-power connection has a high bit error rate and a long unavailability period, which significantly impacts routing protocol architecture. When default routes are unavailable, the protocol is configured to respond to high network conditions and offer alternate routes. RPL is built on the directed acyclic graph (DAG) topology, as previously mentioned. The DAG defines the default paths between nodes in a tree-like layout. On the other hand, a DAG system is more than just a regular tree [82]. A node in a DAG can have several parents, while classical trees can only have one. The RPL organizes the nodes as a destination-oriented DAG. The DAG root is a default Internet path (port) provided by the destination nodes (Sink). One or more DODAG may exist in a network, each of which specifies an RPL Instance with a single identifier. Several RPL instances will run in the same network simultaneously, but they are technically different. A node can be connected to several RPL instances, but each instance can only have one DODAG. The RPL routing protocol combines both mesh and hierarchical topologies as one of its functions. According to its design, the RPL pro-

tolocol acquires a hierarchical mechanism that nodes can participate in one or more DODAGs structures simultaneously based on different parameters such as the application type. RPL supports a mesh topology, enabling routing, if necessary, through sibling nodes, instead of parents and children. In terms of topology control and routing, this hybrid of mesh and hierarchical networks gives a lot of versatility [83]. Figure 2 provides an overview of the two-instance RPL network and the three DODAG networks.

The following features are included in the RPL protocol [29]:

Autoconfiguration: as RPLs comply with IPv6, RPL-based LLNs usually use simple IP routing functions to dynamically find network routes and destinations. This functionality is assured with nearby pathways of detection.

Self-healing: RPL has demonstrated the capacity to respond logically to topology and node faults in the network. Links and nodes are not constant in LLNs and can vary widely. The RPL implements mechanisms to remove or reduce the risk of failure by selecting more than one parent for each DAG node.

Loop avoidance and detection: due to its nonloop existence, a DAG should get a greater rank than its parent nodes. RPL uses reactive procedures to discover loops for topological shifts. It also initiates global and local recovery procedures to fix or prevent loops.

Independence and transparency: RPL is a link-layer protocol that can be used on restricted networks or in combination with highly restricted systems. As a result, RPL is unaffected by data link layer technologies.

Multiple edge routers: in an RPL network, multiple DAGs may be formed, each with its core. A node may be part of several instances and play various roles in each of them. As a consequence, network availability and LB would help the network.

The three traffic patterns shown in Figure 2 of the third DODAG [28] can be used to send RPL packets:

- (i) Multipoint to point (MP2P) uses upward directions from the leaves to the root

- (ii) Point to multipoint (P2MP) uses downward directions, from root to leaf
- (iii) Point-to-point (P2P) uses all up and down directions of network routes

Each of the traffic patterns is explained in turn in the following sections.

6.1.1. MP2P (Multipoint-to-Point) Mode of Operation. RPL protocol can handle MP2P traffic, data aggregation traffic from several nodes, and the DODAG root. In most LLN-based IoT applications, multipoint-to-point traffic accounts for the majority of network traffic streams. 6BRs, which play an essential role in the network and offer an interface for connecting to the Internet, are the most popular MP2P destinations. RPL supports MP2P traffic with DODAG root connections to destinations. Root routers were used to construct upward paths when DODAG was installed. The recommended parent chain is used to construct default paths from nodes to root [84]. The key benefit of MP2P traffic is that it can use partial routing mode, which means that the node only has to store the destination, which is the DAG root.

6.1.2. Point-to-Multipoint (P2MP) Mode of Operation. RPL protocol also characterizes the P2MP operation, depicting traffic forwarded from the root to multiple nodes in a downward direction. RPL uses the DAO (destination advertisement object) control packet mechanism for destination advertising to support external P2MP one-way traffic, which is used for a small number of specific LLN-based IoT applications such as home and industrial automation. The DAO mechanism provides fewer routes in the DAG structure for destination access for routers. Routers deliver DAO messages to their parents or DAG root in one part to install downward directions [85]. In-network prefixes and ad addresses for each destination are given in the DAO messages. Each router that sends a DAO message to the root adds its address to a DAO message reverse routing path. For this reason, the source node can route traffic to its child nodes in the DODAG structure.

6.1.3. Point-to-Point Mode of Operation (P2P). RPL routing protocol offers routing structures considering two DODAG nodes. The 6BR must transfer packets to the destination when the origin is routed because of the P2P traffic support in the RPL network. Two instances exist: (1) if the destination node is at the same point as the sender node in the same propagation range, it can immediately transmit a message to the destination without passing it on to its parent. (2) The P2P mechanism depends on the presiding in the network's storage or nonstorage mode [86]. If not saved, routers do not store information about downward routes (no child information and just process the data source). Each packet must first be sent to the root through the DODAG upward route, after which it will be sent to its destination. Routers in storage mode save downward path routing information locally. If the destination is a router descendant, the message is sent to the router closest to the destination. If the destination is not a descendant, a message is sent to the parent node,

which sends the packet to its destination using the same rules as before. As a result, the packet will be transmitted from the child to the parent to connect the tree to the router, the source, and the destination nodes' first ancestor. The RPL routing protocol's network model is seen below. As a result, RPL distinguishes three kinds of nodes [86]:

6.1.4. Low-Power and Lossy Border Router (LBR) or 6BR. The root of a DODAG is a point of accumulation in the network that suggests the network's capacity to create a DAG. Between the Internet and the LLN, the LBR serves as a firewall (or edge router).

- (i) *Router.* A system that can both produce and send traffic is referred to as a router. This method of routing cannot generate a new DAG because it is relying on an individual
- (ii) *Host.* This term refers to a final system that can produce data traffic but not transmit it. DODAG is the most fundamental topological part of RPL. The DODAG root is a destination-oriented DAG in which a particular node called root is seen in Figure 2

The properties of the DODAG root are as follows [87]:

- (1) Usually serves as a 6BR. (2) Sink inserts data into a graph without causing it to rotate. (3) In DODAG, the node is usually the final destination, acting as a specific transmission point connecting the LLN to IPv6 networks. (4) Ability to create a new DODAG down to the root nodes.

Each node in DODAG is assigned a rank. According to the root "DODAG," a node's rank is specified as the node's status among the other nodes. In the DODAG structure, the root has the lowest rank value. In the downward direction, this rank will increase, and in the upward direction, the value will decrease. Therefore, nodes close to the root have lower ranks than their descendants or lower nodes in this structure, as shown in Figure 2. A DODAG's geometry is close to a tree-cluster topology, with all traffic being stored at the base. The DODAG architecture, on the other hand, differs from the cluster tree in that a node is based on both its parent (with higher rank) and other sibling nodes (with equal rank) [88]. In DODAG, rank is used to avoid and detect routing loops and identify parent and sibling nodes. RPL requires nodes to maintain a list of possible parents and siblings to be used if a parent's routing capacity is disabled. Each router defines a secure set of parents on a path to the DODAG root and assigns itself to a preferred parent depending on the objective function when constructing a network topology. The goal feature specifies how RPL nodes interpret one or more parameters within rankings and pick and optimize DODAG paths. It is also in charge of evaluating routing constraints and optimization goals and measuring rank based on basic routing parameters (such as latency, connection quality, and connectivity). The design of efficient target functions is still a work in progress. In one case, they used the expected transmission count (ETX) to pick a crucial path in RPL routing by successfully passing a packet over a connection. The path from a specific node to DODAG's root represents the path that minimizes the number of ETX from the start to the root [88].

FIGURE 3: RPL control message.

FIGURE 4: DAO message format.

6.2. *RPL Control Messages.* Figure 3 illustrates the structure of RPL messages, which are a new type of ICMPv6 control message. The RPL control message is made up of the following components:

Three fields make up an ICMPv6 header: type, code, and checksum. The body of a message is made up of a base message and many choices. The type area, set to 155 for RPL, defines the type of ICMPv6 control message (IANA approved). The kind of RPL control message is specified in the code area [89].

The RPL category area currently has four codes, each of which is explained separately below [27]:

DODAG information solicitation (DIS): the DIS message is mapped to $x00$ and is used to ask an RPL node for a DODAG knowledge object. In adjacent DODAGs, DIS can be used to analyze neighboring nodes. Flags and fields for potential use are used in the new DIS message format.

DODAG information object (DIO): the DIO message is mapped to $0x01$ and exported by the DODAG root to construct a new DAG, which is then sent via the DODAG structure rendered process. The DIO message contains network knowledge that helps a node discover an RPL Instance, learn its configuration parameters, choose a DODAG parent packet, and keep DODAG updated. Figure 3 depicts the DIO base object shape. The below are the major DIO object fields [90]:

- (i) RPL instance ID, an 8-bit data that starts with the root DODAG, indicates the instance RPL ID of which DODAG is a part
- (ii) Version number shows the version number of a DODAG, which usually increases with each update

of the network information, keeping all nodes up to date with the new update

- (iii) A 16-bit field defines the rank of the DIO message sender node called a *Rank*
- (iv) *DTSN* is an 8-bit flag that is used to hold downward paths open
- (v) *G* is a flag that specifies whether the current DODAG satisfies the application's intent
- (vi) The mode of operation (MOP) defines the RPL instance's operating mode, determined by the root DODAG

There are four different modes of service, each of which serves maintenance and multisegment downward routes differently. By default, upward paths are supported. Each node connecting to DODAG must behave as a router when dealing with the MOP; otherwise, it will be regarded as a leaf node [90].

- (i) *Prf* is a 3-bit field specifying DODAG root precedence over other DODAG roots. Its value ranges from $0x00$ (the default) to $0x07$ (the highest priority)
- (ii) The *DODAGID* is a DODAG root-listed 128-bit IPv6 address that recognizes DODAG uniquely. Finally, an *Options* field can be present in the DIO base object

Destination advertisement object (DAO): the DAO message is mapped to $0x02$ and is used to relay reverse track information to record upstream nodes. DAO messages are

sent to add routing tables with their children's prefixes by some node other than DODAG root and advertise their child's addresses and prefixes. After this DAO message passes through the default DAG path from a specific node to the DODAG root, a complete path between the DODAG root and the DODAG node is created. The DAO base object format is shown in Figure 4. As the figure shows, the DAO Message's key fields are [91]:

- (i) *RPLInstanceID* is an 8-bit data representing the DIO's RPL instance ID
- (ii) Flag *K* indicates that a DAO message needs authentication
- (iii) The exponential number for each DAO message is the *DAOSequence*
- (iv) The 128-bit field *DODAGID* is a DODAG root specified field that specifies a DODAG. Only if flag *D* is equal to one is this field involved

Destination advertisement object acknowledgment (DAO-ACK): the Unicast Message is sent by a DAO receiver (DAO parent or DODAG root) in response to the DAO Message provides *DAOSequence*, *RPLInstanceID*, and termination status information. Do not forget. Nodes greater than 128 mean inaccessible, and a node has to choose a replacement parent [92].

6.3. RPL DODAG Manufacturing Process. The DODAG graph is created step by step. First, as Figure 4 indicates, the root plays a DIO message. The RPL nodes are expected to detect an RPL instance, use its parameters for setting up, choose a parent set, and construct a DODAG graph. This message contains information. The DIO packet recipient node will add the DIO transmitter to its parent list in the routing table and calculates the rank according to the OF inserted in the received DIO. The node's rank value matches its location in the root graph and consistently exceeds its parent rank to ensure the graph is distant from its existence. DIO messages are then updated to the neighbor and sent to him. The node selects a preferred parent based on the parent list and is used as the default gateway to send data to the DODAG core [93]. Both nodes involved in the DODAG graph have a standard ascending path to the root at the end of this step. All preferred parents compose this path. DIO messages are transmitted intermittently using the trickle [94] algorithm depending on the time set to maximize the network-related control messages' transmission frequency. By playing a DIS message to ask for a DIO message from your neighbors, DAO messages are used to build pathways downward. Router nodes in the DODAG structure can administer nodes' routing tables according to the type of service defined in the DIO control packets. For maintaining downward paths in an RPL instance, the RPL routing protocol has two operating modes [93]:

- (i) *Storage Mode.* In this situation, the child sends a DAO message unicast to the preferred parent, storing the content of the DAO messages received by his children before submitting the new DAO packet by

gathering accessible information. The multipart mode may be allowed or disabled in storage mode

- (ii) *No Storage Mode.* The DAO message is sent unicast to the DODAG root in this mode. As a result, middle parents do not save DAO messages; instead, they keep their addresses in the stack of the received DAO packet picture path and send it to their preferred parent. Consequently, no parent stores their child nodes' address in this case, and only the root, which receives all DAO packets, can store and manage all downward paths [95]

6.4. RPL Repair Mechanisms. Inconsistencies and correction loops: The RPL routing protocol integrates loop avoidance, inconsistencies detection, and DODAG correction. As the parent nodes' rank value increases, they approach the root in the DAG structure; the child nodes also tend to select the lower-ranked (higher value) parent as the preferred parent; hence, the infinite counting problem in RPL occurs. It cannot be reconnected to another node because it is broken. Of course, the value of both parents and children does not stop increasing [96]. The RPL routing mechanism prohibits loops in the DODAG structure by limiting the amount of rank increase allowed. If the node could not recognize the rank property, the loop would happen, so we can say that DODAG is a graph without a cycle. An outgoing node must declare a finite rank below—DODAG to prevent this. A different process can be used for the outgoing node, creating an intermediate DODAG and then reconnecting to the initial DODAG. The data route validation function in the routing protocol may also identify anomalies [96]. The routing information in packets is contained within an RPL option carrier in the IPv6 step-by-step process. Here are some definitions of flag:

- (i) Flag "O" down indicates the expected upward or downward direction of the packet. When this flag is activated, the router forwards the packet to a child node using downward routs, or vice versa; the packet will be sent to the parent with a higher value rank in the upward direction towards the root
- (ii) The "flag" R "error-rank" signals the presence of a rank error. The rank error occurs when the rank value and situation of a packet containing the flag below are not adjusted

"Flag" F "error-forward" shows that a node cannot transfer packets to the destination in the case of downward packets. RPL nodes can trigger correction processes when anomalies are observed. These structures would also sustain the network's topology in connection and node failures [97]. A preferred parent is not usable; the local correction process requires choosing another route for routing packets and selecting another parent node among its parent record. It can additionally route data packets from another relative (neighbor) node, like the same rank node. It cannot be ideal for replacing this path. This locally efficient correction function helps the network to converge in an appropriate time frame. As various inconsistencies malfunction due to the

local correction mechanisms, the DODAG root may begin to correct globally by increasing the Number of DODAG graphs. Then, the RPL network is completely reconstructed [97].

6.5. Trickle Algorithm. DIO messages are propagated quasi-periodically using an algorithm called trickle. The trickle algorithm is a transfer scheduling algorithm for the initial local communication between nodes in a network based on a stable model. When a network is stable, nodes exponentially reduce their communication rate, sending trickle messages in just a few packets per hour. In contrast, when a node detects an inconsistencies, it acts with quick trickle messages to resolve the inconsistencies [98]. Initially, the trickle algorithm was proposed for propagation and maintenance in WSN. It has been shown that it can be used for various purposes such as control traffic time, multicast propagation, and path discovery [99].

The Internet Engineering Task Force (IETF) standardized the trickle algorithm to regulate DIO messages' transmission to generate network graphs in RPL. The trickle divides time into non-conformal intervals so that the smallest distance is I_{min} and the size of the highest distance is I_{max} . At each distance, each node tries to send its trickle message based on trickle rules. This algorithm works based on some parameters, variables, and regulations. There are three parameters to configuring the trickle algorithm that is described below [5, 99].

It makes vulnerabilities and protection against attacks difficult. The RPL protocol specifies the number of security mechanisms. It combines local and global processes and loop intrusion and identification techniques. As discussed earlier, it also sets two safeguards for data packet encryption. However, their protection on the communication layer and the transmission or applications level is focused on these networks' standard construction. The following will be thoroughly investigated on attacks against the RPL protocol. In two pages, special RPL attacks and RPL-related routing attacks, RPL attacks are listed in detail [5].

7. RPL Protocol-Specific Attacks

7.1. Internal RPL Attacks. This section describes in detail the specific attacks associated with the RPL protocol that is considered as part of internal attacks:

Storage routing table overhead attacks: as long as storage mode is allowed for these nodes, the RPL protocol is active, conveying routers created by RPL and holding routing tables. Increasing the volume of exchange routing tables in the network topology causes significant overheads that can also promote fake routes by DAO. This saturation inhibits new legal paths and affects network capacity and the possibility of memory overflow [100].

Rank increase attacks: these attacks include purposefully raising an RPL node's rank value to construct a network loop. A rank value in an RPL network is determined by each node and corresponds to the root node's location in the graph structure [101, 102]. The nodes' rank in the downward paths will be continuously increasing to stabilize the DODAG graph structure. Each node's calculated rank must be higher than its parents' rank. The nodes must first delete the parents

in their routing table with values higher than the currently calculated rank to switch their parent and change the rank [103]. In the DODAG structure, each child node selects a preferred parent from their parent list to minimize sending data cost towards the destination. An attacker can advertise a fake rank higher than expected on the network. Therefore, if the new parent's DODAG rank is lower than the previous one and there is no loop prevention mechanism in the RPL, multiple loops will be created in the network. The loop correction function must give many DIO messages (trickle timer reset) and provide a long convergence period in this situation. When node batteries are depleted, and the RPL network gets congested, the attack is part of a resource utilization attack. If the number of ranks increases by each node in the DAG structure can be recorded, both inconsistencies in the graph configurations can be detected. The amount of this type of attack can be reduced. It is worth mentioning that if a node does not have any objective function matching or cannot accommodate the amount of traffic it gets, it can legitimately raise its rank score. After all, the new OF must have mechanisms to detect the intrusions with the loop or update the graph structure when the loop occurs. RPL has inherent capabilities for loop detection or prevention by validating data transmission paths [103].

DAG inconsistencies attack: inconsistency in the DODAG structure is detected by a node when the packet it receives from a higher rank node is set to "O" in its flag bit. For example, though the packets' path does not fit the rank relation, it may cause a graph loop. This problem is controlled by the flag "R" error-rank bit. Since contradictions with a node are found, there are two possibilities [104, 105]:

- (1) If the error-rank flag is not activated, the node first fixes its value and then transmits the data packet. It is not just a matter of route inconsistency; it is a severe condition to the RPL network
- (2) Setting the "R" bit in the received packet means a rank error. The packet receiving node will be ignored if it is already set, and the timer will be reset. In case of this phenomenon, control packets will be sent frequently. The only thing a malicious node does is change the flag or apply a new flag to the header. This attack's immediate result is that the goal node's DIO trickle timer will have to be reset. In this situation, the node transmits DIO messages constantly, causing local chaos in the RPL network, draining the nodes' batteries, and affecting connection availability. All the attacking neighbors are involved in this attack, and therefore, unnecessary traffic is processed. Furthermore, by altering lawful traffic, the aim node discards all packets. It creates a BH that divides the network's components. The trickle timer's reset rate during an RPL option has been limited to no more than 20 resets per hour to reduce the flooding caused by this attack. Also, instead of a fixed threshold, two network feature solutions are used. The first solution is an adaptive threshold with fixed parameters. Another form is a rank attack. The attacker does

not search the rank relationship for a malicious node and does not set the “R” flag if anomalies are identified. The distinction between the DA (destination advertisement) inconsistencies and the DA inconsistencies is that the intruder cannot use the flags to render false circles. For real circles, however, it does not choose any solution. If they occur, the consequences would be identical [105].

Version number attacks: the version number field is an essential part of a DIO control packet that does not change value when sent and received in the DAG structure. If there is a need for a general fix to the graph structure, its value will be increased by the 6BR. If this field value does not change in the received DIOs from a node, the sender is not yet connected to DODAG and cannot be used as a parent. The attacker may cause instability in the graph structure by manipulating and increasing the version number field’s value and retransmitting the DIO to neighbors. The entire DODAG graph will be rebuilt unnecessarily as a result of such an attack. This attack will result in a lot of loops and data packet failure as a result. Unnecessary sequential graph reconstruction also significantly increases control message overhead, node resource loss, and network congestion. VeRA’s security mechanism is provided to prevent vulnerable nodes from root-forging and sending an illegal incremental number. This approach allows the use of a hash-based authentication method. In this case, a node will quickly determine if the root node or another malicious node has changed the version number and unable to usurp DODAG’s origin identification [106].

Routing table falsification: routing information may be formatted or changed in a routing protocol to spread falsification paths to other nodes. By manipulating or forming DAO control messages to construct false downward routes, this attack can be carried out on the RPL network. When storage mode is allowed, this is possible. A malicious node, for example, advertises routes to nodes that are not within the DODAG. The following network is configured since the target nodes have incorrect paths in their routing table. As a result, the path may take longer to complete, packets may be discarded, and the network may become overburdened. The RPL protocol is yet to investigate this attack [107].

Routing information broadcast attacks: in this type of attack, each node in the DAG structure stores control packets received from valid nodes afterward publishes them on the network. Since the topology and routing paths of complex networks shift often, this attack is very disruptive. Thus, routing information broadcast attacks disrupt the correct network topology and persuade nodes to update their routing tables with incorrect and outdated information. The RPL protocol uses sequence counters to guarantee that routing information is new. The version number is integrated into DIO packets. The current path sequence stores alternative routes information in DAO packets [108].

LR attack: in an LR attack, the attacker regularly sends an LR message without connection quality problems. This phenomenon leads to LR around the node. This attack affects the packet delivery ratio more than other attacks. It increases the

number of control packets exchanged and end-to-end delay (E2ED). It also increases the EC of the nodes [109].

Neighbor attack: in this attack, the attacking node broadcasts the DIO packet received from its neighbor without any change. The node receiving this packet may think that a new neighbor has sent this message. He may want to add this node to his list of potential parents or choose him as the preferred parent if that node may not be within the range of the victim node [110].

DIS attack: DIS packets are used to receive network topological information before connecting to it. If the attacker broadcasts the DIS messages, the receiving node of this packet will reset its DIO timer. If the attacker replays DIS messages, the recipient sends a DIO packet in response [60, 111].

Worst parent attack: this attack is called the rating attack, which systematically chooses the worst preferred parent based on the objective function. As a consequence, the course is not optimum, and productivity suffers. This attack is one of the most violent and dangerous attacks on RPL because the child nodes need their parent to route and direct their packets. The neighboring nodes cannot track and detect this type of attack [53, 112].

Storage DAO inconsistencies attack: this type of attack occurs when the DAO control packet detects that the node is set in the downward path. However, in the child node routing table, this route is invalid [113]. DAO inconsistencies loop recovery is a method provided by RPL for fixing these inconsistencies. By sending a flag “F” error forwarding in data packets, RPL router nodes will deprecate downward paths by signaling that a child node could not deliver a packet. The packet is returned to its parent with the active “F” flag, forcing it to connect with another neighbor. Packets sent in downward paths may barely come back to their initial position. This event will happen when the router sends a packet to the parent whose flag bit “F” is fixed and the flag bit value “O” is not set. Suppose the parent node receives a packet with the “F” flag set. In that case, it erases the value entered in that flag and tries to send it to another neighboring node according to its routing table information. The procedure is replicated if the alternative neighbor already has an inconsistent model. The purpose of this attack is to divert the nodes from accessible downward paths. This attack also leads to segregation and instability, and additional congestion if packets must be sent from the following optimal routes. Eventually, the child’s nodes become delayed and hungry. To reduce this attack’s impact on the network, 6553 RFC suggests that the rate of discarded downward routing entries is limited to 20 times per hour [113].

Decreased rank attacks: in the DODAG structure, the further we go to the root, the lower the rank, and the nodes close to the root need more control because they can attract more traffic and become hotspots network. Nodes in the RPL tend to reach a lower rank and a position closer to the root. A malicious node with lower rank ads can attract many nodes to itself or the DODAG Instance and cause an imbalance in the graph structure. An intruder node can change its rank value by forging DIO packets in the RPL [114, 115].

7.2. RPL Protocol-Related Routing Attacks. This section describes in detail the RPL protocol-related attacks, which are another part of internal attacks:

HF attack: HF attacks are packets that one node sends to other nodes to connect to the network. With all broadcast packets of high signal strength and good routing metrics, an attacker can identify itself as a neighbor of many nodes (or even the entire network). However, suppose the nodes are far from the attacking node. In that case, their messages will not reach their destination to connect to that node because the attacker is not within their range. In RPL, this attack occurs when an attacker uses DIO packets for advertising a DODAG. If encryption is used on the network, the attacker must capture a network node to attack with the keys in hand. If the nodes' topology information is known, filtering incoming packets from remote nodes will reduce this attack's impact on the RPL. The RPL itself can significantly reduce this attack's impact within 10 minutes, but some anomalies remain in the network [116, 117].

Sinkhole attacks: there are two phases to this attack: the malicious node is first used to draw a vast traffic volume by displaying forged results (for example, better up and down quality links). Then, after unlawfully processing traffic, it corrects or discards it. The attack can be conveniently carried out in RPL networks by controlling the rank value. Because of false ads, other nodes often chose the malicious node as their selected parent, lowering efficiency. As a result, the routes are not network-optimized. This attack alters the topology of the network and decreases its efficiency. Also, a BH attack is where an attacker tries to divert all network traffic [118, 119].

WH attacks: an off-band communication between two nodes using wired or wireless connections is used in this attack. WH can be used to deliver packets more efficiently than conventional routes. An intruder intercepts packets transmitted by nodes on one side of the network and distributes them to nodes on the other side. This attack is simple to carry out in wireless networks since the intruder will transmit the requested traffic to himself through the WH and decrypt all wireless transmissions. In this type of attack, the intruder uses a tunneling system to send routing information from one part of the network topology to another, thus falsifying the data transmission path during the routing process. If nodes are in the same neighborhood, they can see each other even if they are far apart. As a result, they can create non-optimal paths based on objective function [120, 121].

BH attacks: a malicious intruder throws away all packets to be sent in this type of attack. When combined with SH attacks, this attack can be highly damaging, causing massive traffic to be lost. This attack is classified as a DoS attack. The attacker will detach many nodes from the network if they are in a strategic location on the graph. Gray hole or limited forwarding attacks are another forms of attack in which the attacker only throws a section of the network [122, 123].

Sniffing attacks: in the sniffing process, the attacker can listen to the traffic exchanged through various wired and wireless networks and capture or distort their data without informing the legal sender and receiver. An attacker may use a hacked computer or directly steal packets from shared media on wireless networks to carry out the attack. Partially

FIGURE 5: Techniques used in data mining.

topological information, routing information, and data content can be derived from intercepted packets. Suppose an attacker eavesdrops on control messages in RPL networks. Therefore, the attacker can exploit neighboring nodes' configuration information such as rank, DODAG ID, and RPL instance version number. Intruders can achieve a local view of the network topology, addresses, and packets content exchanged between source and destination by sniffing the network and eavesdropping packets sent and received. This attack is brutal to detect, owing to its static nature. When an unknown intruder is involved, the best way to avoid interception is to encrypt communications [124, 125].

Traffic analysis attacks: these attacks use the features and patterns of traffic on the connection to collect routing information. Furthermore, if the packets are secured, this attack can be carried out. Attacks like sniffing accumulate information about the RPL network, define the parent-child relationship, and partially view the topology. The attacker's rank decides the attack's result. It will process much traffic if it is close to the root node. As a result, it will gather more data than when the node is on the DODAG's edge [126].

Identity attacks: these attacks, also famous as CloneID, occur when a malicious node mimics the legitimate node identity. By providing root node access, which is the critical point in creating and maintaining the DODAG topology and managing routes information and data exchanges, the attacker can listen to traffic and, by forging root identity, launch various attacks such as Sybil on the network. In Sybil attacks, the network performance and services can be disrupted using a malicious server that uses physical nodes' logical inputs and forges their identities [127, 128].

8. Data Mining (DM)

The method of finding fascinating trends in vast volumes of data is known as data mining. It is an intriguing paradigm based on objective laboratory evidence, is novel and theoretically beneficial, and is easy enough for humans to understand. Such intriguing patterns represent information. Many people mistakenly believe that DM entails extracting information from data. Others consider DM to be only one

stage in the process of information discovery. The following steps are known to be part of the information exploration process [127]:

- (1) Data clearance: deletes noise and incompatible data
- (2) Data integration: combining multiple data sources
- (3) Data selection: recovery of data related to analysis from the database
- (4) Data conversion: convert data to a form suitable for DM, such as summary by integration
- (5) DM: using methods to extract data patterns
- (6) Pattern evaluation: identification of correct knowledge-based patterns according to measurement criteria

Steps 1 to 4 are various data analysis methods. The data is prepared for mining and collecting information using imaging techniques and knowledge presentation to show the customers found knowledge. The user engages with a knowledge base during the DM process. The user is provided with the found patterns, which are then saved in the database as new information. This diagram illustrates DM as one of the phases in the information discovery process. It is crucial since it detects secret trends that can be analyzed. On the other hand, DM refers to the whole method of knowledge discovery in business, media, and science (perhaps because it is shorter than the term knowledge discovery from data).

Consequently, a general view of DM success is taken into account: DM is the process of extracting useful information and patterns from massive volumes of data. Databases, data servers, the Web, and other data archives that are dynamically streaming across the environment are examples of data sources [129]. As one of the most application-oriented examples, technologies for data processing data mining utilizes various techniques from other areas. Statistics, ML, pattern analysis, data warehousing and database systems, knowledge recovery, illustration, high-performance computing algorithms, and a wide variety of other technologies are only a few examples. The technologies used in DM are depicted in detail in Figure 5.

8.1. Machine Learning (ML). Machine learning is a growing field that analyzes how computers learn or improve their performance based on input data. This science seeks to automate the recognition of complex patterns and intelligent decision-making [130]. For example, in a post office, ML can identify handwritten postcodes on envelopes after receiving several samples of different codes.

Supervised learning: in this ML field, the dataset under training uses different labels to classify the instances. For example, various postcode images with specific machine understanding concepts are used for classification-based supervised learning [185].

Unsupervised learning: this ML model is generally equivalent to clustering, with the received instances having no label classes. Unsupervised learning uses clustering to identify data classes [131, 132]. The image collection contains

the postal code handwritten digits is delivered as input by unsupervised learning. Because under training data classes are unlabeled, the learning model is inefficient and cannot analyze the received image clusters' semantic concepts and are only adapted to different digits.

Semisupervised learning: this ML technique allows for labeled and unlabeled instances to be used while learning the model. We can use labeled instances to learn data classes in one approach and benefit unlabeled cases to correct class boundaries in something else [133, 134]. One number of instances can be presumed to belong to the positive class. In contrast, the rest of the instances can be assumed to belong to the negative class with two classes. The decision boundary can be defined more accurately using unlabeled instances. Also, two positive instances in the upper right corner, despite being labeled, can be identified as noisy or skewed data [133].

Active learning: active learning is a computer learning system that directly encourages users to engage in the learning mechanism. This learning model asks for a specific user (e.g., an expert in a subject) to label an instance, which may be a collection of unlabeled instances or created by the learning software. Given the small range of occasions that can be questioned for marking, this strategy improves the model's accuracy by incorporating information from human users. DM and ML have many parallels. ML also relies on model accuracy when it comes to classification and clustering. DM emphasizes the reliability and scalability of extraction techniques on massive data collections and methods used in structured data and the development of new and alternative methods [135, 136].

8.2. Classification. The task of discovering a model (or function) representing and separating data classes or concepts is known as classification. The model's extraction is based on evaluating a collection of experimental results (data objects with class labels). Next, the model simulates a tag class of data targets for items not classified in a particular class. The learning stage (where a classification model is created) and the classification stage (where the data is classified) are also part of the data classification process (the expanded model is adopted to anticipate delivered data class tag) [137]. A classifier is constructed in the first step to represent a predefined set of data classes or concepts. A classification algorithm produces this learning step (or learning step) by classifying a classifier by evaluating and learning from various database instances and class tags. The model is then used to classify the data in the second process. Before applying the model to the actual results, the accuracy of the category forecast must be determined. A series of trials accomplish this. Models can be created using various tools, including classification principles, decision trees, mathematical formulas, and neural networks [137, 138].

8.3. Data Mining in the Intrusion Detection System. Data mining techniques can be classified based on differences in performance, model representation, priority criteria, and algorithms. In the field of IDSs, one of the main functions of the models is classification. The classification technique is used to separate data as normal, destructive, and offensive.

Three decision tree classification techniques, support vector machine, and Bayesian method are used in this project, each explained separately [139, 140].

8.4. Decision Tree. A decision tree is similar to a flowchart in that it has a tree structure. Internal (nonleaf) nodes in this tree reflect a property test. Each leaf node (or end node) represents a class, and each branch represents the test output. The decision tree classifies in the following way: the values of the instance X attributes in the decision tree are evaluated and reviewed for an instance like X that does not have a class name [141]. The path from the root to the leaf node containing the class predicted for instance X is generated. Classification laws can be quickly translated from decision trees. Decision tree categories have been prevalent due to the lack of specialized knowledge with parameter setting. They are very suitable for discovering exploratory learning. Decision trees can also be used for multidimensional data. The knowledge gained is in the form of a visual tree and is generally easy for humans to understand. The steps to learn and classify the decision tree are straightforward. Decision tree classifiers are usually very reliable. However, efficient usage is dependent on the data available. Attribute selection parameters are used during tree creation to choose the attribute that best separates instances into distinct classes. Many branches of decision trees can display noisy or out-of-date training results. These branches can be identified and deleted using tree pruning to increase classification accuracy on unobserved data [141, 142].

8.5. Support Vector Machine (SVM). SVMs are a classification system for linear and nonlinear results. In brief, SVM is an algorithm that operates like this. This algorithm employs a nonlinear mapping to convert educational data to a higher dimension [143]. The cloud looks for the best linear page in this new dimension (i.e., the “decision boundary” separating one class’s instances from another). With a convenient nonlinear mapping large enough, data can be separated from two separate classes via a cloud [144]. The SVM algorithm uses backup vectors (“instructional instances”) and margins (defined by backup vectors) to find the cloud page. These concepts will be explained further. Because of their ability to model dynamic nonlinear decision boundaries, SVMs are highly accurate and less vulnerable to overemphasis than other approaches, considering their slow training time. The discovered help vectors often serve as a full explanation of the model that was studied. In numerical forecasting and grouping, SVM can be used [144].

8.6. Bayesian Method. Statistical classifiers include Bayesian classifiers. This classification can be used to predict the possibility of an instance entering a specific class. This classification is based on Bayesian philosophy, which will be discussed further down [145]. The primary Bayesian classifier’s efficiency, also known as the naïve Bayesian classifier, is comparable to decision trees and selective neural network classifiers in classification algorithms studies [144]. When used for massive datasets, Bayesian classifiers demonstrate high accuracy and speed. The influence of one attribute’s value on the given class mark

is believed to be independent of the importance of the other characteristics in basic Bayesian classifiers. This assumption is called “class-conditional independence” and simplifies the calculations. For this reason, in its naming, “naïve” has been used in a superficial sense [144].

9. ML Methods in RPL Protocol

9.1. Feature Selection Methods for Building an Intrusion Detection System. In the ML process, feature extraction and selection are two crucial stages. ML models are trained using features. FS methods are beneficial for determining a subset of features within a dataset that decreases processing time and increases classification accuracy. In particular, three different selection methods exist (1) method of filtering, (2) method of wrapping, and (3) method of embedding [124, 146]. Filter methods preprocess data by calculating and predicting the target feature based on the relationship between features. CFS (correlation-based function selection) is a heuristic search technique that pairs the feature evaluation formula by calculating the necessary correlation between the features and the class identifier. The critical goal of feature discovery is to identify a subset of strongly associated features with the class identifier but not with each other. The feature reduction methods for IDS have received much attention, mainly when using the KDD dataset [147–149]. Shubhangi and Meenu have created an IDS that uses a heuristic technique to detect denial of service attacks and filters the attributes. Using the KDD dataset [150], they applied the concepts of knowledge benefit, gain ratio, and correlation. Swapnil and Sanyam [151] also take a similar view. Sun and Kasongo developed their IDS using filter-based methods [152]. This paper goes into greater depth on how FS can enhance classification accuracy and overall IDS efficiency.

CHA-IDS is a compression header analysis-based intrusion detection framework built for RPL [64]. In their research, Stephen and Arockiam [96] discussed rank inconsistency attack (RInA) mitigation. The rank value is tampered with in RInA to make the network vulnerable. E2V, which consists of three stages, has been proposed to alleviate this rank attack. Pursuing and mitigating RInA-based vulnerabilities such as BH, SH, and limited forwarding is the primary goal of this pattern. In the first action, the rank must be approximated. Next, malicious nodes must be identified and removed. The energy level is benefited to discriminate authentic nodes from spurious ones also distinguish rank inconsistencies. Nevertheless, only various types of rank attacks seem to be the target of this detection model.

Neerugatti and Reddy [153] provide a similar algorithmic mechanism that RPL uses to discover rank attacks. The k -nearest neighbor algorithm represents the MLTKNN solution for ranking attacks. “The rank attack in the RPL protocol is the physical location of the node about the boundary router (root node) on neighbor nodes,” according to [153]. When constructing a DODAG graph in RPL, the attacking node, by tricking the 6BR, will be able to create a path through advertising a fake rank. MLTKNN has been suggested to detect this malicious or intruder node. The proposed methodology is tested with 30 motes in the Cooja simulation.

However, it is worth noting that only the rank attack is discussed in this job. Shin et al. proposed a new IDS system for anomalous intrusion discovery devices in RPL [154]. This new solution can detect packet drop attacks in RPL and detect the fake packets falling by the network's data packets losses. Following [154], "nodes in RPL retain a packet distribution ratio of their forwarding links to compute a routing metric ETX." The suggested approach uses this value to obtain nodes' usual behavior since this phenomenon indicates the network's data packets loss in those communication paths [154]. For their intrusion detection system, the authors used the Cooja emulator, which is used to evaluate Contiki-based systems' performance. "The findings of the evaluation suggest that the approach is effective at identifying malicious packet dropping attacks." This scheme is specifically designed to detect legitimate packet-dropping attacks. It does not protect against RPL or WSN other attacks, as previously stated. Bhandari et al. [155] proposed a "congestion-aware routing protocol (CoAR) that depends on the selection of an alternate parent to ease network congestion" [155]. By combining various routing metrics and using the "multicriteria decision-making (MCDM)" mechanism by the child nodes, the authors' proposed solution selects the preferred parent node among many candidate parents. The proposed method uses the neighborhood index metric to break the tie of routing points.

Yavuz et al. [156] have proposed a modular deep learning approach using a seven-layer structure of ANN to detect different types of attacks such as version number, HF, and decreased rank attack in IoT. In this research, multilayer vision and NB classifiers are used for analysis. Neerugatti and Reddy [157] suggested WH attack detection and introduced a novel method (ADWA). The authors' ADWA approach uses an acknowledgment mechanism to detect WH in RPL. Contiki-Cooja simulation outputs using TelosB sky notes reveal significant improvements in latency, packet distribution rate, and detection of WH attack metrics. Although the above research offers excellent countermeasures against attacks in the RPL, they do not provide any strategies to detect and prevent both rank and WH attacks simultaneously.

9.2. Available Improvements for RPL Using Intelligence Approaches. Objective functions (OF) are responsible for setting routing rules for the RPL protocol; however, structuring is not necessary for OF. As a result, RPL gave researchers the freedom to improve or evolve the OF following the requirements. Various routing approaches [104, 158] have been suggested to enhance network efficiency by improving multiple performance parameters, such as packet distribution ratio in the network, transferability (throughput), EC, cost of operation (overhead), and other cases. The precise measurement of connection efficiency is an essential consideration for connectivity in a wireless network. Ancillotti et al. provide a mechanism for improving the measuring probe based on reinforcement learning (RL) for RPL called RL-based link quality estimation (LQE) [159]. The RL-probe feature uses asynchronous mode for LQE, which puts it in both proactive and reactive phases alongside synchronous. The obtained sig-

nal strength indicator (RSSI) near the side of the ETX metric is analyzed in the proactive process. The reactive phase performs an RPL rapid local fix to conducting the LQE. The synchronous mode in LQE divides the nodes into clusters to enhance probing. It also benefits from the unique prioritization of created classes based on a multiarm bandit (MAB) to improve probing. The author of RL-probe streamlined the probing process but did not suggest tuning the relation metric. Clustering also added to the power overhead.

Researchers have developed a context-aware method for LB in RPL (CLRPL) [160] for IoT infrastructures with significant and complex traffic loads. Their new OF, called CAOF, solves the thundering herd phenomenon (Herd Decampment Phenomenon) [87, 161] in the DODAG structure and calculates each node's rank based on its parent's rank, residual energy, and ETX. Their other new OF, called CARF (context-aware routing factor), uses the parent chain to balance the load and residual energy instead of relying on a single parent. Their proposed method has a significant DIO overhead but avoids the illusion of equality with the preferred parent selection and improves network resource consumption and packet losses.

To overcome the congestion problem in parent nodes, which arises from the nonsymmetric distribution of child nodes, the authors using the MCDM Mechanism, propose a new congestion-aware objective function (CoA-OF) for RPL (CoAR) [155]. This objective function uses the three ETX, RE, and QU metrics based on the technique for order choice mechanism by similarity to ideal solution (TOPSIS) [162], to select the preferred parent. Congestion detection is performed using a comparative threshold solution for buffer occupancy calculated depending on past and present traffic. The proposed solution imposes more energy overhead on parents' regular settings. It improves packet delivery rates, power consumption, and throughput in high traffics.

The authors have proposed a multiobjective OF that depends on Quality of Service (OFQS) [163] for RPL that automatically adjusts various instances based on the criteria set out in the smart grid (SG) specifications. This new OF uses delay, ETX, and three-mode power state (adjusted depending on the nodes' remaining resources) to make routing decisions. OFQS assigns a weight to each route based on these three criteria. They also divided the traffic into three categories: essential, noncritical, and seasonal. The path with the shortest delay (between 1 and 30 seconds) and more significant than 99.5 percent efficiency is chosen by critical traffic. Noncritical traffic, on the other hand, takes the path with the shortest delay and best reliability of 98 percent. Periodic traffic takes a route with a modest delay (about 5 minutes to 4 hours) and a 98 percent reliability. E2ED, PDR, and network lifespan are all improved by the suggested approach. On the other hand, their tuning criteria are locked in stone and cannot cope with a complex network. The ML approach can be used to solve it.

The authors in [164] provide a versatile objective function. The data forwarder is selected based on a combination of forwarding delay, ETX, and EC criteria for applications that require real-time data exchange, data durability, and energy efficiency. This OF sets a specific weight for each metric and

TABLE 5: Existing enhancements in RPL with intelligence methods.

References	Objectives	Technique	Advantages	Disadvantages	Implementation
Ancillotti et al. [159]	Predicting the quality of links	Multiarm bandit (MAB)	Packet delivery ratio (PDR) improvement	Imposes more control overhead	Cooja and IoT testbed
Taghizadeh et al. [160]	Increasing the high-speed network lifetime and decreasing packet losses	CAOF and CARF OF's	LB and enhances network lifetime	Increases overhead due to DIO control packets	Cooja
Bhandari et al. [155]	Mitigating the congestion problem in communication routes	TOPSIS	Boosts PL, PRR, EC, and E2ED	High overhead and frequent parent nodes switching in heavy traffic	Cooja with Contiki 2.7
Nassar et al. [163]	To benefit from a superior smart grid infrastructure	OFQS	Improves network lifetime, PDR, and E2ED	The problem of communication links redundancy and fixed setting specifications	FIT-IoT lab
Lamaazi et al. [164]	Designing a protocol with real-time data transmission, high reliability, and optimal EC capabilities	Weighted metric	Improves EC and PDR	Imposes more control overhead	Cooja with Contiki 3.0
Cao and Wu [165]	To pick the right parent, use the weighted distribution principle	Chaotic genetic algorithm	Throughput, E2ED, and residual energy improvement	Imposes more network overhead	—
Lamaazi and Benamar [166]	Using a combination of metrics, find the best route	Fuzzy logic	Increasing network lifetime and PDR Decreasing overhead	Failure to configure metrics during their combination	Cooja with Contiki 2.7
Bahramlou and Javidan [167]	To maximize employing network resources	Aggregation	Improves PDR, DIO overhead, and retransmissions	The problem of congestion in dense networks	Cooja
Zier et al. [168]	Decreasing control overhead	Multi-constraint OF	Enhances E2ED, EC, and routing overhead	Delaying network convergence time	Cooja
Fabian et al. [169] Ghaleb et al. [170]	Benefiting from LB mechanism to improve network stability	Fuzzy logic LB	EC, throughput, LB, and PDR improvement	Prospecting the network loops, increasing EC and network convergence period	Cooja
Kechiche et al. [171]	Including quality of service to get time-sensitive real-time and time-sensitive applications	Fuzzy logic	Improves PDR and delay	Increasing EC	Cooja

obtains the composite additive metric. The weight is defined based on the application type or the form of traffic. This additive composite metric is applied to each entry of the parent node routing table. For each entry, the parent table is reconstructed according to the calculated metric. The proposed solution improves the packet delivery ratio and EC and imposes additional overhead on the system. RPL has been enhanced with the chaotic genetic algorithm (CGA) [165]. This algorithm's fundamental goal is to use chaos and genetic algorithms to improve the parent selection process. CGA improves the search by using a genetic algorithm's global search efficiency to find the best solution using chaotic algorithm ergodicity. A composite metric is a combination of HC, residual energy ratio, ETX, and queue length. Each metric

has a specific weight. A noisy genetic algorithm optimizes the weighting factors in CM to choose the correct parent. This genetic algorithm boosts residual energy, E2ED, and performance rate metrics. The network overhead, on the other hand, was not taken into account in this analysis.

Lamaazi and Benamar [166] suggest a new objective function called OF-EC. To make routing decisions, OF-EC uses a fuzzy logic technique. ETX and energy use (EC) are merged in the OF-EC. By choosing the right relative, the proposed OF will minimize EC and packet loss. It does, however, raise the pace at which parents shift. Bahramlou and Javidan [167] use the aggregation method to make the most scarce capital. Researchers estimate the number of children to calculate each parent's rank, a technique that, in heavy traffic,

FIGURE 6: Number publications since 2012 based on the web of science.

FIGURE 7: Publication types published with "RPL protocol" topics.

FIGURE 8: Publication countries with "RPL protocol" topics.

improves the DIO packets overhead, packet retransmissions, delivery rates, and EC but will increase congestion conditions in parents. The trigger feature in this proposed solution monitors the network environment and selects less-congested parents as the preferred parent. They also offer an efficient aggregation approach that minimizes node resource consumption by reducing data packets and combining correlated data. Zier et al. [168] suggested E-RPL as a way to fulfill QoS routing requirements while reducing network access overhead. They restrict the nodes that can wait for DIO to monitor the DIO overhead for ETX. In the DAG structure, nodes remain to receive DIOs from their neighbors before publishing their DIS requests. Otherwise, they will send DIS to receive DIO packets. The sink or gateway will first produce its DIO and then release DIS or DAO. The authors provide a multiconstrained objective function for E-RPL that considers energy and delay metrics with random weights to calculate the rank. Their commitment decreases EC as well as the time it takes from start to finish. However, it lengthens the time it takes for the network to come together.

A new objective function based on fuzzy routines by Fabian et al. [169] provided for the environment dynamic adjustment using EC and ETX metrics. When the node battery level exceeds the defined threshold, ETX is used to calculate the node rank. Suppose that the two-residual energy and cumulative ETX will be used to determine the node rank more minor than the threshold. Third, as the battery runs out, this node is turned off. While the proposed goal feature improves PDR and throughput, it does so at the expense of increased EC. An LB system is used by Ghaleb et al. [170] to pick the parent, which increases network stability. Each node in the DAG structure creates a list of its children (CHlist) based on the received data packets' analysis. Preferred parents are also selected based on two ETX and the number of child nodes metrics. Shifting LB details can occur with the trickle timer exceeded. To stop this trend, the authors have designed a fast propagation timer for CHlist and trickle. If a node has parents with the same rank, a standard metric will be used to select the parent. The CHlist will be examined, and the node with the fewest children will be

chosen as the default parent. The Balance Timer was created to remove the need for regular parent adjustments. The authors' proposed solution could delay network convergence and loop due to rapid packet propagation. Also, the packet delivery ratio and EC in the network will increase. Table 5 shows a similar approach. The author represents an opportunistic objective function based on fuzzy logic (OOP-OF) [58]. This new OF uses the number of children, ETX, and HC metrics for the parent node and seeks to ensure QoS for applications that require reliability and low latency. The proposed fuzzy system evaluates the mentioned metrics as input and combines them based on fuzzy rules. Finally, the aggregated dataset is de-fuzzified, and the parent node routing table is reconstructed based on these outputs. The proposed method will improve latency and delivery rate but increase EC.

10. Statistical Analysis of Review

In this paper, the review of research is presented based on intrusion detection methods of ML. The analysis is done using research sources of "Google scholar", "Crossref", "Scopus", and "Web of Science" resources. Based on the web of science search, only nine papers are published with "RPL" and "machine learning" in their titles and abstracts. Based on Scopus's results, 32 papers are published with "RPL" and "machine learning" in topics. For analysis of the papers with "RPL protocol" in the topics, the web of science database results with 344 papers is illustrated in Figures 6–8. The publication's plot was published with the "RPL protocol" in the topics; the maximum number of papers belongs to 2020. Based on Figure 6, the number of papers on this topic has increased. Moreover, regarding Figure 7, most of the papers are research papers, and 17 review papers are published on this topic. Besides, most of the papers (47/344) are submitted with authors from India. Then, South Korea, England, China, France, Italy, the USA, Iran, Spain, and Saudi Arabia are in the other ranks.

11. Discussion

The inherent characteristics of low-power and lossy networks such as resource constraints, unstable infrastructure, frequent link failures, unreliable communications, and topology dynamics [88] predispose them to various attacks and threats and make it even more difficult detection and mitigation these intrusions effects. Encryption or authentication-based security solutions for major routing solutions in network structures such as WSN quickly deplete node resources due to high computational overload and are unsuitable for LLN and RPL [53, 188]. Given the widespread threats to IoT security, much research has been conducted on known RPL attacks to date. A comprehensive RPL-based intrusion detection and countermeasures solution should be able to detect multiple simultaneous or cooperative attacks. Such a mechanism should also detect and mitigate the effects of malicious nodes, isolate malicious nodes from normal, use an appropriate notification mechanism to inform other nodes, identify mobile nodes, and evaluate the impact of network dynamics in the event of an attack [5].

Various papers have examined different types of attacks, including known or unknown attacks on the RPL protocol. Some of these intrusions are studied more than others, such as SH, BH, SF, and DIS attacks. Other types have received less attention. No study designs its IDS based on Worst Parent attacks, or rare IDSs have focused on neighbor attacks, rank, and DAG inconsistency. No IDS designed to detect or counter all types of attacks has been provided so far, so a comprehensive and ideal IDS should be able to detect all attacks and distinguish between similar functions. Therefore, this critical issue should be further studied in the future. The proposed mechanisms for intrusion detection into the RPL-based IoT, the better it can classify different attacks and evaluate the impact and depth of attacks on the network structure, the more accurate and precise algorithm will be.

Numerous studies have used various metrics such as energy overhead, the ability to detect mobile nodes from static, scalability, the ability to reduce the impact of attacks, and increase the detection rate to evaluate their performance. IDSs that, in addition to the authors' claim, also perform well in the real world must prove their comprehensiveness in addressing various metrics to detect unknown attacks to an acceptable level and be resilient to the system in this situation. A review of multiple studies shows that specification-based mechanisms constitute the predominant part of the mechanisms proposed for IDSs, and anomaly and misuse-based detection, as well as hybrids, are next. Few studies on digital signature-based IDS have been conducted to date. Specification-based IDSs are highly prevalent on LLNs because they keep less CPU and memory busy. To understand the behavior of different types of attacks, classify and analyze them, and provide an efficient intrusion detection solution suitable for LLN and IoT applications, we will need to update conventional datasets [172]. The datasets provided for intrusion detection, which result from different simulations or based on data obtained from different testbeds, such as the KDD 1999 dataset, are usually extracted based on application layer intrusions [173]. They are not the result of LLN-specific traffic flows or threats in the RPL. Therefore, a reliable RPL-based IoT dataset is not yet available. Researchers should evaluate the effectiveness and validity of their proposed approach based on independent simulations and experimental results.

IDSs based on ML algorithms for RPL require a particular database based on RPL and LLN events and processes to enable training and evaluation of ML-based data. Using ML betters malicious behaviors detection on RPL. It distinguishes such behaviors from normal processes in the network appropriately. ML-based techniques combining with DL-based solutions can better meet the challenge of resource constraints in the IoT by optimizing the feature selection process and reducing their dimensions [174]. ML-based learning models in IDS design can improve identifying and dealing with unknown and new attacks and improves IDS behavior based on experienced and untested data by providing new models more intelligently and independently of human control. The combination of ML algorithms with big data [175] can help to realize real-time intrusion detection scenarios [176] when new attacks occur, and IDS optimally trained when exposed to such attacks.

12. Challenges and Open Research Issues

This section will look at fundamental challenges that may benefit future researchers and offer suggestions for addressing them. Identifying existing gaps and taking advantage of opportunities leads us to design a robust yet secure, adaptable, and intelligent IDS that is comprehensive, lightweight, cost-effective, and integrates with new technologies in line with the Internet of Things will guide.

Ability to detect and analyze IDS against new and unknown attacks: IDSs are designed to detect a wide range of attacks, such as RPL attacks. Various studies have shown that specification-based and hybrid mechanisms can better detect unknown attacks than others that IDS has not been trained to detect. So far, little research has focused on identifying new and lesser-known attacks on RPLs. The IDS detection capability, interoperability, and scalability of the various solutions against these intrusions need to be evaluated more accurately. Also, there is a lack of research that assesses IDS from identifying and countering all known types of attacks in the RPL and unknown intrusions. A comprehensive IDS has not yet been presented in this regard.

Collaborative IDS design against cooperative attacks: cooperative attacks are among the most complex ones that can significantly jeopardize DAG performance. These attacks can carry out their threat in a distributed manner by controlling multiple border routers and multiple sensor nodes. So far, little research has focused on identifying these types of attacks. No IDS has been developed that can detect these attacks with the participation of various DODAGs.

Node mobility detection and topology dynamics: most designed IDSs consider the network topology to be static. Over time, several nodes enter and exit the DODAG structure, resulting in many instability and losses of the LLN structure. The various attacks introduced for RPL can be performed using mobile nodes, so IDS must track node mobility. When designing IDS, the dynamics of the network topology must be taken into account.

Overhead and cost metrics: due to the limitations mentioned at the level of LLN nodes, the cost should be considered one of the main parameters in IDS design. The practical solutions proposed for IDS do not consider the cost in parallel with the robustness of IDS against various threats. Attacks that threaten the DAG structure complicate the process of detecting such attacks. Intrusion detection algorithms to detect and deal with these threats cause computational overhead, memory, and energy resources depletion of LLN nodes. Therefore, the stability and robustness of IDS against various security threats should be considered at the same time as its lightweight.

Proper placement of IDS on the network: IDS should capture all traffic exchanged between network devices, including sensor nodes, hosts, and user-side equipment, and monitor comprehensively inbound and outbound network traffic and events. Therefore, the proper position of IDS can directly impact the optimal monitoring of network traffic. Proper IDS placement can lead to better performance, such as higher detection rates and less energy and computational overhead to other network nodes.

Utilizing new ML models to detect intelligent attacks: new ML methods such as active learning for optimal IoT-based IDS training can overcome the problem of data scarcity. Due to the topology dynamics, the multiple needs and requirements of the applications, and the heterogeneity of the nodes in the IoT infrastructure, traditional scenario-based methods will not necessarily be effective. Therefore, ML-based methods can lead to efficient and lightweight intrusion detection mechanisms. Simultaneously, the LLN node resource constraints have challenged the widespread ML-based mechanisms for LLN and the RPL protocol. So, ML algorithms used need to be updated to fit LLN structures for IoT intrusion detection. No IDS has been developed to detect complex ML-based attacks that explore network security vulnerabilities.

Supporting various cyber-physical systems technologies and applications: major IoT-based IDS solutions are optimized for 6LoWPAN networks. In contrast, a wide range of cyberphysical systems (CPS) such as smart homes or industrial and enterprise applications use other standards such as Bluetooth and Wi-Fi. CoAP and MQTT are also used in the IoT application layer. IDS must cover a wide range of different IoT standards and technologies and identify and track the various attacks and threats that arise in them.

Ability to analyze real-time security notifications in IDS: by adopting an appropriate and real-time strategy for handling network security notifications such as attack type, attacker characteristics and location, and adverse effects, more proper decisions can be made against them. Many LLN nodes and the large volume of generated notifications, a significant portion of which have a lower priority, make notification processing tedious, complex, and time-consuming. So, in future research, real-time notification processing should be given special attention. Proper IDS placement in the network and data correlation and abstraction techniques can make faster and easier analyzing the notifications.

Support for QoS metrics: many IoT applications are real-time and require minimal latency. Most IoT intrusion detection studies have limited their scenarios to small- or medium-sized networks. While in the real world, IoT can be an infrastructure of massive nodes with multiple resources. Lack of scalability support can jeopardize IDS performance in identifying complex and widespread threats in the IoT, resulting in depletion of node resources, reduced network performance, and user dissatisfaction. Therefore, IDS for IoT applications, while highly robust, must be scalable and support QoS.

Analyzing encrypted traffic by IDS: encryption is one of the most common techniques for securing data on the network. According to the Gartner report, 80% of enterprise web traffic is encrypted by 2019 [177]. Most IDSs designed for the IoT cannot process encrypted traffic, so attacking nodes can use the encryption technique to their advantage to escape detection by IDS. According to Cisco, 70% of web-based malware traffic is encrypted, and 60% of organizations' attempts to decrypt malware web traffic have failed [178]. Using Cisco Encrypted Traffic Analytics (ETA) technology, metadata and encrypted traffic can be analyzed, and

malicious activity can be detected, regardless of the protocol type and featured included in the IP packets. ETA with passive monitoring can detect all kinds of threats without decrypting traffic.

13. Conclusion

IoT security is of great importance due to its increasing pervasiveness and sensitive application areas. On the other hand, the secure routing protocols proposed each have their weaknesses and will not guarantee complete security for IoT. Accordingly, the attacking plan to these networks is prosperous, and another strategy must be considered to identify the vulnerabilities. Intrusion detection systems (IDS) lead us to this goal. The RPL protocol is subject to various types of security attacks. Low-power and lossy networks' inherent features such as dynamic topology, resource constraints, infrastructure instability, unreliable communications, high losses, and low bit rates make them vulnerable to all kinds of attacks. These limitations and problems are not only specific to RPL-based infrastructures. However, they can also be seen in a variety of WSNs or even wired communications. RPL specifies many protection protocols, including general and local repair mechanisms, loop avoidance, and detection strategies. It also encrypts data packets using two security modes. The normal development of such networks is focused on the link layer, transmission layer, and application layer protection. However, it has believed that the intruder could get around the link layer's protection by gaining access to a shared key. An intruder may be a faulty or misconfigured node that degrades network performance by its behavior. This paper offers a concise summary of IoT intrusion research efforts. Relevant intrusion detection systems for IoT or IoT intrusion detection techniques that may be part of an intrusion detection system were analyzed in the literature. These papers were published from 2009 to 2021. A classification was used to categorize these papers based on the following features: authentication approach, IDS displacement, protection risk, identification method, and IDS architectonics. Based on the analysis done, it can be inferred that intrusion detection system architectures for IoT are only in their early stages.

Data Availability

The paper is a review and data is not applicable.

Disclosure

The funding sources were not involved to support the study design, collection, analysis, interpretation of data, writing of the manuscript, or in the decision to submit the manuscript for publication.

Conflicts of Interest

We declare no conflict of interest.

References

- [1] K. Ashton, "That 'internet of things' thing," *RFID Journal*, vol. 22, no. 7, pp. 97–114, 2009.
- [2] A. E. Varjovi and S. Babaie, "Green internet of things (GIoT): vision, applications and research challenges," *Sustainable Computing: Informatics and Systems*, vol. 28, article 100448, 2020.
- [3] T. Kramp, R. Van Kranenburg, and S. Lange, "Introduction to the Internet of Things," in *Enabling Things to Talk*, pp. 1–10, Springer, Berlin, Heidelberg, 2013.
- [4] P. Gokhale, O. Bhat, and S. Bhat, "Introduction to IOT," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 5, no. 1, 2018.
- [5] Z. A. Almusaylim, A. Alhumam, and N. Z. Jhanjhi, "Proposing a secure RPL based internet of things routing protocol: a review," *Ad Hoc Networks*, vol. 101, article 102096, 2020.
- [6] T. Winter, P. Thubert, A. Brandt et al., *RFC 6550: RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*, RFC, 2012.
- [7] Z. Shelby and C. Bormann, *6LoWPAN: The Wireless Embedded Internet*, Wiley, 1 edition, 2009.
- [8] N. Kushalnagar, G. Montenegro, and C. Schumacher, *RFC 4919: IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*, RFC, 2007.
- [9] G. Mulligan, "The 6LoWPAN architecture," in *Proceedings of the 4th workshop on Embedded networked sensors - EmNets '07*, pp. 78–82, New York, NY, USA, June 2007.
- [10] Z. Shelby and C. Bormann, "6LoWPAN: the wireless embedded internet-part 1: why 6LoWPAN?," *EE Times*, vol. 23, 2011.
- [11] I. Tomić and J. A. McCann, "A survey of potential security issues in existing wireless sensor network protocols," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1910–1923, 2017.
- [12] M. Khurana, T. P. Singh, and T. Choudhury, "Effective threat and security modelling approach to devise security rating of diverse IoT devices," *Data Driven Approach Towards Disruptive Technologies: Proceedings of MIDAS*, pp. 583–593, Springer, Singapore, 2021.
- [13] M. Pishdar, Y. Seifi, M. Nasiri, and M. Bag-Mohammadi, "PCC-RPL: an efficient trust-based security extension for RPL," *Information Security Journal: A Global Perspective*, pp. 1–11, 2021.
- [14] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schönwälder, "Mitigation of topological inconsistency attacks in RPL-based low-power lossy networks," *International Journal of Network Management*, vol. 25, no. 5, pp. 320–339, 2015.
- [15] D. S. Ibrahim, A. F. Mahdi, and Q. M. Yas, "Challenges and issues for wireless sensor networks: a survey," *Journal of Global Scientific Research*, vol. 6, no. 1, pp. 1079–1097, 2021.
- [16] S. Rastegari, P. Hingston, and C.-P. Lam, "Evolving statistical rulesets for network intrusion detection," *Applied Soft Computing*, vol. 33, pp. 348–359, 2015.
- [17] V. Hajisalem and S. Babaie, "A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection," *Computer Networks*, vol. 136, pp. 37–50, 2018.
- [18] B. B. Zarpelao, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, "A survey of intrusion detection in Internet of

- Things,” *Journal of Network and Computer Applications*, vol. 84, pp. 25–37, 2017.
- [19] M. Mehic, M. Niemiec, S. Rass et al., “Quantum key distribution: a networking perspective,” *ACM Computing Surveys (CSUR)*, vol. 53, no. 5, pp. 1–41, 2020.
 - [20] S. Raza, L. Wallgren, and T. Voigt, “SVELTE: real-time intrusion detection in the internet of things,” *Ad Hoc Networks*, vol. 11, no. 8, pp. 2661–2674, 2013.
 - [21] P. Pongle and G. Chavan, “Real time intrusion and wormhole attack detection in internet of things,” *International Journal of Computer Applications*, vol. 121, no. 9, pp. 1–9, 2015.
 - [22] J. Granjal and A. Pedroso, “An intrusion detection and prevention framework for internet-integrated CoAP WSN,” *Security and Communication Networks*, vol. 2018, Article ID 1753897, 14 pages, 2018.
 - [23] H. Bostani and M. Sheikhan, “Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach,” *Computer Communications*, vol. 98, pp. 52–71, 2017.
 - [24] A. Sivanathan, “IoT Behavioral Monitoring via Network Traffic Analysis,” 2020, <http://arxiv.org/abs/2001.10632>.
 - [25] L. García, L. Parra, J. M. Jimenez, J. Lloret, and P. Lorenz, “IoT-based smart irrigation systems: an overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture,” *Sensors*, vol. 20, no. 4, article 1042, 2020.
 - [26] E. J. Cho, J. H. Kim, and C. S. Hong, “Attack model and detection scheme for botnet on 6LoWPAN,” in *Asia-Pacific network operations and management symposium*, pp. 515–518, Springer, 2009.
 - [27] C. Liu, J. Yang, R. Chen, Y. Zhang, and J. Zeng, “Research on immunity-based intrusion detection technology for the Internet of Things,” in *2011 Seventh International Conference on Natural Computation*, pp. 212–216, Shanghai, China, July 2011.
 - [28] A. le, J. Loo, K. Chai, and M. Aiash, “A specification-based IDS for detecting attacks on RPL-based network topology,” *Information*, vol. 7, no. 2, p. 25, 2016.
 - [29] S. Misra, P. V. Krishna, H. Agarwal, A. Saxena, and M. S. Obaidat, “A learning automata based solution for preventing distributed denial of service in internet of things,” in *2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing*, pp. 114–122, Dalian, China, October 2011.
 - [30] P. Kasinathan, G. Costamagna, H. Khaleel, C. Pastrone, and M. A. Spirito, “An IDS framework for internet of things empowered by 6LoWPAN,” in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1337–1340, New York, NY, USA, November 2013.
 - [31] L. Wallgren, S. Raza, and T. Voigt, “Routing attacks and countermeasures in the RPL-based internet of things,” *International Journal of Distributed Sensor Networks*, vol. 9, no. 8, 2013.
 - [32] A. Gupta, O. J. Pandey, M. Shukla, A. Dadhich, S. Mathur, and A. Ingle, “Computational intelligence based intrusion detection systems for wireless communication and pervasive computing networks,” in *2013 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–7, Enathi, India, December 2013.
 - [33] P. Kasinathan, C. Pastrone, M. A. Spirito, and M. Vinkovits, “Denial-of-Service detection in 6LoWPAN based Internet of Things,” in *2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 600–607, Lyon, France, October 2013.
 - [34] J. P. Amaral, L. M. Oliveira, J. J. Rodrigues, G. Han, and L. Shu, “Policy and network-based intrusion detection system for IPv6-enabled wireless sensor networks,” in *2014 IEEE International Conference on Communications (ICC)*, pp. 1796–1801, Sydney, NSW, Australia, June 2014.
 - [35] D. Oh, D. Kim, and W. W. Ro, “A malicious pattern detection engine for embedded security systems in the Internet of Things,” *Sensors*, vol. 14, no. 12, pp. 24188–24211, 2014.
 - [36] T.-H. Lee, C.-H. Wen, L.-H. Chang, H.-S. Chiang, and M.-C. Hsieh, “A lightweight intrusion detection scheme based on energy consumption analysis in 6LoWPAN,” in *Advanced Technologies, Embedded and Multimedia for Human-centric Computing. Lecture Notes in Electrical Engineering*, vol. 260, Y. M. Huang, H. C. Chao, D. J. Deng, and J. Park, Eds., pp. 1205–1213, Springer, Dordrecht, 2014.
 - [37] J. Krimmling and S. Peter, “Integration and evaluation of intrusion detection for CoAP in smart city applications,” in *2014 IEEE Conference on Communications and Network Security*, pp. 73–78, San Francisco, CA, USA, October 2014.
 - [38] C. Cervantes, D. Poplade, M. Nogueira, and A. Santos, “Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things,” in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 606–611, Ottawa, ON, Canada, May 2015.
 - [39] D. H. Summerville, K. M. Zach, and Y. Chen, “Ultra-lightweight deep packet anomaly detection for Internet of Things devices,” in *2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, pp. 1–8, Nanjing, China, December 2015.
 - [40] N. K. Thanigaivelan, E. Nigussie, R. K. Kanth, S. Virtanen, and J. Isoaho, “Distributed internal anomaly detection system for Internet-of-Things,” in *2016 13th IEEE annual consumer communications & networking conference (CCNC)*, pp. 319–320, Las Vegas, NV, USA, January 2016.
 - [41] J. Sengupta, S. Ruj, and S. D. Bit, “A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT,” *Journal of Network and Computer Applications*, vol. 149, p. 102481, 2020.
 - [42] D. Airehrour, J. Gutierrez, and S. K. Ray, “A trust-aware rpl routing protocol to detect blackhole and selective forwarding attacks,” *Journal of Telecommunications and the Digital Economy*, vol. 5, no. 1, pp. 50–69, 2017.
 - [43] O. Gnawali, “RFC 6719: The Minimum Rank with Hysteresis Objective Function,” *Internet Engineering Task Force (IETF) Request For Comments*, 2012.
 - [44] P. Thubert, “RFC 6552: Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL),” *RFC*, pp. 1–14, 2012.
 - [45] A. H. Chowdhury, M. Ikram, H.-S. Cha et al., “Route-over vs mesh-under routing in 6LoWPAN,” in *Proceedings of the 2009 international conference on wireless communications and mobile computing: Connecting the world wirelessly*, pp. 1208–1212, Leipzig, Germany, 2009.
 - [46] A. Rehman, M. M. Khan, M. A. Lodhi, and F. B. Hussain, “Rank attack using objective function in RPL for low power and lossy networks,” in *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, pp. 1–5, Sharjah, United Arab Emirates, 2016.

- [47] D. Airehrour, J. A. Gutierrez, and S. K. Ray, "SecTrust-RPL: a secure trust-aware RPL routing protocol for Internet of Things," *Future Generation Computer Systems*, vol. 93, pp. 860–876, 2019.
- [48] B. Farzaneh, M. Koosha, E. Boochanpour, and E. Alizadeh, "A new method for intrusion detection on RPL routing protocol using fuzzy logic," in *2020 6th International Conference on Web Research (ICWR)*, pp. 245–250, Tehran, Iran, 2020.
- [49] A. Verma and V. Ranga, "ELNIDS: Ensemble learning based network intrusion detection system for RPL based Internet of Things," in *2019 4th International conference on Internet of Things: Smart innovation and usages (IoT-SIU)*, pp. 1–6, Ghaziabad, India, 2019.
- [50] A. Verma and V. Ranga, "Evaluation of network intrusion detection systems for RPL based 6LoWPAN networks in IoT," *Wireless Personal Communications*, vol. 108, no. 3, pp. 1571–1594, 2019.
- [51] E. G. Ribera, B. M. Alvarez, C. Samuel, P. P. Ioulianou, and V. G. Vassilakis, "Heartbeat-based detection of blackhole and greyhole attacks in RPL networks," in *2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, pp. 1–6, Porto, Portugal, 2020.
- [52] C. Samuel, B. M. Alvarez, E. G. Ribera, P. P. Ioulianou, and V. G. Vassilakis, "Performance evaluation of a wormhole detection method using round-trip times and hop counts in RPL-based 6LoWPAN networks," in *2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, pp. 1–6, Porto, Portugal, 2020.
- [53] M. A. Boudouaia, A. Ali-Pacha, A. Abouaissa, and P. Lorenz, "Security against rank attack in RPL protocol," *IEEE Network*, vol. 34, no. 4, pp. 133–139, 2020.
- [54] P. Ekparinya, V. Gramoli, and G. Jourjon, "Impact of man-in-the-middle attacks on ethereum," in *2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS)*, pp. 11–20, Salvador, Brazil, 2018.
- [55] E. Ciklabakkal, A. Donmez, M. Erdemir, E. Suren, M. K. Yilmaz, and P. Angin, "ARTEMIS: an intrusion detection system for MQTT attacks in Internet of Things," in *2019 38th Symposium on Reliable Distributed Systems (SRDS)*, pp. 369–3692, Lyon, France, 2019.
- [56] P. Bhale, S. Dey, S. Biswas, and S. Nandi, "Energy efficient approach to detect sinkhole attack using roving IDS in 6LoWPAN network," in *Innovations for Community Services. I4CS 2020*, pp. 187–207, Springer, 2020.
- [57] M. Zaminkar and R. Fotohi, "SoS-RPL: securing internet of things against sinkhole attack using RPL protocol-based node rating and ranking mechanism," *Wireless Personal Communications*, vol. 114, no. 2, pp. 1287–1312, 2020.
- [58] A. Verma and V. Ranga, "Mitigation of DIS flooding attacks in RPL-based 6LoWPAN networks," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, article e3802, 2020.
- [59] A. Verma and V. Ranga, "The impact of copycat attack on RPL based 6LoWPAN networks in Internet of Things," *Computing*, vol. 103, no. 7, pp. 1479–1500, 2021.
- [60] G. Guo, "A Lightweight Countermeasure to DIS Attack in RPL Routing Protocol," in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 0753–0758, NV, USA, 2021.
- [61] M. Yadollahzadeh Tabari and Z. Mataji, "Detecting sinkhole attack in RPL-based internet of things routing protocol," *Journal of AI and Data Mining*, vol. 9, no. 1, pp. 73–85, 2021.
- [62] A. A. Anitha and L. Arockiam, "VeNADet: version number attack detection for RPL based Internet of Things," *Solid State Technology*, vol. 64, no. 2, pp. 2225–2237, 2021.
- [63] M. Sheikhan and H. Bostani, "A security mechanism for detecting intrusions in internet of things using selected features based on mi-bgsa," *International Journal of Information & Communication Technology Research*, vol. 9, no. 2, pp. 53–62, 2017.
- [64] M. N. Napiyah, M. Y. I. Bin Idris, R. Ramli, and I. Ahmedy, "Compression header analyzer intrusion detection system (CHA - IDS) for 6LoWPAN communication protocol," *IEEE Access*, vol. 6, pp. 16623–16638, 2018.
- [65] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schönwälder, "Using the RPL protocol for supporting passive monitoring in the Internet of Things," in *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, pp. 366–374, Istanbul, Turkey, 2016.
- [66] A. Mayzaud, R. Badonnel, and I. Chrisment, "A taxonomy of attacks in RPL-based Internet of Things," *International Journal of Network Security*, vol. 18, no. 3, pp. 459–473, 2016.
- [67] N. Sousa, J. V. Sobral, J. J. Rodrigues, R. A. Rabêlo, and P. Solic, "ERAOF: a new RPL protocol objective function for Internet of Things applications," in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech)*, pp. 1–5, Split, Croatia, 2017.
- [68] N. Farah, M. Avishek, F. Muhammad, A. Rahman, M. Rafni, and D. Md, "Application of machine learning approaches in intrusion detection system: a survey," *International Journal of Advanced Research in Artificial Intelligence*, vol. 4, no. 3, pp. 9–18, 2015.
- [69] N. Lu, Y. Sun, H. Liu, and S. Li, "Intrusion detection system based on evolving rules for wireless sensor networks," *Journal of Sensors*, vol. 2018, Article ID 5948146, 8 pages, 2018.
- [70] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.
- [71] F. Alam, R. Mehmood, I. Katib, and A. Albeshri, "Analysis of eight data mining algorithms for smarter Internet of Things (IoT)," *Procedia Computer Science*, vol. 98, pp. 437–442, 2016.
- [72] H. Yin and K. Gai, "An empirical study on preprocessing high-dimensional class-imbalanced data for classification," in *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems*, pp. 1314–1319, New York, NY, USA, 2015.
- [73] F.-L. Fan, J. Xiong, M. Li, and G. Wang, "On interpretability of artificial neural networks: a survey," *IEEE Transactions on Radiation and Plasma Medical Sciences*, 2021.
- [74] M. Q. H. Abadi, S. Rahmati, A. Sharifi, and M. Ahmadi, "HSSAGA: designation and scheduling of nurses for taking care of COVID-19 patients using novel method of hybrid salp swarm algorithm and genetic algorithm," *Applied Soft Computing*, vol. 108, article 107449, 2021.
- [75] A. Sharifi, M. Ahmadi, M. A. Mehni, S. J. Ghouschi, and Y. Pourasad, "Experimental and numerical diagnosis of

- fatigue foot using convolutional neural network,” *Computer Methods in Biomechanics and Biomedical Engineering*, pp. 1–13, 2021.
- [76] A. K. Bediya and R. Kumar, “Real time DDoS intrusion detection and monitoring framework in 6LoWPAN for Internet of Things,” in *2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON)*, pp. 824–828, Greater Noida, India, 2020.
- [77] A. Saleem, M. K. Afzal, M. Ateeq, S. W. Kim, and Y. B. Zikria, “Intelligent learning automata-based objective function in RPL for IoT,” *Sustainable Cities and Society*, vol. 59, article 102234, 2020.
- [78] A. Khraisat and A. Alazab, “A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges,” *Cybersecurity*, vol. 4, no. 1, pp. 1–27, 2021.
- [79] O. Balci, “Verification validation and accreditation of simulation models,” in *Proceedings of the 29th conference on Winter simulation*, pp. 135–141, Atlanta, GA, USA, 1997.
- [80] P. Sanmartin, A. Rojas, L. Fernandez, K. Avila, D. Jabba, and S. Valle, “Sigma routing metric for RPL protocol,” *Sensors*, vol. 18, no. 4, p. 1277, 2018.
- [81] N. Srinidhi, S. M. Dilip Kumar, and K. Venugopal, “Network optimizations in the Internet of Things: a review,” *Engineering Science and Technology, an International Journal*, vol. 22, no. 1, pp. 1–21, 2019.
- [82] O. Gaddour, A. Koubâa, N. Baccour, and M. Abid, “OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol,” in *2014 12th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pp. 365–372, Hammamet, Tunisia, 2014.
- [83] A. J. Witwit and A. K. Idrees, “A comprehensive review for RPL routing protocol in low power and lossy networks,” in *Communications in Computer and Information*, pp. 50–66, Springer, 2018.
- [84] K. S. Bhandari and G. Cho, “An energy efficient routing approach for cloud-assisted green industrial IoT networks,” *Sustainability*, vol. 12, no. 18, p. 7358, 2020.
- [85] O. Gaddour and A. Koubâa, “RPL in a nutshell: a survey,” *Computer Networks*, vol. 56, no. 14, pp. 3163–3178, 2012.
- [86] M. Zhao, A. Kumar, P. H. Joo Chong, and R. Lu, “A comprehensive study of RPL and P2P-RPL routing protocols: implementation, challenges and opportunities,” *Peer-to-Peer Networking and Applications*, vol. 10, no. 5, pp. 1232–1256, 2017.
- [87] X. Niu, “Optimizing DODAG build with RPL protocol,” *Mathematical Problems in Engineering*, vol. 2021, Article ID 5579564, 8 pages, 2021.
- [88] D. Pancaroglu and S. Sen, “Load balancing for RPL-based Internet of Things: a review,” *Ad Hoc Networks*, vol. 116, article 102491, 2021.
- [89] M. Halder, M. Sheikh, M. Rahman, and M. Rahman, “Performance analysis of CoAP, 6LoWPAN and RPL routing protocols of IoT using COOJA simulator,” *International Journal of Scientific and Engineering Research*, vol. 9, pp. 1671–1677, 2018.
- [90] P. Perazzo, C. Vallati, G. Anastasi, and G. Dini, “DIO suppression attack against routing in the Internet of Things,” *IEEE Communications Letters*, vol. 21, no. 11, pp. 2524–2527, 2017.
- [91] B. Ghaleb, A. al-Dubai, E. Ekonomou, M. Qasem, I. Romdhani, and L. Mackenzie, “Addressing the DAO insider attack in RPL’s Internet of Things networks,” *IEEE Communications Letters*, vol. 23, no. 1, pp. 68–71, 2019.
- [92] I. U. Onwuegbuzie, S. A. Razak, and I. F. Isnin, “Control messages overhead impact on destination oriented directed acyclic graph—a wireless sensor networks objective functions performance comparison,” *Journal of Computational and Theoretical Nanoscience*, vol. 17, no. 2, pp. 1227–1235, 2020.
- [93] E. Aljarrah, M. B. Yassein, and S. Aljawarneh, “Routing protocol of low-power and lossy network: survey and open issues,” in *2016 International Conference on Engineering & MIS (ICEMIS)*, pp. 1–6, Agadir, Morocco, 2016.
- [94] S. S. Solapure and H. H. Kenchannavar, “Design and analysis of RPL objective functions using variant routing metrics for IoT applications,” *Wireless Networks*, vol. 26, no. 6, pp. 4637–4656, 2020.
- [95] H.-S. Kim, J. Ko, D. E. Culler, and J. Paek, “Challenging the IPv6 routing protocol for low-power and lossy networks (RPL): a survey,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2502–2525, 2017.
- [96] R. Stephen and L. Arockiam, “E2V: techniques for detecting and mitigating rank inconsistency attack (RInA) in RPL based Internet of Things,” *Journal of Physics: Conference Series*, vol. 1142, p. 012009, 2018.
- [97] M. Wang, Z. Zou, C. Pu, P. Wang, and Y. Yin, “A time synchronization scheme based on RPL for 6LoWPAN networks,” in *2020 Chinese Automation Congress (CAC)*, pp. 5969–5974, Agadir, Morocco, 2020.
- [98] R. K. Yadav and N. Awasthi, “A survey on enhanced RPL: addressing the mobility in RPL,” in *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pp. 1189–1195, Palladam, India, 2020.
- [99] H. Lamaazi and N. Benamar, “A novel approach for RPL assessment based on the objective function and trickle optimizations,” *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 4605095, 9 pages, 2019.
- [100] P. R. Satav, P. M. Jawandhiya, and V. M. Thakare, “Secure route selection mechanism in the presence of black hole attack with AOMDV routing algorithm,” in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCCUBEA)*, pp. 1–6, Pune, India, 2018.
- [101] W. Choukri, H. Lamaazi, and N. Benamar, “RPL rank attack detection using Deep Learning,” in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*, pp. 1–6, Sakheer, Bahrain, 2020.
- [102] S. IBRAHIMY, H. LAMAAZI, and N. BENAMAR, “RPL assessment using the rank attack in static and mobile environments,” in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)*, pp. 1–6, Sakheer, Bahrain, 2020.
- [103] H. Wang, G. Wang, Y. Li, D. Zhang, and L. Lin, “Transferable, controllable, and inconspicuous adversarial attacks on person re-identification with deep mis-ranking,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 342–351, Seattle, WA, USA, 2020.
- [104] P. O. Kamgueu, E. Nataf, and T. D. Ndie, “Survey on RPL enhancements: a focus on topology, security and mobility,” *Computer Communications*, vol. 120, pp. 10–21, 2018.

- [105] R. Sahay, G. Geethakumari, and K. Modugu, "Attack graph-based vulnerability assessment of rank property in RPL-6LOWPAN in IoT," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, pp. 308–313, Singapore, 2018.
- [106] G. Levitin, L. Xing, and Y. Xiang, "Co-residence data theft attacks on N-Version programming-based cloud services with task cancelation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–10, 2020.
- [107] B. Al-Musawi, P. Branch, M. F. Hassan, and S. R. Pokhrel, "Identifying OSPF LSA falsification attacks through non-linear analysis," *Computer Networks*, vol. 167, p. 107031, 2020.
- [108] A. G. Finogeev and A. A. Finogeev, "Information attacks and security in wireless sensor networks of industrial SCADA systems," *Journal of Industrial Information Integration*, vol. 5, pp. 6–16, 2017.
- [109] A. Raoof, A. Matrawy, and C.-H. Lung, "Secure routing in IoT: evaluation of RPL's secure mode under attacks," in *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Waikoloa, HI, USA, 2019.
- [110] H. Yaohui, C. Qinghua, H. Bing, and W. Yang, "Vulnerability of complex networks under neighbor nodes attack strategies," in *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, pp. 1193–1197, Chengdu, China, 2018.
- [111] C. Pu, "Spam DIS attack against routing protocol in the Internet of Things," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, pp. 73–77, Honolulu, HI, USA, 2019.
- [112] R. Sahay, G. Geethakumari, and B. Mitra, "A feedforward neural network based model to predict sub-optimal path attack in IoT-LLNs," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, pp. 400–409, Melbourne, VIC, Australia, 2020.
- [113] C. Pu, "Mitigating DAO inconsistency attack in RPL-based low power and lossy networks," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 570–574, Las Vegas, NV, USA, 2018.
- [114] A. Dvir and L. Buttyan, "VeRA-version number and rank authentication in RPL," in *2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, pp. 709–714, Spain, 2011.
- [115] A. D. Seth, S. Biswas, and A. K. Dhar, "Detection and verification of decreased rank attack using round-trip times in RPL-based 6LoWPAN networks," in *2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pp. 1–6, New Delhi, India, 2020.
- [116] M. A. Hamid, M. Rashid, and C. S. Hong, "Routing security in sensor network: hello flood attack and defense," *IEEE ICNEWS*, vol. 2, pp. 2–4, 2006.
- [117] T. Aditya Sai Srinivas and S. Manivannan, "Prevention of hello flood attack in IoT using combination of deep learning with improved rider optimization algorithm," *Computer Communications*, vol. 163, pp. 162–175, 2020.
- [118] A.-u. Rehman, S. U. Rehman, and H. Raheem, "Sinkhole attacks in wireless sensor networks: a survey," *Wireless Personal Communications*, vol. 106, no. 4, pp. 2291–2313, 2019.
- [119] M. M. Iqbal, A. Ahmed, and U. Khadam, "Sinkhole attack in multi-sink paradigm: detection and performance evaluation in RPL based IoT," in *2020 International Conference on Computing and Information Technology (ICCIT-1441)*, pp. 1–5, Tabuk, Saudi Arabia, 2020.
- [120] S. Deshmukh-Bhosale and S. S. Sonavane, "A real-time intrusion detection system for wormhole attack in the RPL based Internet of Things," *Procedia Manufacturing*, vol. 32, pp. 840–847, 2019.
- [121] P. Perazzo, C. Vallati, D. Varano, G. Anastasi, and G. Dini, "Implementation of a wormhole attack against a rpl network: challenges and effects," in *2018 14th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, pp. 95–102, Isola 2000, France, 2018.
- [122] R. M. Reddy and V. Neerugatti, "Anomaly based technique for detection and prevention of black hole attacks in RPL based networks," in *International Conference on Universal Computing, Communications and Data Engineering (CCODE-2018)*, India, 2018 <https://ssrn.com/abstract=3295459>.
- [123] G. Simoglou, G. Violettas, S. Petridou, and L. Mamatas, "Intrusion detection systems for RPL security: a comparative analysis," *Computers & Security*, vol. 104, p. 102219, 2021.
- [124] S. Mangelkar, S. N. Dhage, and A. V. Nimkar, "A comparative study on RPL attacks and security solutions," in *2017 International Conference on Intelligent Computing and Control (I2C2)*, pp. 1–6, Coimbatore, India, 2017.
- [125] A. M. Pasikhani, J. A. Clark, P. Gope, and A. Alshahrani, "Intrusion detection systems in RPL-based 6LoWPAN: a systematic literature review," *IEEE Sensors Journal*, vol. 21, no. 11, pp. 12940–12968, 2021.
- [126] R. Mehta and M. Parmar, "Trust based mechanism for securing iot routing protocol rpl against wormhole & grayhole attacks," in *2018 3rd International Conference for Convergence in Technology (I2CT)*, pp. 1–6, Pune, India, 2018.
- [127] M. A. Kareem and S. Tayeb, "ML-based NIDS to secure RPL from routing attacks," in *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 1000–1006, NV, USA, 2021.
- [128] S. M. H. Mirshahjafari and B. S. Ghahfarokhi, "Sinkhole+ CloneID: a hybrid attack on RPL performance and detection method," *Information Security Journal: A Global Perspective*, vol. 28, no. 4-5, pp. 107–119, 2019.
- [129] M. Ahmadi and M. Qaisari Hasan Abadi, "A review of using object-orientation properties of C++ for designing expert system in strategic planning," *Computer Science Review*, vol. 37, p. 100282, 2020.
- [130] J. Waring, C. Lindvall, and R. Umeton, "Automated machine learning: review of the state-of-the-art and opportunities for healthcare," *Artificial Intelligence in Medicine*, vol. 104, p. 101822, 2020.
- [131] S. Dorosti, S. Jafarzadeh Ghouschi, E. Sobhrakhshankhah, M. Ahmadi, and A. Sharifi, "Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location," *Soft Computing*, vol. 24, no. 13, pp. 9943–9964, 2020.
- [132] K. P. Sinaga and M.-S. Yang, "Unsupervised K-means clustering algorithm," *IEEE access*, vol. 8, pp. 80716–80727, 2020.
- [133] H. Xie, M. E. Hussein, A. Galstyan, and W. Abd-Elmageed, "Muscle: strengthening semi-supervised learning via concurrent unsupervised learning using mutual information maximization," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2586–2595, Waikoloa, HI, USA, 2021.

- [134] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.
- [135] Y. Tian, R. Yuan, D. Xue et al., "Determining multi-component phase diagrams with desired characteristics using active learning," *Advanced Science*, vol. 8, no. 1, p. 2003165, 2021.
- [136] J. Wu, V. S. Sheng, J. Zhang et al., "Multi-label active learning algorithms for image classification," *ACM Computing Surveys (CSUR)*, vol. 53, no. 2, pp. 1–35, 2020.
- [137] G. An, M. Akiba, K. Omodaka, T. Nakazawa, and H. Yokota, "Hierarchical deep learning models using transfer learning for disease detection and classification based on small number of medical images," *Scientific Reports*, vol. 11, no. 1, pp. 4250–4259, 2021.
- [138] O. A. Wahab, A. Mourad, H. Otrouk, and T. Taleb, "Federated machine learning: survey, multi-level classification, desirable criteria and future directions in communication and networking systems," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1342–1397, 2021.
- [139] A. I. Nasiru, M. D. Abdulrahman, and A. Adams, "Comparative analysis of selected data mining algorithms for intrusion detection system," *Technoscience Journal for Community Development in Africa*, vol. 1, no. 1, pp. 81–89, 2020.
- [140] A. N. Cahyo, E. Winarko, and A. Musdholifah, "Survey of data mining techniques for intrusion detection systems," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*, pp. 1–8, Indonesia, 2020.
- [141] P. Kannimuthu and J. Thangamuthu, "Decision tree trust (DTTrust)-based authentication mechanism to secure RPL routing protocol on internet of battlefield thing (IoBT)," *International Journal of Business Data Communications and Networking (IJBDCN)*, vol. 17, no. 1, pp. 1–23, 2021.
- [142] S. S. Ambarkar and N. M. Shekoker, "Improving security of IoT networks using machine learning-based intrusion detection system," in *Advanced Computing Technologies and Applications*, pp. 199–210, Springer, 2020.
- [143] I. F. Kilincer, F. Ertam, and A. Sengur, "Machine learning methods for cyber security intrusion detection: datasets and comparative study," *Computer Networks*, vol. 188, p. 107840, 2021.
- [144] V. Bruno, M. D'Orazio, C. Ticconi et al., "Machine learning (ML) based-method applied in recurrent pregnancy loss (RPL) patients diagnostic work-up: a potential innovation in common clinical practice," *Scientific Reports*, vol. 10, no. 1, 2020.
- [145] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset," *IEEE access*, vol. 9, pp. 22351–22370, 2021.
- [146] A. Gül and E. Adalı, "A feature selection algorithm for IDS," in *2017 International Conference on Computer Science and Engineering (UBMK)*, pp. 816–820, Tukey, 2017.
- [147] A. Das and R. B. Nayak, "A divide and conquer feature reduction and feature selection algorithm in KDD intrusion detection dataset," in *IET Chennai 3rd International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2012)*, India, 2012.
- [148] G. Meena and R. R. Choudhary, "A review paper on IDS classification using KDD 99 and NSL KDD dataset in WEKA," in *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, pp. 553–558, India, 2017.
- [149] K. D. D. Cup, 1999, <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>.
- [150] S. Dongre and M. Chawla, "Analysis of feature selection techniques for denial of service (DoS) attacks," in *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*, pp. 1–4, India, 2018.
- [151] S. Umbarkar and S. Shukla, "Analysis of heuristic based feature reduction method in intrusion detection system," in *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 717–720, India, 2018.
- [152] S. M. Kasongo and Y. Sun, "A deep learning method with filter based feature engineering for wireless intrusion detection system," *IEEE access*, vol. 7, pp. 38597–38607, 2019.
- [153] V. Neerugatti and A. R. Mohan Reddy, "Machine learning based technique for detection of rank attack in RPL based internet of things networks," *International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN*, vol. 8, pp. 2278–3075, 2019, <https://ssrn.com/abstract=3435598>.
- [154] S. Shin, K. Kim, and T. Kwon, "Detection of malicious packet dropping attacks in RPL-based internet of things," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 31, no. 2, pp. 133–141, 2019.
- [155] K. Bhandari, A. Hosen, and G. Cho, "CoAR: congestion-aware routing protocol for low power and lossy networks for IoT applications," *Sensors*, vol. 18, no. 11, p. 3838, 2018.
- [156] F. Y. Yavuz, D. Ünal, and E. Gül, "Deep learning for detection of routing attacks in the internet of things," *International Journal of Computational Intelligence Systems*, vol. 12, no. 1, pp. 39–58, 2018.
- [157] V. Neerugatti and R. M. Reddy, "Acknowledgement based technique for detection of the wormhole attack in RPL based Internet of Things networks," *Asian Journal of Computer Science and Technology*, vol. 8, Supplement 3, pp. 100–104, 2019.
- [158] B. Ghaleb, A. Y. al-Dubai, E. Ekonomou et al., "A survey of limitations and enhancements of the IPv6 routing protocol for low-power and lossy networks: a focus on core operations," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1607–1635, 2019.
- [159] E. Ancillotti, C. Vallati, R. Bruno, and E. Mingozzi, "A reinforcement learning-based link quality estimation strategy for RPL and its impact on topology management," *Computer Communications*, vol. 112, pp. 1–13, 2017.
- [160] S. Taghizadeh, H. Bobarshad, and H. Elbiaze, "CLRPL: context-aware and load balancing RPL for IoT networks under heavy and highly dynamic load," *IEEE access*, vol. 6, pp. 23277–23291, 2018.
- [161] J. Hou, R. Jadhav, and Z. Luo, "Optimization of parent-node selection in RPL-based networks," *Internet Engineering Task Force (IETF) draft*, pp. 1–11, 2017, <https://tools.ietf.org/html/draft-hou-roll-rpl-parent-selection-00>.
- [162] J. Papathanasiou and N. Ploskas, "Multiple criteria decision aid," in *Methods, Examples and Python Implementations*, Vol. 136, Springer, 2018.
- [163] J. Nassar, N. Gouvy, and N. Mitton, "Towards multi-instances QoS efficient RPL for smart grids," in *Proceedings of the 14th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, pp. 85–92, ACM: New York, 2017.
- [164] H. Lamaazi, A. El Ahmadi, N. Benamar, and A. J. Jara, "OF-ECF: a new optimization of the objective function for parent

- selection in RPL,” in *2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 27–32, Spain, 2019.
- [165] Y. Cao and M. Wu, “A novel RPL algorithm based on chaotic genetic algorithm,” *Sensors*, vol. 18, no. 11, p. 3647, 2018.
- [166] H. Lamaazi and N. Benamar, “OF-EC: a novel energy consumption aware objective function for RPL based on fuzzy logic,” *Journal of Network and Computer Applications*, vol. 117, pp. 42–58, 2018.
- [167] A. Bahramlou and R. Javidan, “Adaptive timing model for improving routing and data aggregation in Internet of things networks using RPL,” *IET Networks*, vol. 7, no. 5, pp. 306–312, 2018.
- [168] A. Zier, A. Abouaissa, and P. Lorenz, “E-RPL: a routing protocol for IoT networks,” in *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, UAE, 2018.
- [169] P. Fabian, A. Rachedi, C. Gueguen, and S. Lohier, “Fuzzy-based objective function for routing protocol in the internet of things,” in *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, UAE, 2018.
- [170] B. Ghaleb, A. Al-Dubai, E. Ekonomou, W. Gharib, L. Mackenzi, and M. B. Khala, “A new load-balancing aware objective function for RPL’s IoT networks,” in *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 909–914, UK, 2018.
- [171] I. Kechiche, I. Bousnina, and A. Samet, “A novel opportunistic fuzzy logic based objective function for the routing protocol for low-power and lossy networks,” in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 698–703, Morocco, 2019.
- [172] A. Taivalsaari and T. Mikkonen, “A roadmap to the programmable world: software challenges in the IoT era,” *IEEE Software*, vol. 34, no. 1, pp. 72–80, 2017.
- [173] L. Sanchez, L. Muñoz, J. A. Galache et al., “SmartSantander: IoT experimentation over a smart city testbed,” *Computer Networks*, vol. 61, pp. 217–238, 2014.
- [174] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, “Network intrusion detection for IoT security based on learning techniques,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019.
- [175] E. Ahmed, I. Yaqoob, I. A. T. Hashem et al., “The role of big data analytics in Internet of Things,” *Computer Networks*, vol. 129, pp. 459–471, 2017.
- [176] H. Hromic, D. Le Phuoc, M. Serrano et al., “Real time analysis of sensor data for the internet of things by means of clustering and event processing,” in *2015 IEEE International conference on communications (ICC)*, pp. 685–691, UK, 2015.
- [177] Y. Borchani, “Advanced malicious beaconing detection through AI,” *Network Security*, vol. 2020, no. 3, pp. 8–14, 2020.
- [178] “Cisco Encrypted Traffic Analytics White Paper,” *Cisco Public*, vol. 2019, 2021, <https://www.cisco.com/c/en/us/solutions/collateral/enterprise-networks/enterprise-network-security/nb-09-encrytd-traf-anlytcs-wp-cte-en.html>.
- [179] L. Nie, Z. Ning, M. S. Obaidat et al., “A reinforcement learning-based network traffic prediction mechanism in intelligent internet of things,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 2169–2180, 2021.
- [180] L. Nie, X. Wang, S. Wang et al., “Network traffic prediction in industrial internet of things backbone networks: a multi-task learning mechanism,” *IEEE Transactions on Industrial Informatics*, vol. 8, 2021.
- [181] M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, “Detection of brain lesion location in MRI images using convolutional neural network and robust PCA,” *International Journal of Neuroscience*, vol. 4, pp. 1–2, 2021.
- [182] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, “QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network,” *BioMed Research International*, vol. 18, 2021.
- [183] S. Xu, W. Ding, and Z. Liu, “Automatic dialogic instruction detection for k-12 online one-on-one classes,” in *International conference on artificial intelligence in education*, pp. 340–345, Springer, Champions, 2020.
- [184] J. Wu, B. Cui, C. Chen, and X. Long, “A high efficiency and accuracy method for x86 undocumented instruction detection and classification,” in *In International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 295–303, Springer, Champions, 2021.
- [185] A. Varmaghani, A. Matin Nazar, M. Ahmadi, A. Sharifi, S. Jafarzadeh Ghouschi, and Y. Pourasad, “DMTC: optimize energy consumption in dynamic wireless sensor network based on fog computing and fuzzy multiple attribute decision-making,” *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 9953416, 14 pages, 2021.