www.czasopisma.pan.pl PAN www.journals.pan.pl

jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2023, VOL. 69, NO. 2, PP. 309-318
Manuscript received October 17, 2022; revised April, 2023. DOI: 10.24425/ijet.2023.144366

Design and Implementation of Intrusion Detection Systems using RPL and AOVD Protocols-based Wireless Sensor Networks

Joseph Kipongo, Theo G. Swart, and Ebenezer Esenogho

***Abstract*— Wireless Sensor Network (WSN) technology has grown in importance in recent years. All WSN implementations need secure data transmission between sensor nodes and base stations. Sensor node attacks introduce new threats to the WSN. As a result, an appropriate Intrusion Detection System (IDS) is required in WSN for defending against security attacks and detecting attacks on sensor nodes. In this study, we use the Routing Protocol for Low Power and Lossy Networks (RPL) for addressing security services in WSN by identifying IDS with a network size of more or less 20 nodes and introducing 10% malicious nodes. The method described above is used on Cooja in the VMware virtual machine Workstation with the InstantContiki2.7 operating system. To track the movement of nodes, find network attacks, and spot dropped packets during IDS in WSN, an algorithm is implemented in the Network Simulator (NS2) using the Ad-hoc On-Demand Distance Vector (AODV) protocol in the Linux operating system.**

***Keywords*—Intrusion Detection Systems; wireless sensor networks; Cooja simulator; sensor nodes; NS2**

I. Introduction

A WIRELESS sensor network (WSN) is a network of basic sensing devices that can detect changes in parameters and communicate with other devices over a particular geographic region for applications such as target tracking, surveillance, environmental monitoring, and so on. Since sensor nodes are severely limited in processing power, storage capacity, and energy, routing and data aggregation in WSN are extremely difficult owing to intrinsic features. The architecture of WSN is shown in Figure 1.

Wireless technology advancements have led to the creation of low-cost, low-power sensor nodes [1], [2]. The WSN consists of theses nodes. These nodes are now part of the internet of things (IoT). The IoT consists of billions of such nodes interconnected. As a result, since IPv4 address space may be insufficient, these nodes use IPv6 over Low-power Wireless Personal Area Network (6LoWPAN) [3], [4]. Conventional routing protocols cannot handle these networks. For such networks, the Routing Protocol for Low Power and Lossy Networks (RPL), a new routing protocol, has been standardized.

RPL is intended to be a proactive, distance-vector routing technology. It begins by detecting routes as soon as the RPL network is activated.

A simple and effective routing protocol called Ad-hoc On-Demand Distance Vector (AODV) was created, particularly for use in multi-hop wireless ad-hoc networks of mobile nodes [5]. When needed, routes are constructed. AODV uses traditional routing rules, one entry per destination, and sequence numbers for checking for and preventing routing loops. AODV can achieve a fast and efficient intrusion detection system (IDS) [6].

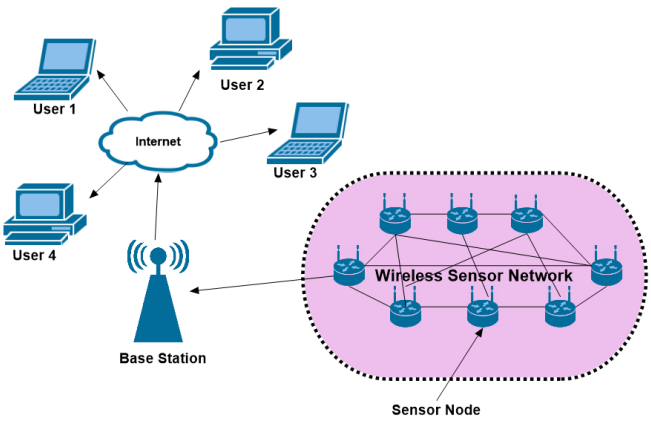

Fig. 1. Wireless Sensor Networks Architecture

Sensor networks must be autonomous and sensitive to evolutionary changes in real time, without user or administrator input. When it comes to security risks, this need is even more urgent. It makes sense to use an IDS that can find third-party attempts to take advantage of potential attacks and warn of malicious attacks.

WSNs communicate via transceivers [7], [8]. IDS can find WSN threats by collecting and analyzing data to find intrusions and misuse of systems and networks.

Wireless IDS may monitor and analyze user and system actions, identify known attack patterns, and detect WSN intrusion attempts. Each sensor should report to a base station to prevent missing an essential activity. Intrusion detection is

First Author and Second Author are with Center for Telecommunications, Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg, South Africa (e-mail: josephkips01@gmail.com, tgswart@uj.ac.za).

Third Author is with Center for Telecommunications, Department of Electrical and Electronic Engineering Science, University of Johannesburg, Johannesburg, South Africa and Department of Electrical and Electronic Engineering, University of Botswana, Gaborone, Botswana (e-mail: drebenic4real@gmail.com).

about detecting system intrusions. We use IDS so that we can know when attackers have gotten into the WSN.

The two-level WSN security are: First-level network protection uses cryptography and firewalls. The IDS protects against second-level internal attacks [9]. IDS only detects intrusions and does not prevent them [10]. Intrusion is the illegal access, modification, and forwarding of network packets. IDSs are classified into four categories, namely signature-based, anomaly-based, specification-based, and hybrid-based IDSs [11], [12]. Using attack rules, the signature-based IDS detects network attacks [13]. Using statistical, data mining, machine learning, and AI approaches, anomaly-based IDS identifies attacks. Manually generated rules in specification-based IDS identify known and undiscovered threats. The hybrid IDS combines signature, anomaly, and specification-based approaches [14], [15].

WSNs are vulnerable to attacks because of limited processor power, battery life, storage capacity, wireless bandwidth, communication range, and random sensor node placement [14]. In WSN, attackers may simply target sensor nodes. Attackers can target different network levels. The IDS detects those WSN attackers; thus, the WSN IDS must consider restricted processing, energy, and bandwidth [16]. IDS detection rates go down because wireless sensor devices and network technologies create a lot of data.

Due to network dynamics and the power needed to analyze massive amounts of data from a remote environment, detecting an intruder with high accuracy is difficult. IDSs are also needed for user identification, authorization, and suspicious activity detection. Intrusions are malicious attempts to access a network and perform illegal actions. IDS identifies those harmful illegal actions for a better secured network. The IDS in the WSN architecture is shown in Figure 2.

This research presents an evaluation and implementation of an IDS algorithm in WSN.

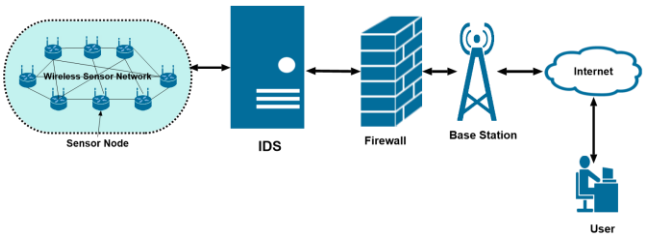

Fig. 2. Intrusion detection in wireless sensor networks

II. Literature Survey

Govindasamy and Punniakodi [17] present an efficient IDS WSN using ZigBee. The study shows significant energy, memory, and processing consumption. In this study, Energy Efficient Intrusion Detection Systems (EE-IDS) and Energy Efficient Intrusion Detection Systems with Energy Prediction (EE-IDSEP) are made to protect ZigBee-based WSNs from wormhole (an attack that causes all received packets to be routed via a pair of malicious nodes [18], [19]) and network threats.

Singh *et al.* [20] discussed an IDS in WSN using an advanced fuzzy IDS. In this research, an Advanced Hybrid Intrusion Detection System (AHIDS) that automatically identifies WSN intrusions is presented. AHIDS utilizes a cluster-based architecture with an improved Leach algorithm to decrease the amount of energy consumed by sensor nodes. It uses fuzzy rule sets and the Multilayer Perceptron Neural Network to figure out when something is not right.

Belavagi and Muniyal [21] present a multiple IDS WSN using the Routing Protocol for Low Power and Lossy Networks (RPL). RPL networks have a huge variety of small sensor nodes. A border router connects these nodes to the Internet. Therefore, these nodes can be attacked. The IDS model can detect ranked attacks, selective forwarding, wormholes, and denial of service (DoS). The research focuses on identifying several intrusions in networks with 10, 40, and 100 nodes with malicious nodes. The Cooja simulator was used for simulation and evaluation.

Titorenko and Frolov [22] present new methods for detecting network intrusion. It outlines the benefits and drawbacks of the current IDS. False positive issues and IDS repercussions are discussed. Software, hardware, hardware-software, and organizational methods are used to protect data today. All these techniques improve network security. In [23], IP packets are duplicated. Destination route entries estimate the amount of duplication. NS2 provides a tree-ID for each transmitted packet, and the receiver's IP layer eliminates redundant packets. This approach decreases packet replication attacks.

In [24], IDS was implemented using MANETs. NS2 simulation was used to test the system, and the AODV routing protocol was used to route packets quickly and easily.

To identify the location of a given event using wireless sensor networks, the approach in [25] employs sensor binary beliefs, a probability matrix whose greatest value indicates event location. This approach also produces accurate and fault-tolerant results.

Jian *et al.* [26] describe WSN intrusion detection. Semantic and multi-agent algorithms are used. Layers in the framework include the network layer (which specifies the topology), the semantic layer (which relates to security ontology), the model layer, and the cooperative layer (which refers to how the nodes co-operate between each other for intrusion detection). They have characterized agent nodes and common nodes, with the latter including the intrusion detection model. The sensor nodes collect data and transfer it to the agent node for detection. Another algorithm, called "detect intrusion," checks the result for intrusion detection by using security ontology components.

Combining intrusion detection and prevention technologies increases network latency, and the node uses a lot of energy. An energy-efficient routing protocol to address the problem is presented in [7]. The approach comprises three stages: initial construction, data transfer, and re-construction. Network routing and topology are built in the first stage, which is initial construction. In the data transfer, the node sends an event. Reconstructing the network architecture and routing table reduces communication overhead and energy consumption [27].

Chhaya *et al.* [28] present a smart grid WSN IDS for cybersecurity and topology control. Smart grid technology can revolutionize the power grid, and the adoption of this technology necessitates the integration of electrical and communication infrastructure. Smart grid technology features full-duplex communication; automated metering; renewable energy integration; distribution automation; and power grid monitoring and management. They are vulnerable to cyberattacks due to their low computing power. Since a smart grid will contain crucial and vital electrical power grid

infrastructure, cyberattack defenses must provide privacy, data availability, and security.

WSN design issues include node localization and intrusion detection. The research in [29] provides a multi-objective manta ray foraging optimization (MRFO) based node localization with an IDS approach for WSN. The aim is to detect unknown nodes and detect network intrusions.

Kathirvel and Subramaniam [30] present an improved IDS and response for WSN. Recent network attacks influence lifespan, throughput, latency, energy consumption, and packet loss. Traditional network security techniques like IDS will not be enough. In this study, researchers devised two different improved IDSs. First, the suggested improved IDS uses an optimization algorithm to generate optimum clusters. Second, the researchers determine each sensor node's trust value using an algorithm called a multi-purpose differential evolution.

III. Research Method

A. WSN IDS using RPL protocol

Cooja is compatible with a wide variety of wireless sensor nodes, sometimes called motes. As a means of implementing various nodes, we chose to use SkyMote.

To detect network intrusions, malicious mote activity is simulated. The simulation is evaluated by utilizing packet transmissions to identify intrusions, and IDS overhead traffic analysis is recorded based on power consumption and attack detection. Figure 3 depicts the IDS using the RPL protocol workflow. Tables I and II show the RPL protocol algorithm and the simulation parameters, respectively.

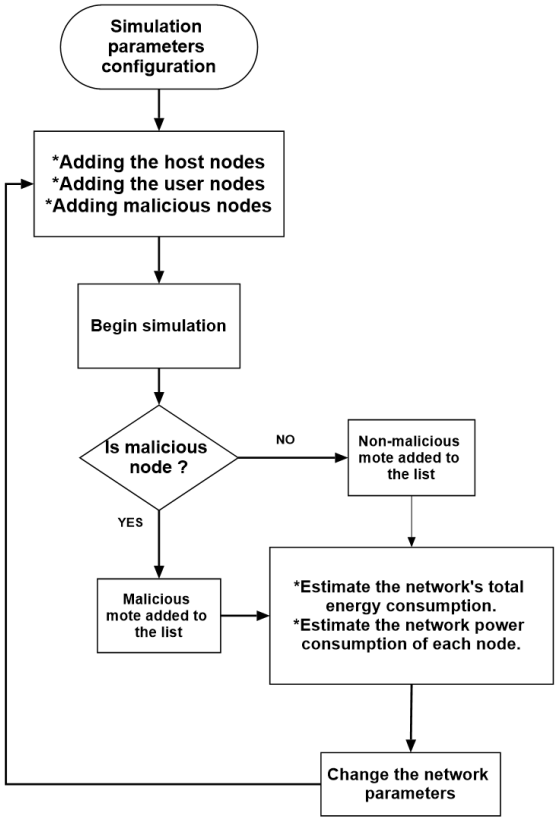

Fig. 3. Flowchart of IDS in WSN using Cooja simulator

The simulation begins by identifying the host node and user nodes. In addition, malicious nodes are injected into the network. Real-time node communication identifies intrusions. When an attacker attacks an RPL-based 6LoWPAN network, they can use compromised nodes to provide false rank information to the IPv6 Mapper about themselves and their neighbors. Table I shows the initial network parameters.

TABLE I
IDS IN WSN USING RPL PROTOCOL ALGORITHM

Algorithm 1: IDS in WSN using RPL protocol	
1:	**For** every node in the Network IDS **do**
2:	Estimate the total energy consumption
3:	Estimate the network power consumption
4:	**If** the node is malicious then
5:	The mode is added to the malicious node list
6:	**Else**
7:	Mote is added to the non-malicious list
8:	**End if**
9:	**End for**

TABLE II
SIMULATION PARAMETERS USED FOR IDS IN WSN USING RPL PROTOCOL

Parameters	Values
Operating system	InstantContiki2.7
Simulator	Cooja
Topology	Random positioning
Routing protocol	RPL
Number of collect-view sensor nodes	10
Number of IoT nodes	5
Number of resources node	3
Number of attacker nodes	3
Simulation time	20 minutes
Node type	SkyMote
Transmission range	50 m
Radio channel	26
CSMA ContikiMAC channel check rate	8 Hz
MAC layer	CSMA
Link layer	ETX
Number of bytes collected from CollectView-Mote	140 bytes
Number of bytes collected from Mote-CollectView	57807 bytes

B. WSN IDS using AODV protocol

The traditional IDS has three components: a data-collection component, a response component and a detection component. Each component is designed at a different stage to meet the specific requirements of IDS, with the data collection component collecting WSNs data to feed into the detection component, which in turn feeds into the response module, which is then executed based on the detection component's output. The components involved in wireless transmission are: receiving, transmitting with a delay, detecting, responding, and routing data packets between nodes. The procedure as a whole will drain the computational power of the microcontroller unit and the energy of the nodes.

Because malicious nodes are randomly chosen throughout the simulation process, once they get the request packet, the malicious node will immediately shut down its IDS and send out a huge number of response packets. Once there are DoS attacks

on WSN nodes, the targeted node uses IDS to figure out if it was attacked or not. If it was, it takes the steps needed to get rid of the bad nodes and get the network back to normal operation and peak performance.

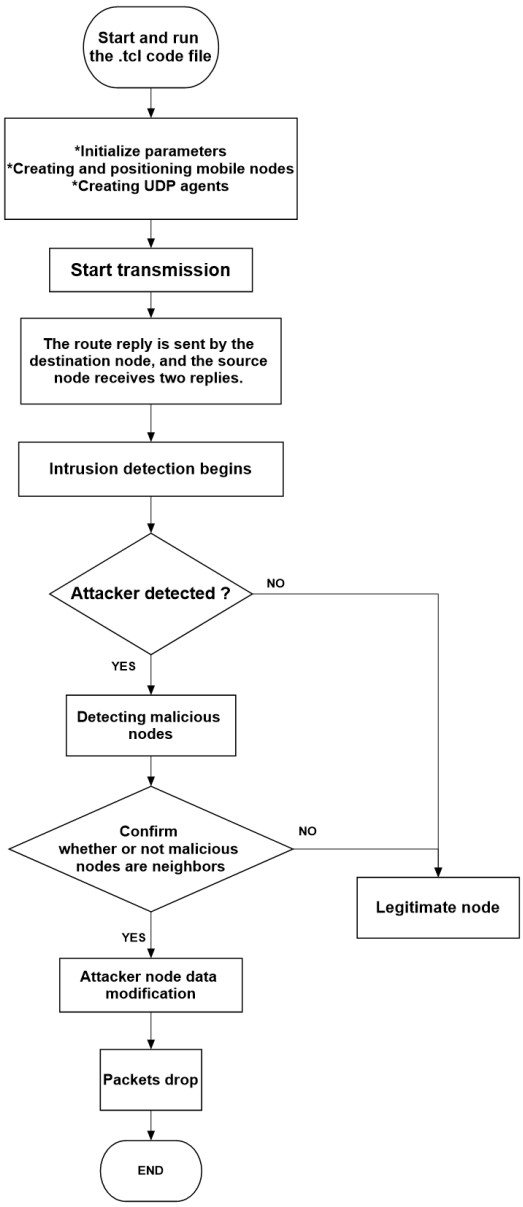

Fig. 4. Flowchart of IDS in WSN using NS2 for the AODV protocol

TABLE IV
IDS IN WSN USING AODV PROTOCOL ALGORITHM

Algorithm 2: IDS in WSN using AODV protocol	
1:	Initialize parameters
2:	Creating and positioning mobile nodes
3:	Start transmission
4:	**If** the attacker is detected
5:	Detect malicious nodes
6:	**If** malicious nodes are neighbors, **then**
7:	Attacker node data modification
8:	Packets drop
9:	**Else**
10:	No packet drops from legitimate nodes
11:	**End if**
12:	**End if**

The detection component is important to the IDS and has a significant effect on network throughput. DoS attacks on AODV-based WSNs are simulated with the help of network simulation software NS2. Analyses are performed on the trace files that are produced by the simulation model. Figure 4 shows the workflow of IDS in WSN using NS2 for the AODV protocol and its algorithm is shown in Table IV. Table V shows the simulation parameters for IDS in WSN using the AODV protocol.

TABLE V
NS2-SIMULATION PARAMETERS USED FOR IDS IN WSN USING AODV PROTOCOL

Parameters	Values
Operating system	Ubuntu 22.04
Simulator	NS2.35
Channel type	Wireless
Antenna type	Omni antenna
Link layer type	LL
Interface queue type	DropTail
Network interface type	Wireless Phy
MAC type	Mac/802_11
Number of mobile nodes	40 nodes
Routing protocol	AODV
Area	800 x 800 m^2
Transmission type	CBR
Packet size	512 bytes
Transmission protocol	UDP
Simulation time	150 seconds
Propagation model	Two-ray ground
CBR bandwidth	0.05 Mbs
Tx power	0.6 W
Rx power	0.3 W
Initial energy	1000 J

IV. COOJA EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we cover the experimental construction of a WSN simulator, collecting network data, and evaluating it using an RLP-based IDS model that we built. The experiments are carried out in the Cooja simulator, which runs on the InstantContiki2.7 operating system installed on the VMware workstation.

The SkyMote sensor type is used to design the WSN architecture because it is compatible and can work with a lot of sensors.

A simulated network of 18 sensor nodes, including three malicious nodes, is depicted in Figure 5. Malicious nodes (nodes 16, 17 and 18) in the simulation perform DoS attacks on the network. Through an IDS-based monitor node, sensor nodes that have been added to the network can talk to each other.

Collect View is used to collect network power usage statistics. As part of the malicious node implementation and traffic flow management, we modify the usual behavior of sensor nodes to cause them to generate the desired attack. The malicious node decreases the packet delay timer, resulting in a significant increase in network traffic. Because of this, it makes network sensors that are being attacked use more CPU power.

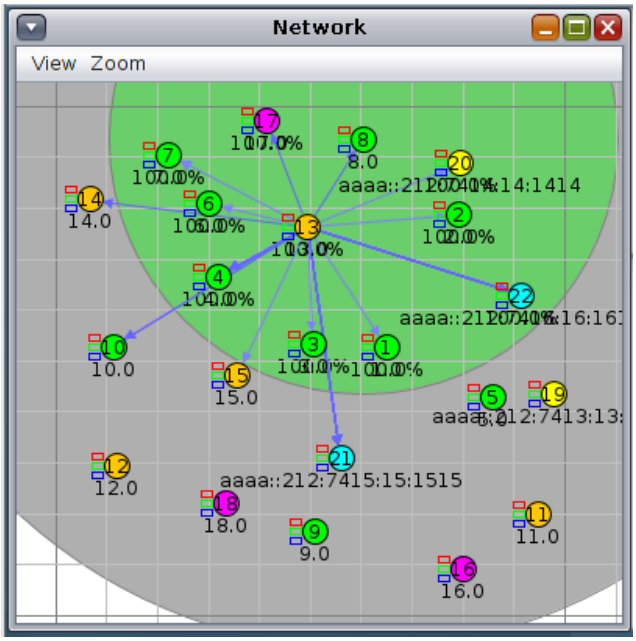

Fig. 5. Network simulation showing malicious nodes and non-malicious node with ongoing traffic

Figures 6 and 7 show the average and instantaneous power consumption (in mW) respectively, for the 18 nodes under attack, including the malicious nodes. CPU power is the amount of energy needed to run each node's tasks. The Low Power Mode (LPM) is the name for the total amount of energy used by a sensor node when it is in a power-saving mode.

The power consumption of each sensor node is divided into four parts, with the yellow component indicating the power consumption of radio communication. Second, the green portion represents radio listening power consumption, the blue portion represents CPU power consumption, and the red portion represents LPM power usage. The network devices are examined based on their power usage. If any node uses more power than expected, malicious activity could happen on the network. Otherwise, it is seen as normal behavior.

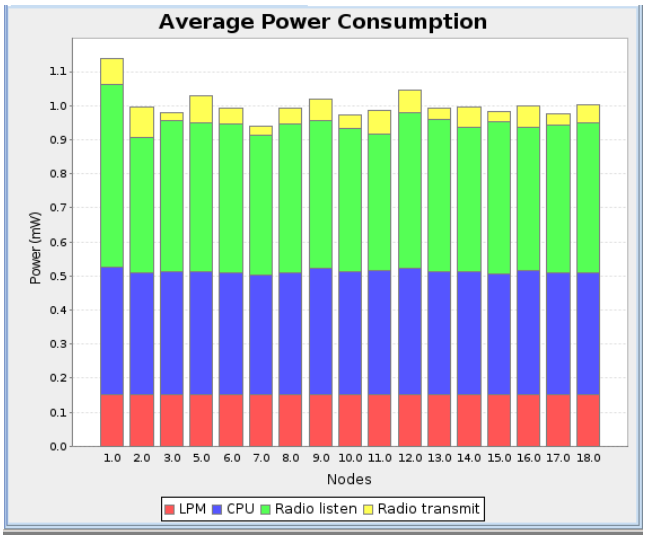

Fig. 6. Average power consumption under attack

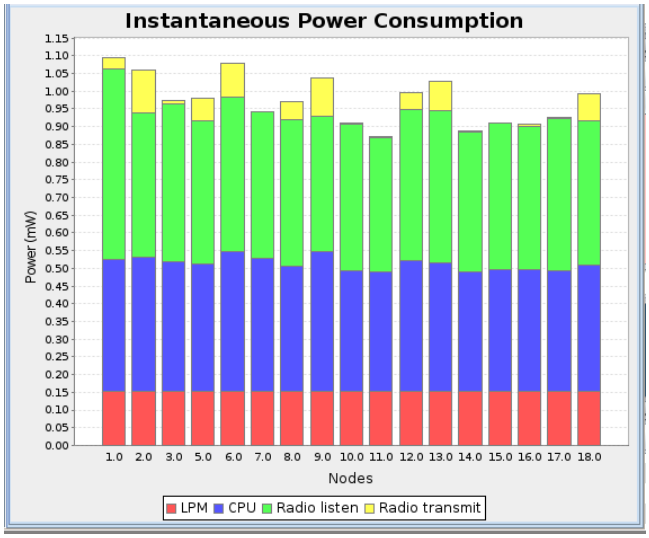

Fig. 7. Instantaneous power consumption under attack

In Figure 8, we see that node 5 consumes more power (1.94 mW) with respect to time compared to the other nodes. During intrusion detection, the malicious node 16 uses 1.63 mW more power than other malicious nodes.

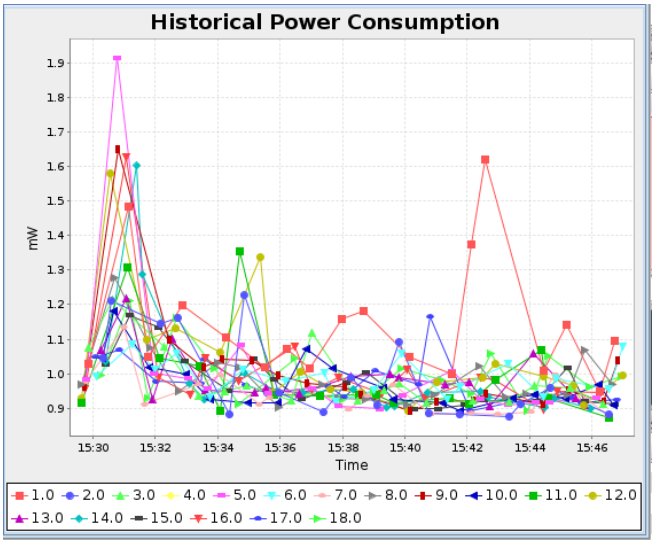

Fig. 8. Historical power consumption

Figures 9 and 10 show the total packets received and the average routing metric of the sensor nodes, respectively, with respect to time. Based on what we learned from our simulation, the number of packets received during an intrusion keeps going back and forth between 10 and 22 packets with respect to time, and the total number of packets received from the 15 nodes is 200. The routing metrics in the Cooja simulator are computed using expected transmissions (ETX). It calculates the estimated number of data transfers from the source node to the destination node. Whenever a source node transfers data packets, the destination node acknowledges receipt of the packets. Otherwise, it indicates packet loss.

The ETX equation is given as:

$$ETX = \frac{1}{df \times dr} \quad (1)$$

where *df* is the forwarding ratio (probability of a successful transmission of packets from sources to destinations) and *dr* is the reversing ratio (the reverse probability from the destination to the source) [31].

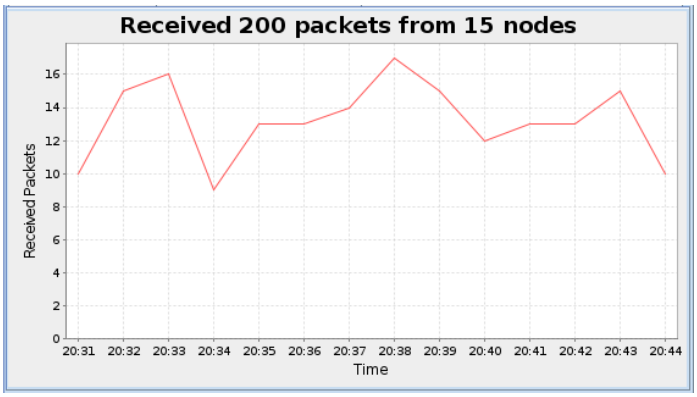

Fig. 9. Number of packets received over time

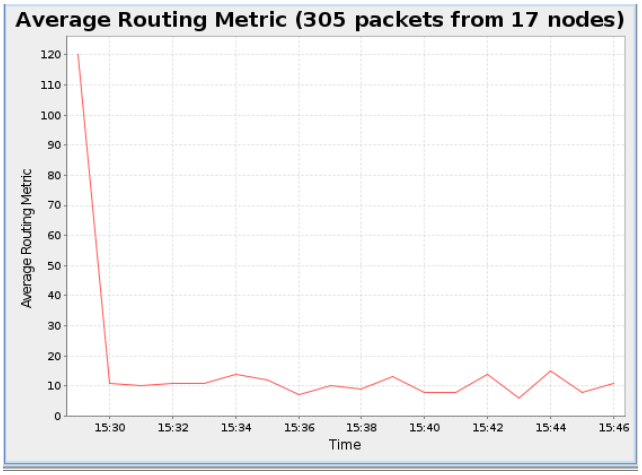

Fig. 10. Average routing metric

V. NS2 EXPERIMENTAL RESULTS AND ANALYSIS

A. *Simulations and experiments*

Upon executing the TCL file as shown in Figure 11, the graph in Figure 12 displays an animated illustration of an IDS attack network based on nodes scattered randomly according to the AODV protocol on a square surface of 800 by 800 m^2 and also shows packet drops during attacks. Gnuplot was used to build graphs from the runtime data files.

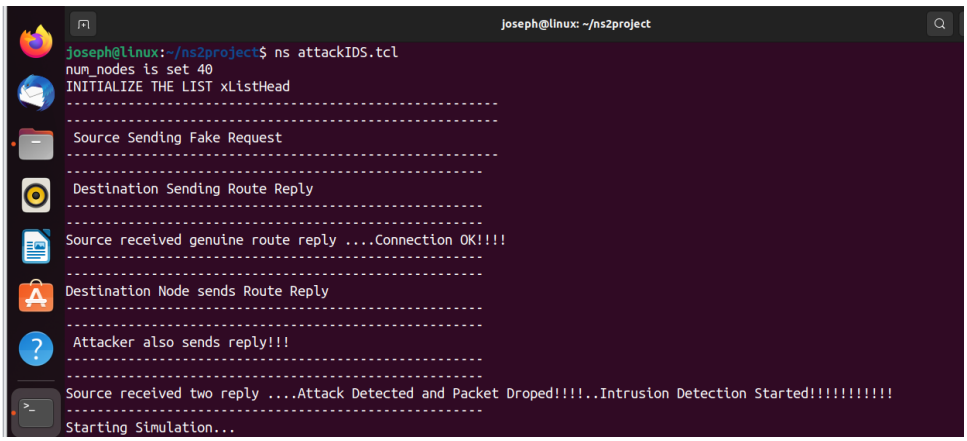

Fig. 11. Execution of the TCL file on Ubuntu terminal

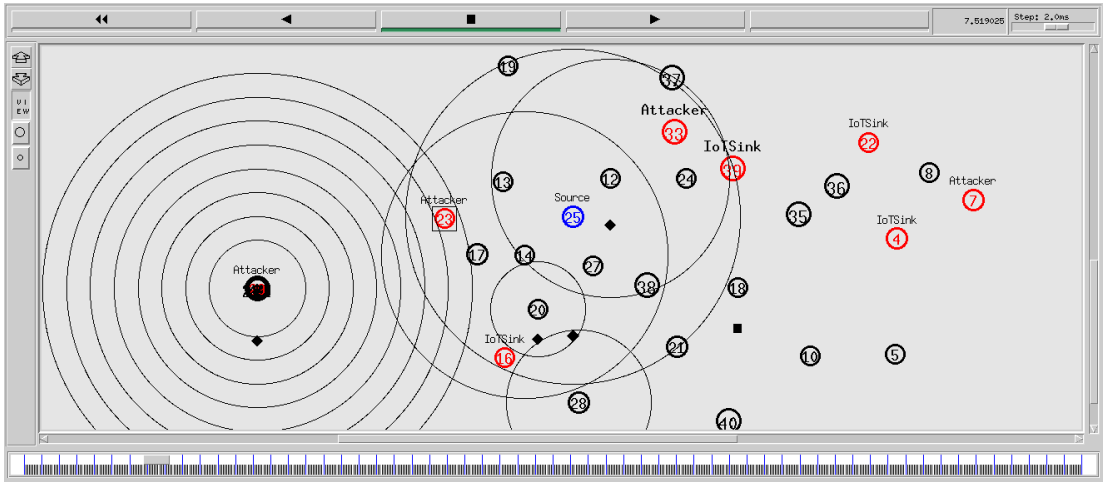

Fig. 12. Network IDS simulation using the AODV protocol with packet drops during attacks

B. Performance metrics

The binary classification issue applies to the WSN intrusion detection technology. It may be represented by a 2x2 confusion matrix, as illustrated in Figure 13. The predicted classes might be either yes or no. Yes, means that there is an attacker node in the network; no means that there is a normal node in the network.

	Predicted No (normal node)	Predicted Yes (attacker node)
Predicted No (normal node)	TN	FP
Predicted Yes (attacker node)	FN	TP

Fig. 13. Confusion matrix

There are four parts to a confusion matrix: TPR (True Positive Rate), the prediction is yes (attacker node is present), and we have a malicious node. TNR (True Negative Rate), the prediction is no (normal node is present), and we have no malicious node. FPR (False Positive Rate), the prediction is yes (the attacker node is present), but we don't have a malicious node. FNR (False Negative Rate), the prediction is no (the normal node is present), but the malicious node is present.

1) Detection accuracy

The number of occurrences is known as the detection accuracy (DA), given as:

$$DA = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (2)$$

2) True Positive Rate

TPR is also referred to as sensitivity or detection rate. It indicates the proportion of successfully detected attacker nodes, and is given by:

$$TPR = \frac{TP}{TP + FN} \quad (3)$$

3) False Positive Rate

The FPR describes the false alarm's probability, and it represents the actual attacker nodes' percentage that were expected to be normal nodes. Its equation is given as:

$$FPR = \frac{FP}{FP + TN} \quad (4)$$

4) False Negative Rate (FNR)

FNR is the normal flow percentage expected to be abnormal flow. Its equation is given as:

$$FNR = \frac{FN}{FN + TP} \quad (5)$$

5) Throughput

Throughput is defined as the total number of active packet arrivals detected at the destination during a given time period divided by the time. Its equation is given as:

$$Throughput = \frac{Number\ of\ packet\ sent}{Time} \quad (6)$$

C. Experimental results

Figure 14 shows the average UDP throughput (in kbps) versus time. It shows that the throughput increases during attacks. Figure 15 shows the throughput value, which is about 206.873 kbps and the end-to-end delay is 2.6219 seconds in the Ubuntu terminal using the throughput *awk* script and the output trace files.

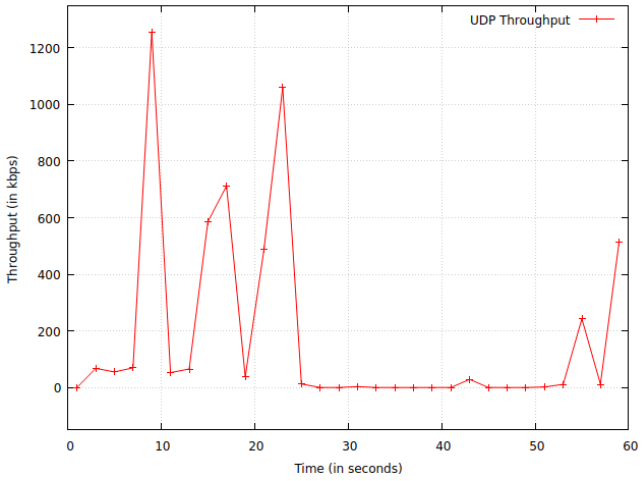

Fig. 14. Average UDP throughput

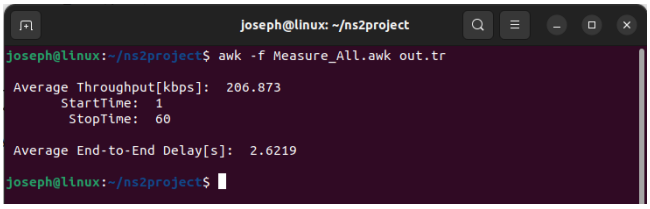

Fig. 15. Throughput in kbps

Using the *Measure_All.awk* script and the output trace file, the calculated packet delivery ratio (PDR) is 0.1423, which is 14.23% as seen in Figure 16. The total energy dissipated is 990 J with an average of 24.75 J, as seen in Figure 17 using the *Energy1.awk* script and the output trace file in the Ubuntu terminal.

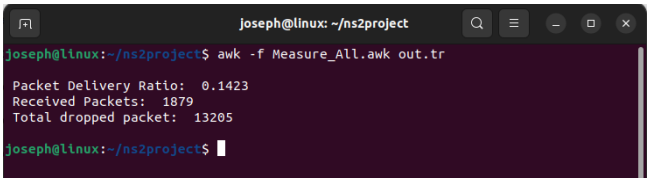

Fig. 16. Calculated packet delivery ration

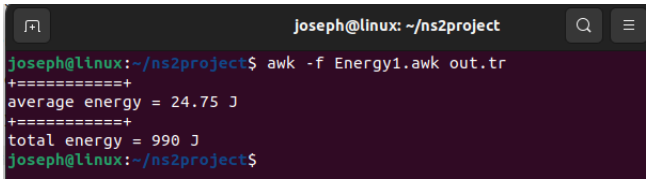

Fig. 17. Calculated energy dissipated

The data from the randomly deployed sensor nodes has been sent on to the closest sink node via a direct transmission, saving considerable energy. By reducing the amount of energy used during network transmission, the remaining energy has been used to improve the energy balance inside the cluster nodes.

Figure 18 illustrates the average energy consumption of the nodes during intrusion detection. At the start of the simulation, we give each sensor node an initial energy value of 1000 joules. This is referred to as the initial energy. The energy in simulation is used to depict the level of WSN node energy at any given moment. The initial energy value is supplied as an input parameter. Every packet transmitted and received by a sensor node consumes a certain amount of energy. As a consequence, the initial energy value of a sensor node decreases. The level of a WSN node's energy consumption at any moment in the simulation is calculated by subtracting the current amount of energy from the initial energy value. When a sensor node's energy level approaches zero, it can no longer send or receive packets.

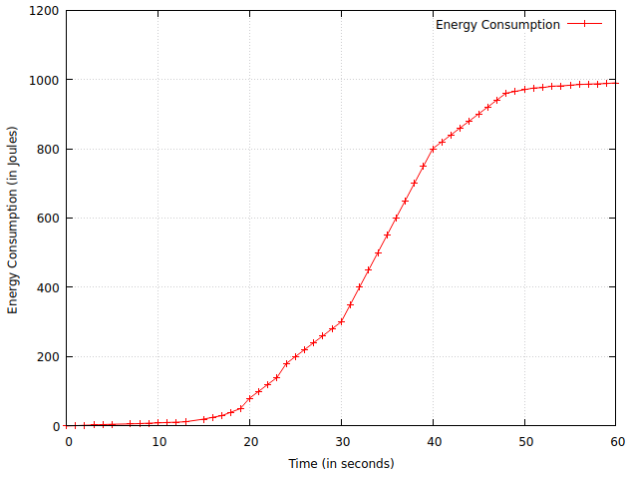

Fig. 18. Energy consumption

Figures 19 and 20 illustrate the FPR and the DA, respectively. When the number of nodes increases, the FPR increases as well. The total rate for 40 nodes is about 83%. The RPL detection accuracy is 36% greater than the AOVD detection accuracy for 20 sensor nodes.

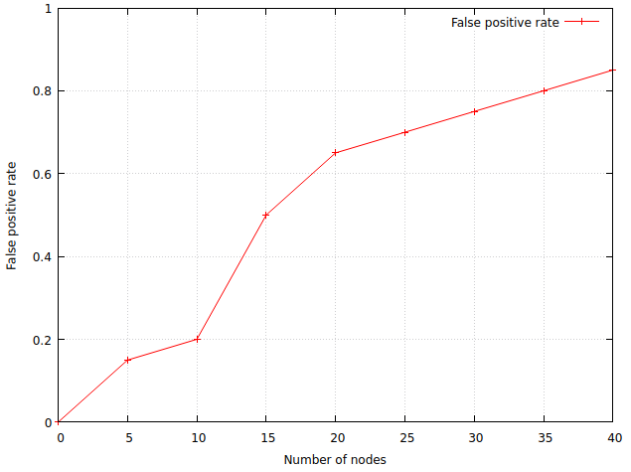

Fig. 19. False positive rate

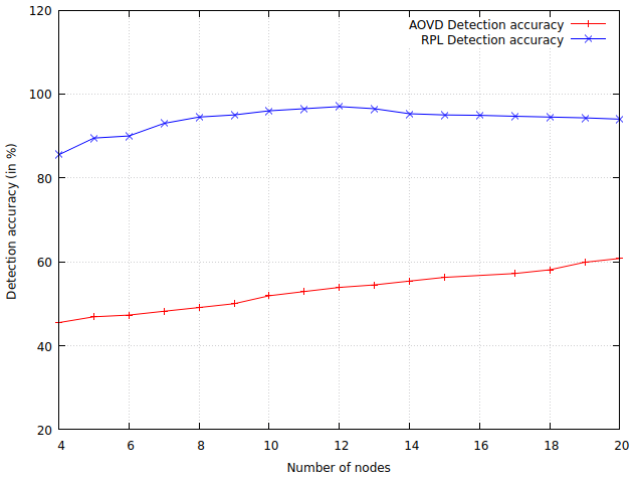

Fig. 20. AOVD and RPL detection accuracy

The network lifetime is calculated as the time it takes for the particular sensor node to run out of energy, since each node is programmed to transmit the collected data to the sink node through wireless transmission. The large number of sensor nodes in the network increases network lifetime. AOVD and RPL network lifetimes are illustrated in Figure 21. We can see that, as the number of nodes increases, so does the network lifetime, and RPL's network lifetime is greater than AOVD's network lifetime.

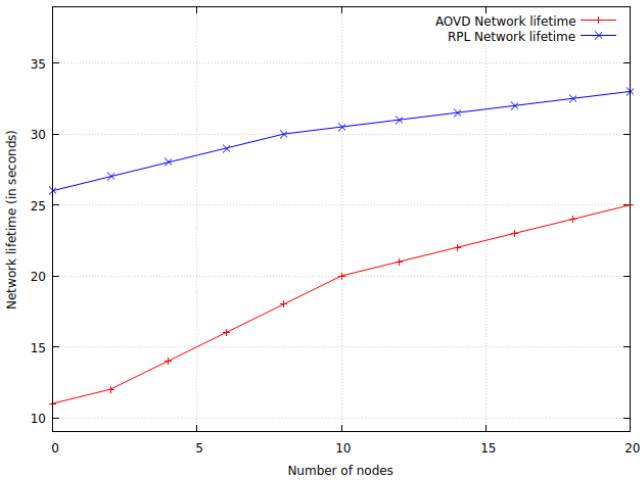

Fig. 21. AOVD and RPL network lifetimes

VI. Conclusion and Future Work

In this study, the design and implementation of IDS in WSN were elaborated in depth using both the RPL protocol in the Cooja simulator on InstantContiki2.7 in VMware workstation and the AODV protocol in NS2 in the Ubuntu operating system. Different attacks are detected on these simulated networks, and the network's performance is investigated for several parameters such as detection accuracy, false positive rate, throughput, average power consumption, and energy dissipation.

We also noticed that the simulation of network IDS consumes a lot of power during an attack, with more CPU power and listening power consumption than usual. This will have an effect on WSN-IoT devices, causing them to be disabled. Based on the results of the analysis, one of the easiest ways to spot an attack

is to look at how much CPU power is being used by normal nodes.

The results of the experiment in the Cooja simulator show that detection is 97% accurate, which is 36% more accurate compared to the NS2 DA using the AOVD protocol. The experiment in NS2 displays the packet drop during intrusions.

Implementing a machine-learning model in sensor nodes takes more time because of the training and testing that must be done on the algorithm. So, as future work, we need to come up with small machine learning models for implementing IDS in software-defined wireless sensor networks (SDWSN) so that we can use less memory space.

APPENDIX

Abbreviations used in this article are summarized in Table VI.

TABLE VI
LIST OF ABBREVIATIONS

Abbreviations	Interpretations
RPL	Routing Protocol for Low Power and Lossy Networks
AOVD	Ad-hoc On-Demand Distance Vector
NS2	Network Simulator 2
WSN	Wireless Sensor Network
IDS	Intrusion Detection System
NIDS	Network Intrusion Detection System
IPv6	Internet Protocol Version 6
IPv4	Internet Protocol Version 4
6LoWPAN	IPv6 over Low-power Wireless Personal Area Network
IoT	Internet of Things
MANETs	Mobile Ad-hoc Networks
EE-IDS	Energy Efficient Intrusion Detection Systems
EE-IDSEP	Energy Efficient Intrusion Detection Systems with Energy Prediction
AHIDS	Advanced Hybrid Intrusion Detection System
DoS	Denial of Service
MRFO	Multi-objective Manta Ray Foraging optimization
CPU	Central Processing Unit
ETX	Expected Transmission
CSMA	Carrier-Sense Multiple Access
UDP	User Datagram Protocol
LL	Link Layer
CBR	Constant Bit Rate
MAC	Media Access Control
LMP	Low Power Mode
TPR	True Positive Rate
TNR	True Negative Rate
FPR	False Positive Rate
FNR	False Negative Rate
DA	Data Accuracy
PDR	Packet Delivery Rate
SDWSN	Software-Defined Wireless Sensor Networks
TCL	Tool Command Language

REFERENCES

[1] I. Butun, S. D. Morgera, and R. Sankar, "A Survey of Intrusion Detection Systems in Wireless Sensor Networks," *IEEE Commun. Surv. TUTORIALS*, vol. 16, no. 1, pp. 266–282, 2014, https://doi.org/10.1109/SURV.2013.050113.00191

[2] J. Kipongo, E. Esenogho, and T. G. Swart, "Efficient topology discovery protocol using IT-SDN for software-defined wireless sensor network," *Bull. Electr. Eng. Informatics*, vol. 11, no. 1, pp. 256–269, 2022, https://doi.org/10.11591/eei.v11i1.3240

[3] K. Nikolina, "Overview of the progress of IPv6 adoption in Croatia," *2022 45th Jubil. Int. Conv. Information, Commun. Electron. Technol.*, pp. 405–408, 2022, https://doi.org/10.23919/MIPRO55190.2022.9803479

[4] N. Chuangchunsong, "Performance Evaluation of IPv4 / IPv6 Transition Mechanisms : IPv4-in-IPv6 Tunneling Techniques," *Int. Conf. Inf. Netw. 2014*, pp. 238–243, 2014, https://doi.org/10.1109/ICOIN.2014.6799698

[5] A. Al-Nasser, R. Almesaeed, and H. Al-Junaid, "A comprehensive survey on routing and security in mobile wireless sensor networks," *Int. J. Electron. Telecommun.*, vol. 67, no. 3, pp. 483–496, 2021, https://doi.org/10.24425/ijet.2021.137838

[6] O. A. Osanaiye and A. S. Alfa, "Denial of Service Defence for Resource Availability in Wireless Sensor Networks," *IEEE Access*, vol. 6, pp. 6975–7004, 2018, https://doi.org/10.1109/ACCESS.2018.2793841

[7] G. Divyashree, A. Durgabhavani, A. Gudoor, and M. B. Shetty, "Intrusion Detection System In Wireless Sensor Network," *Int. J. Recent Technol. Eng.*, vol. 8, no. 1, pp. 2047–2051, 2019.

[8] E. Baraneetharan, "Role of Machine Learning Algorithms Intrusion Detection in WSNs : A Survey," *J. Inf. Technol. Digit. World*, vol. 02, no. 03, pp. 161–173, 2020.

[9] S. Godala and R. P. Vaddella, "A Study on Intrusion Detection System in Wireless Sensor Networks," *Int. J. Commun. Networks Inf. Secur.*, pp. 127–141, 2020, https://doi.org/10.17762/ijcnis.v12i1.4429

[10] N. Islam, F. Farhin, I. Sultana, and M. S. Kaiser, "Towards Machine Learning Based Intrusion Detection in IoT Networks," *Tech Sci. Press*, pp. 1801–1821, 2021, https://doi.org/10.32604/cmc.2021.018466

[11] V. Gowdhaman and R. Dhanapal, "An intrusion detection system for wireless sensor networks using deep neural network," *Soft Comput.*, pp. 1–9, 2021, https://doi.org/10.1007/s00500-021-06473-y

[12] S. Rizwana, K. M. Gayathri, and N. Thangadurai, "Intrusion Detection Algorithm for Packet Loss Minimization in Wireless Sensor Networks," *Int. J. Eng. Adv. Technol.*, vol. 8958, no. 6, pp. 69–74, 2019, https://doi.org/10.35940/ijeat.E7453.088619

[13] I. Gupta and K. Gupta, "Evaluation of Intrusion Detection Schemes in Wireless Sensor Network," *Int. Organ. Sci. Res. J. Comput. Eng.*, vol. 18, no. 2, pp. 60–63, 2016, https://doi.org/10.9790/0661-1802056063

[14] S. Smys and H. Wang, "Hybrid Intrusion Detection System for Internet of Things (IoT)," *J. IoT Soc. Mobile, Anal. Cloud*, vol. 02, no. 04, pp. 190–199, 2020.

[15] M. Latah and L. Toker, "An Efficient Flow-based Multi-level Hybrid Intrusion Detection System for Software-Defined Networks," *CCF Trans. Netw. (2020*, pp. 261–271, 2020, https://doi.org/https://doi.org/10.1007/s42045-020-00040-z

[16] T. Xiaopeng, S. Shaojing, H. Zhiping, G. Xiaojun, and Z. Zhen, "Wireless Sensor Networks Intrusion Detection Based on SMOTE and the Random Forest Algorithm," *MDPI Sensors*, 2019, https://doi.org/10.3390/s19010203

[17] J. Govindasamy and S. Punniakodi, "Energy efficient intrusion detection system for ZigBee based wireless sensor networks," *Int. J. Intell. Eng. Syst.*, vol. 10, no. 3, pp. 155–165, 2017, https://doi.org/10.22266/ijies2017.0630.17

[18] Z. Tun and A. H. Maw, "Wormhole Attack Detection in Wireless Sensor Networks," *World Acad. Sci. Eng. Technol. 46 2008*, pp. 545–550, 2008.

[19] M. A. Patel and M. M. Patel, "Wormhole Attack Detection in Wireless Sensor Network," *Proc. Int. Conf. Inven. Res. Comput. Appl. ICIRCA 2018*, no. Icirca, pp. 269–274, 2018, https://doi.org/10.1109/ICIRCA.2018.8597366

[20] R. Singh, J. Singh, and R. Singh, "Fuzzy Based Advanced Hybrid Intrusion Detection System to Detect Malicious Nodes in Wireless Sensor Networks," *Hindawi Wirel. Commun. Mob. Comput.*, pp. 1–14, 2017, https://doi.org/https://doi.org/10.1155/2017/3548607

[21] M. C. Belavagi and B. Muniyal, "Multiple intrusion detection in RPL based networks," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 1, pp. 467–476, 2020, https://doi.org/10.11591/ijece.v10i1.pp467-476

[22] A. A. Titorenko and A. A. Frolov, "Analysis of modern intrusion detection system," *2018 IEEE Conf. Russ. Young Res. Electr. Electron. Eng.*, pp. 142–143, 2018, https://doi.org/10.1109/EIConRus.2018.8317049

[23] V. B. Joshi and R. H. Goudar, "Intrusion detection and defense mechanism for packet replication attack over MANET using swarm intelligence," *2013 Int. Conf. Pattern Recognition, Informatics Mob. Eng.*, pp. 152–156, 2013, https://doi.org/10.1109/ICPRIME.2013.6496464

[24] S. P. Botkar and S. R. Chaudhary, "An Enhanced Intrusion detection System using Adaptive Acknowledgment based Algorithm," *2018 IEEE Conf. Russ. Young Res. Electr. Electron. Eng.*, pp. 606–611, 2018.

[25] J. V. A. Sukumar, I. Pranav, M. M. Neetish, and J. Narayanan, "Network Intrusion Detection Using Improved Genetic k-means Algorithm," *2018 Int. Conf. Adv. Comput. Commun. Informatics*, pp. 2441–2446, 2018, https://doi.org/10.1109/ICACCI.2018.8554710

[26] W. Jian, F. Zhi-feng, and C. Yong, "Design and Implementation of

Lightweight Wireless Lan Intrusion Detection System," *IEEE Trans. Comput.*, 2012, https://doi.org/10.1109/MINES.2012.96

[27] A. Cherepanov, I. Tyshchenko, M. Popova, and D. Vakhnin, "Building Energy Efficient Wireless Sensor Networks," *Int. J. Electron. Telecommun.*, vol. 63, no. 1, pp. 45–49, 2017, https://doi.org/10.1515/eletel-2017-0007

[28] L. Chhaya, P. Sharma, G. Bhagwatikar, and A. Kumar, "Wireless Sensor Network Based Smart Grid Communications : Cyber Attacks, Intrusion Detection System and Topology Control," *MDPI Electron.*, pp. 1–22, 2017, https://doi.org/10.3390/electronics6010005

[29] R. Punithavathi, R. T. Selvi, R. Latha, G. Kadiravan, and V. Srikanth, "Robust Node Localization with Intrusion Detection for Wireless Sensor Networks," *Intell. Autom. Soft Comput.*, pp. 143–156, 2022, https://doi.org/10.32604/iasc.2022.023344

[30] A. Kathirvel and M. Subramaniam, "Improved Intrusion Detection and Response System for Wireless Sensor Network," *Int. J. Forensic Sci.*, pp. 1–22, 2020, https://doi.org/10.23880/ijfsc-16000203

[31] M. R. Rahman, M. M. Islam, E. A. Shahaz, and Y. Alsaawy, "Application Specific Energy Aware and Reliable Routing Protocol for Wireless Sensor Network," *2019 7th Int. Conf. Smart Comput. Commun. ICSCC 2019*, pp. 1–5, 2019, https://doi.org/10.1109/ICSCC.2019.8843687