2019 5th International Conference on Web Research (ICWR)

An Anomaly-Based IDS for Detecting Attacks in RPL-Based Internet of Things

Behnam Farzaneh
Department of Electrical and Computer Engineering
Isfahan University of Technology
Isfahan, Iran
b.farzaneh@ec.iut.ac.ir

Mohammad Ali Montazeri
Department of Electrical and Computer Engineering
Isfahan University of Technology
Isfahan, Iran
montazeri@cc.iut.ac.ir

Shahram Jamali
Department of Electrical and Computer Engineering
University of Mohaghegh Ardabili
Ardabil, Iran
jamali@uma.ac.ir

***Abstract*—The Internet of Things (IoT) is a concept that allows the networking of various objects of everyday life and communications on the Internet without human interaction. The IoT consists of Low-Power and Lossy Networks (LLN) which for routing use a special protocol called Routing over Low-Power and Lossy Networks (RPL). Due to the resource-constrained nature of RPL networks, they may be exposed to a variety of internal attacks. Neighbor attack and DIS attack are the specific internal attacks at this protocol. This paper presents an anomaly-based lightweight Intrusion Detection System (IDS) based on threshold values for detecting attacks on the RPL protocol. The results of the simulation using Cooja show that the proposed model has a very high True Positive Rate (TPR) and in some cases, it can be 100%, while the False Positive Rate (FPR) is very low. The results show that the proposed model is fully effective in detecting attacks and applicable to large-scale networks.**

Keywords— IoT; RPL; 6LoWPAN; IDS; Neighbor Attack; DIS Attack

I. INTRODUCTION

The Internet of Things (IoT) was first presented by Kevin Ashton in 1999 [1]. The core concept of the IoT is the development of environmental intelligence and independent control. The IoT describes the next generation of the Internet that objects are physically recognized and accessed through the Internet individually [2]. In the future, according to Cisco's forecast, the number of Internet-connected objects will increase to 50 billion in 2020. In fact, the Internet enables objects of the world, in which everything, such as physical objects and devices, is self-identified and intelligent, allow computers to manage and organize them [3].

In the IoT, security should be provided for all devices and information resources. The IoT includes Low-Power and Lossy Networks (LLN) with very low-security capabilities for internal and external attacks. The LLN networks are limited in terms of power, memory and power processing, and operate on batteries. It is also connected to a variety of communication links such as IEEE 802.15.4. These networks use IPv6 and also have high loss rates and low data rates [2, 4].

The thought behind the creation of IPv6 over Low-powered Wireless Personal Area Networks (6LoWPAN) was that the Internet protocol could be utilized on the smallest devices. The low number of IPv4 addresses for sensor nodes, the smallest packet length (127 bytes) in the physical layer and the long sleep most of the nodes for storing energy are major restrictions of IEEE 802.15.4 that made 6LoWPAN be created for the IoT [5]. The protocol used in these networks is Routing over Low-Power and Lossy Networks (RPL) [6], which is a distance-vector and source-routing protocol, and standardized for IoT in 2012.

Fig. 1 shows the typical stack of protocols used in the IoT.

Having a network of resource-constrained with new protocols and connected to the Internet will make security a major challenge that needs to be carefully considered. In RPL, encryption and authentication methods can only protect networks from external attack (the Internet side) and cannot protect networks against internal attack [2, 7]. Hence, the protocol is exposed to a variety of internal attacks such as sinkhole attack, selective-forwarding attack, wormhole attack, neighbor attack, DIS attack, rank attack, version number attack, worst parent attack, DAG inconsistency attack, etc. Therefore, an Intrusion Detection System (IDS) as a second line defense is needed to monitor network operations and detect intruders.

In this paper, the purpose is to provide an anomaly-based lightweight IDS for detection of neighbor and DIS attacks on the RPL protocol. The proposed method in this paper can find the source of these attacks in real-time. The IDS is distributed in every physical object of the LLN and each node monitors its neighbors to detect attacks. Also, IDS can be used to detect similar attacks. Simulations show that the proposed model is applicable to large-scale networks.

The rest of this paper is organized as follows: The foundations of the RPL protocol, IDS, neighbor attack and DIS attack are reviewed in Section II. The related work is reported in Section III. The proposed model is introduced in Section IV. The experimental results and simulation of the proposed model are presented in Section V and the paper is concluded in Section VI.

II. BACKGROUND

In this Section, we review the foundations of the RPL protocol, IDS, neighbor attack and DIS attack in the proposed model.

<table>
<tr><td>Application</td><td colspan="2">CoAP</td></tr>
<tr><td>Transport</td><td>TCP</td><td>UDP</td></tr>
<tr><td rowspan="3">Network</td><td colspan="2">RPL</td></tr>
<tr><td>IPv6</td><td>ICMP</td></tr>
<tr><td colspan="2">6LoWPAN</td></tr>
<tr><td>MAC</td><td colspan="2">IEEE 802.15.4 TSCH MAC</td></tr>
<tr><td>Physical</td><td colspan="2">IEEE 802.15.4 PHY</td></tr>
<tr><td>Layers</td><td colspan="2">IoT Protocol Stack</td></tr>
</table>

Fig. 1. Stack of protocols used in IoT

978-1-7281-1431-6/19/$31.00 ©2019 IEEE

A. The RPL Protocol

The RPL routing protocol is based on the formation of a Destination-Oriented Directed Acyclic Graph (DODAG) that has a tree-like structure with a single root (sink). The root is the final destination node in the DODAG, which acts as a Border Router (BR) and connects the LLN networks with IPv6 networks. An RPL network can have one or more DODAGs that together specify an RPL instance with a unit identifier. A network can also have several RPL instances but these instances are logically independent. A node can connect to several RPL instances but only one DODAG belongs to each instance. A node may have several parents. The choice of the parent is based on the rank. The rank of a node is defined as the position of a node associated with other nodes in the DODAG with respect to the root. The root has the lowest rank and equals 1. The rank increases with the distance from the root and decreases with the root approaches. The rank is determined by the Objective Function (OF) which is the OF metrics can be Expect Number of Transmission (ETX), hope count, energy, delay or other metrics. Various OFs such as Objective Function Zero (OF0) or the Minimum Rank with Hysteresis Objective Function (MRHOF) are presented in the RPL for rank calculation [6].

RPL networks support 3 traffic patterns: Point to Point, MultiPoint to Point, and Point to MultiPoint. In the process of constructing RPL topology, each router node in the network identifies a parent set on a path to the root of DODAG and a preferred parent is selected based on the OF. The RPL routing protocol includes 3 types of control messages: DAG Information Object (DIO), DAG Information Solicitation (DIS), and Destination Advertisement Object (DAO). Upstream paths are built with DIO control messages. The DIO messages are sent alternately by Trickle algorithm according to a timestamp setting to keep track of routing information up to date. The DIO messages are used to find an RPL instance, keep the DODAG, choose a DODAG parent set and comprehend the configuration parameters. The DAO messages are used to build downstream paths and release destination information upward in the DODAG from the child toward the root or parent. Also, the DIS message is used to discover a new neighbor and add a node to the network [6, 8].

Fig. 2 shows the architecture of the RPL routing protocol.

B. Intrusion Detection System (IDS)

The role of an IDS in the network is to detect the attacks and report them. There are different kinds of IDS which are divided into 4 types as follows [2]:

- Signature-based IDS: This method is based on the pattern match. In this method, pre-made intrusion patterns (signature) are maintained as a rule. So that each pattern includes different types of intrusion, and if there is such a pattern in the network, it will be alerted. This type of IDS is also called Misuse-based IDS.
- Anomaly-based IDS: In this method, a view of normal behavior is created. An anomaly may indicate an intrusion. To detect abnormal behavior, we must identify normal behaviors and find specific patterns and rules for them that is called normal profile. Behaviors that follow these patterns are normal, and events that deviates more than the normal range of these patterns are identified as abnormal behavior. The measures used in anomaly-based detection are threshold detection, statistical measures, rule-based measures and other measures such as clustering methods, neural networks, genetic algorithms.
- Specification-based IDS: In this method, a network expert must manually define the rules for each type of profile. Intrusion is detected when the behavior of the network deviates from the defined rules.
- Hybrid IDS: In this method, a hybrid of the above methods is used.

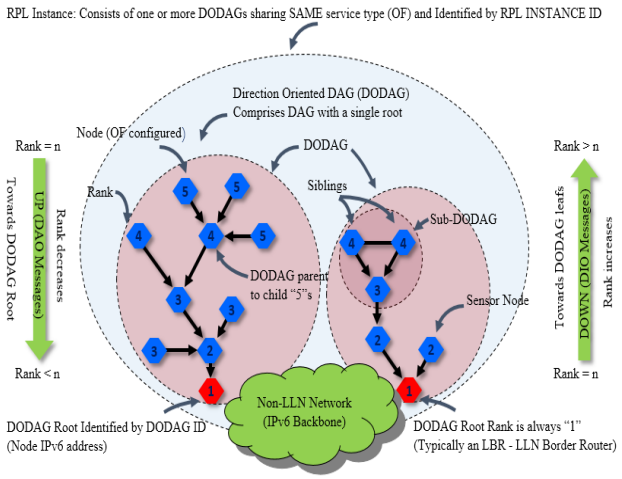

Fig. 2. The architecture of the RPL routing protocol [9]

C. Neighbor Attack

In this attack, the attacker nodes broadcast to neighbors, DIO messages received from its neighbors without adding any changes in the network. The node that receives this message may imagine that it has sent a new neighbor to the DIO message and wants to add this node to its potential parent list or select it as the parent, if that node may not be within the range of the victim node and as a result the network path changes. This attack can be dangerous in combination with other attacks like the DIS attack and the wormhole attack [10].

D. DIS Attack

DIS messages are sent to neighbor nodes to receive topological information from the network before connecting it. In this attack, the attacker exploits this feature and can send DIS messages in two ways, broadcast and unicast to its neighbors. If the DIS messages are broadcasted by the attacker, the node that receives this message reset its DIO timer and if the DIS messages are unicasted by the attacker, the message receiver sends a DIO message in response, indicating that the sender is interested in joining the network [10].

III. Related Work

Research on IDS in the IoT is still at the beginning. As mentioned, in the RPL routing protocol, security mechanisms such as encryption can prevent an external attack. However, they cannot protect the network against internal attacks [2, 7]. Various types of internal attacks in the RPL routing protocol and mechanisms to deal with them have been implemented in recent years. In the area of IDS in RPL-based networks, the methods proposed so far have focused on attacks such as sinkhole attack, selective-forwarding attack, wormhole attack, and used different methods for detections.

As a preliminary work, Lee et al. [11] provided a specification-based IDS for RPL networks that can only detect two attacks, rank and local repair but none are validated and there is no simulation or numerical analysis for them. Pongle et al. [12] provided a survey of attacks on RPL and 6LowPAN in the IoT, and have provided security mechanisms to deal with attacks using different types of IDS. A real-time hybrid IDS for the IoT called SVELTE by Raza et al. [13] which has investigated spoofed or altered information, sinkhole, and selective-forwarding attacks. The IDS has combined both of the anomaly-based and specification-based methods. The IDS modules of SVELTE are placed in centralized on the BR, and network nodes are responsible for sending RPL network data to BR and noticing BR if receive network malicious traffic.

Lee et al. [14] in the proposed lightweight IDS, consider energy consumption as a metric for analyzing the behavior of the nodes. Each node in the network monitors its energy consumption at an instance rate of 0.5 seconds and if energy consumption deviates from the threshold value, the IDS assume the node as the malicious node. Pongle et al. [15] introduced a new IDS for IoT that can detect wormhole attacks. This method uses local information from the node and the neighbor information to identify the attack. This method is very efficient in terms of energy and memory consumption. Bostani et al. [8], proposed a novel real-time hybrid IDS using Machine Learning (ML) algorithm based on the MapReduce approach that consists of anomaly-based and specification-based IDS modules for detecting sinkhole and selective-forwarding attacks.

Currently, there is only one IDS available to detect both neighbor and DIS attacks by Lee et al. [10]. The authors have provided a specification-based approach that focuses on RPL attacks. They have classified the RPL network into multiple clusters with a number of similar nodes. Each cluster in the network has a cluster head that has direct communication with the total cluster member. In each cluster, there is a sample of IDS and is done monitoring of the cluster members by the overhearing of their communication. The members of each cluster should report information about themselves and their neighbors to clusters. The proposed model is compared with the method of Lee et al. [10] in Section V.

IV. Proposed Model

In this Section, we discuss the system model and phases of the proposed IDS for RPL networks to detect the attacks as introduced in Section II.

A. System Model

The IDS placement strategy presented in this paper is fully distributed, and each node in the LLN network collects information and performs the intrusion detection. The system architecture is stand-alone. In the stand-alone architecture, each node acts as a monitor node and each node monitors its neighbors to detect attacks, independently. The detection method is an anomaly-based lightweight IDS and based on the threshold value for dealing with neighbor and DIS attacks. The proposed IDS is presented for the RPL routing protocol according to RFC6550 [6], in which the nodes are considered constant.

Fig. 3 shows the architecture of the proposed model. As you see, the 6LoWPAN network allows nodes to connect to the Internet through the 6LoWPAN Border Router (6BR). The IDS modules are also located in each node.

B. Detection of Neighbor Attack

In this paper, for detection of neighbor attack, dynamic threshold values are used for each node in the network, which is that the reception of DIO messages more than these threshold values, an estimate for detecting attacker node, therefore the proposed method is based on the number of neighbors, which acts dynamically for each node. By studying the rate of sending DIO messages by nodes to its neighbors in different states, since most of the DIO messages received by a node from its neighbors are near average, it was found that the number of messages received from its neighbors can be modeled by the Normal Distribution.

Normal distribution [2] can be considered as the most important continuous distribution among the continuous probability distributions used in the statistics. The main reason for this phenomenon is the role of normal distribution in the Central Limit Theorem (CLT). In the CLT, it is shown that under conditions, the sum of the values of the various variables that each of which has a finite average and dispersion, has a distribution very near to the normal distribution with an increasing number of variables. The normal distribution probability function has two parameters, the first determines the Average (μ), and the second determines the Standard Deviation (σ). The probability function curve around the average has a symmetric distribution. As you can see in Fig. 4, the normal distribution is called the Bell Curve. In the standard normal distribution, about 68% of values are within 1 standard deviation of the average (between -1 and 1), about 95% of values are within 2 standard deviations of the average (between -2 and 2) and also about 99.7% of values are within 3 standard deviations of the average (between -3 and 3).

In this paper, we use the normal distribution features and use the threshold values in accordance to determine dispersion as follows:

$$\text{Threshold} = \mu + k\sigma \qquad (1)$$

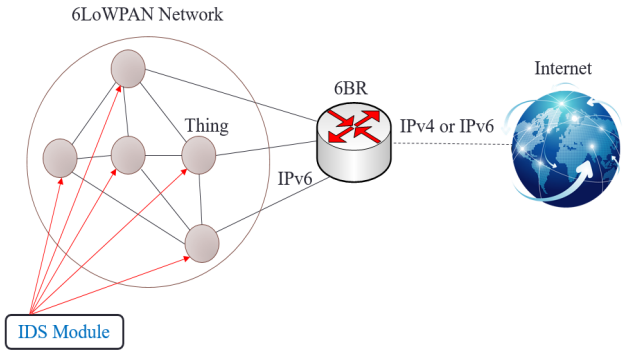

Fig. 3. The architecture of the proposed model

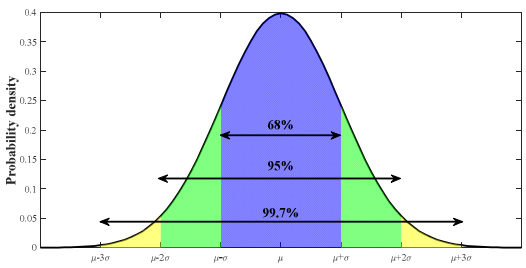

Fig. 4. Normal Distribution

where μ is the average, σ is the standard deviation and k is the coefficient that determines the distance from the average according to standard deviation.

In order to generate a profile of normal behavior, networks with different sizes were considered and separately, the influence number of neighbors on the coefficient k was investigated. For each network, the maximum value of the coefficient for the number of neighbors is stored, which with an upper approximation, is the highest coefficient of a node in the network with that number of neighbors. Hence, an equation was found to find the maximum coefficient according to the number of neighbors. Therefore, with different states and scenarios and considering the number of neighbors, as shown in Fig. 5, the coefficient was obtained according to the number of neighbors and based on which the profile of the normal behavior was formed.

Equation (2) was obtained based on the profile of normal behavior as follows:

$$k = -5E{-}05x^4 + 0.0037x^3 - 0.0899x^2 + 0.9281x - 0.7903 \quad (2)$$

where k is the standard deviation and is considered as the threshold value of normal behavior. Also, x is the number of neighbors of the node in the network.

The proposed model is presented in 4 different phases as follows:

- Phase I: Each normal node for all its neighbors counts and stores the number of received DIO messages in specific time windows.
- Phase II: Each normal node in specific time intervals according to the number of neighbors, the k value is calculated in (2) and by calculating the average and standard deviation in the current time window, produces the threshold value from (1).
- Phase III: Each normal node in specific time intervals, the number of received messages is calculated by neighbors nodes so far, and if the threshold value obtained in (1) is greater, it identifies that node as an attacker, otherwise consider it to be a normal node.
- Phase IV: The nodes identified as attackers are temporarily blocked. The reason for the temporary blocking is that if the normal nodes of the system were mistaken considered as an attacker due to the excessive transmission of DIO messages in a specific time interval, it can be to get out of block mode and interact with it.

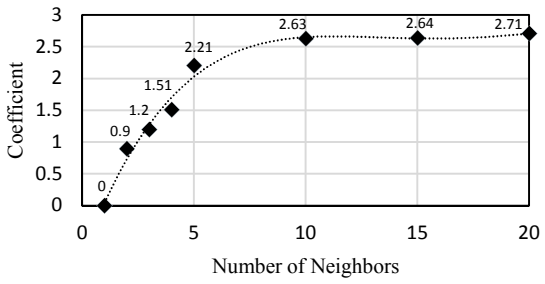

Fig. 5. The coefficient according to the number of neighbors

Algorithm 1 shows the pseudocode to detect a neighbor attack.

C. Detection of DIS Attack

In this paper, the purpose is to provide an anomaly-based lightweight IDS for detection of DIS attack. Hence, a study was conducted on the normal mode of receiving DIS messages by neighbors. As you can see in Table I, in order to generate a profile of normal behavior, networks with different sizes and at different distances were considered. Each node also stores the number of DIS messages received from its neighbors and prints the maximum received amount of its neighbors in specific time intervals. In the end, the maximum number of DIS messages received in all networks is calculated.

The maximum number of DIS messages received by neighbors in the normal mode is 3. We consider this maximum value as a threshold value for detection of DIS attack. If more than this amount of DIS messages is received from neighbors in specific time intervals, that neighbor is considered as an attacker.

As the number of neighbors is higher, the attacker is more likely to mature, but a little later detection of the attack occurs and the duration of the detections will be higher but the proposed method in both attacks has an appropriate speed for detections. In simulations in specific time intervals, the number of DIO messages received from neighbors is reset every 5 minutes. The value of block_threshold in simulations is equals 2. Temporary blocking is also 1 minute. It should be noted that in the simulation of DIS attack, the attacker randomly sends DIS messages every 5 seconds to 60 seconds.

Algorithm 2 shows the pseudocode to detect a DIS attack.

V. Evaluation

In this paper, we implement our IDS in the Contiki operating system and evaluate them using the Cooja simulator [16]. The base node for the simulation is Tmote Sky [17], which has 10kB of RAM and 48kB of ROM. Parameters required for simulation are given in Table II.

Algorithm 1. Detection of Neighbor Attack

```
Require: Count of DIO for each neighbor and number of Neighbors
        Avg= (sum_of_all_DIOs_from_neighbors)/Neighbors_num
for each neighbor in neighbor list do
        Temp+= power2(Avg - neighbor->DIO_count)
end for
    Standard_Deviation= sqrt(temp/ Neighbors_num)
    Calculate k based on Neighbors_num
for each neighbor in neighbor list do
        if ((neighbor->DIO_count) > Avg + k × standard_deviation) then
                print (neighbor is a Neighbor attacker)
                if (neighbor->block_count < block_threshold) then
                        Temporarily block neighbor
                else
                        Permanently block neighbor
                end if
    end if
end for
```

TABLE I. The Normal Mode of Receiving DIS Messages by Neighbors

		Number of Nodes			
		10	*20*	*30*	*40*
Distance	***20 m***	1	2	3	2
	30 m	2	2	1	2
	40 m	2	2	2	2

To detect both attacks, networks with a size of 20, 30, 40 nodes were considered for simulation. Then, this number of nodes is located at 20, 30 and 40 meters, in which the number of neighbors of the nodes varies. In each of the networks described, 1 node, 20% of the nodes and 30% of the nodes are categorized as attacker node. In the simulation, the simulation time is 30 minutes and there is a server node in the network. The attacker nodes start the attack, 75 seconds after starting the simulation. The range of the nodes is also 50 m. Given that the unicast or broadcast of the DIS messages does not differ in the detection of DIS attack by the proposed method, the DIS attacks are implemented as a broadcast of the DIS messages.

Fig. 6 shows how the nodes are located in the simulation, in which the green color refers to the server and the yellow and purple colors represent the normal and attacker nodes, respectively.

In this paper, the performance of the proposed method was evaluated with True Positive Rate (TPR) and False Positive Rate (FPR).

The metrics of TPR and FPR are obtained using (3) and (4), respectively as follows:

$$TPR = TP / (TP + FN) \quad (3)$$

$$FPR = FP / (FP + TN) \quad (4)$$

Algorithm 2. Detection of DIS Attack

Require: Count of DIS for each neighbor

```
for each neighbor in neighbor list do
    if ((neighbor->DIS_count) > DIS_threshold) then
        print (neighbor is a DIS attacker)
        if (neighbor->block_count < block_threshold) then
            Temporarily block neighbor
        else
            Permanently block neighbor
        end if
    end if
end for
```

TABLE II. NETWORK PARAMETER USED IN SIMULATION ANALYSIS

Parameters	Values
Sensor Node Operating System	Contiki 2.7
Simulator	Cooja
Routing Protocol	RPL
Radio Medium Model	Unit Disk Graph Medium (UDGM): Distance Loss
Range of Nodes	Rx and Tx: 50 m, Interference: 100m
Mote Type	Tmote Sky
Duty Cycle	ContikiMAC
Size of Deployment Area	100 × 100 m
Number of Nodes	20, 30, 40
Distance Between Neighbors	20m, 30m, 40m
Number of Sinks	1
Number of Attacker Nodes	1 to 20% and 30% of Nodes
Physical Layer	IEEE 802.15.4
MAC Layer	ContikiMAC
MAC Driver	CSMA/CA
Channel Selection	By Default 26
Network Layer	ContikiRPL
Network Driver	Sicslowpan
Transport Layer	UDP
Traffic model	Constant Bit Rate
Simulation Duration	30 min

TP and FN are the number of positive samples (i.e., attacks) that are classified correctly and incorrectly, respectively. FP and TN are the number of negative samples (i.e., normal samples) that are classified incorrectly and correctly, respectively. It should be noted that for networks if the attacker there is not on the network, the FP value is zero. To calculate the detection rate, each scenario is repeated 5 times, and each time the attacker's node position is randomly selected in the simulation. After 5 repetitions, the average results are taken. The results of simulations for neighbor and DIS attacks are shown in Table III.

As can be seen in Table III, the performance of the IDS module is suitable in terms of TPR and FPR. From Table III, it is observed that the proposed model for detection of DIS attack has ideal results, where the TPR is 100% and also the FPR is 0%. Also, the proposed model for the detection of the neighbor attack shows high detection results not long after the attack starting. However, when the simulation runs for a long time, although the TPR is very high and in some cases, it can be 100%, but the FPR increases slightly. This is because the

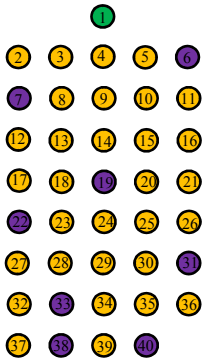

Fig. 6. Network topology (40 nodes) with 20% of the nodes as attacker nodes

TABLE III. THE RESULTS OF SIMULATIONS FOR NEIGHBOR AND DIS ATTACKS

Number of Nodes	Distance (m)	Number of Attackers	TPR (%)		FPR (%)	
			DIS Attack	*Neighbor Attack*	*DIS Attack*	*Neighbor Attack*
40	**40**	***1***	100	100	0	3.66
		20%	100	96.11	0	5.85
		30%	100	96.43	0	0.2
	30	***1***	100	100	0	1.44
		20%	100	99.05	0	1.77
		30%	100	98.71	0	5.86
	20	***1***	100	100	0	0.55
		20%	100	98.06	0	1.26
		30%	100	98.5	0	1.35
30	**40**	***1***	100	100	0	1.17
		20%	100	99.64	0	3.42
		30%	100	98.57	0	9.61
	30	***1***	100	100	0	0.7
		20%	100	99.65	0	1.47
		30%	100	97.38	0	2.95
	20	***1***	100	100	0	1.06
		20%	100	99.3	0	1.75
		30%	100	98.56	0	0.3
20	**40**	***1***	100	100	0	2.73
		20%	100	98.88	0	3.86
		30%	100	94.47	0	2.22
	30	***1***	100	98.88	0	1.3
		20%	100	99.55	0	2.31
		30%	100	99.45	0	5.72
	20	***1***	100	100	0	0.3
		20%	100	98.86	0	1.48
		30%	100	98.74	0	1.41

separation of the neighbor nodes around the tampered nodes would be difficult due to the fact that the initiated attacks alter them as attacker nodes. The simulation results show that the acceptable performance of the proposed IDS in the detection of neighbor and DIS attacks.

The selected TPR from Table III for neighbor and DIS attacks are shown in Fig. 7 and Fig. 8.

The proposed method is compared with IDS provided by Lee et al. [10]. In this comparison, the network is designed according to [10] and the proposed method is evaluated and compared with the values expressed in method [10]. As you can see in Table IV, the FPR for the proposed method is very lower for both the neighbor and DIS attacks. The proposed method is fully distributed and there is no need for clustering. The proposed method also finds more than 1 attacker (20% and 30% of the nodes as the attacker) with very good results, which Lee et al. [10] did not comment on.

VI. Conclusion and Future Works

Security in the IoT is a key, vital, and important topic and has been considered as an important research topic. In this research, we tried to provide an IDS to deal with neighbor and DIS attacks. In the IDS, the placement strategy is fully distributed. The system architecture is stand-alone. The detection method is an anomaly-based lightweight IDS and based on the threshold value for dealing with neighbor and DIS attacks but it can also be used to detect similar attacks. The proposed method is adaptable and in which configuration or specific conditions are not foreseen and can be used in various applications. The simulation results show high TPR with a small percentage of FPR. The proposed model is fully effective in detecting attacks and applicable to large-scale networks.

For future work, due to IDS architecture in Section IV, we can mention the following:

- Developing a proposed model for Detecting other specific internal attacks such as Local Repair attack
- Consider the mobility of nodes in detecting attacks
- Focus on identifying external attacks (the Internet side) and using appropriate firewalls

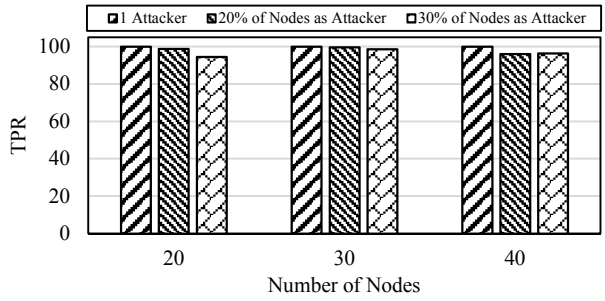

Fig. 7. TPR for detection of Neighbor attack at a 40m distance

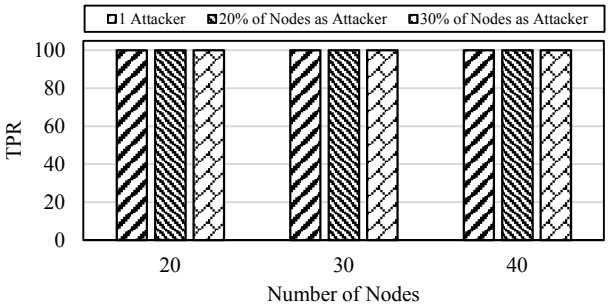

Fig. 8. TPR for detection of DIS attack at a 40m distance

TABLE IV. Comparison of Proposed Method with Lee et al. [10] Method

Type of Attacks	Method	TPR (%)	FPR (%)
Neighbor Attack	*Proposed Method*	100	1.4
	Lee et al. [9]	100	2.64
DIS Attack	*Proposed Method*	100	0
	Lee et al. [9]	100	5.92

References

[1] S. Li, L. Da Xu, and S. Zhao, "The internet of things: a survey," *Information Systems Frontiers,* vol. 17, no. 2, pp. 243-259, 2015.

[2] B. Farzaneh, "Intrusion Detection and Identification of Attacks in the RPL-Based Internet of Things (IoT)," M.Sc. Thesis, Department of Electrical & Computer Engineering, Isfahan University of Technology, Isfahan, Iran, 2018.

[3] F. Hu, *Security and privacy in Internet of things (IoTs): Models, Algorithms, and Implementations*. United Kingdom: CRC Press, pp. 1-586, 2016.

[4] J. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis, "Connecting low-power and lossy networks to the internet," *IEEE Communications Magazine,* vol. 49, no. 4, pp. 1-6, 2011.

[5] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks,* vol. 54, no. 15, pp. 2787-2805, 2010.

[6] T. Winter *et al.*, "RPL: IPv6 routing protocol for low-power and lossy networks," Internet Eng. Task Force (IETF), RFC 6550, 2012.

[7] T. Tsao, R. Alexander, M. Dohler, V. Daza, A. Lozano, and M. Richardson, "A security threat analysis for the routing protocol for low-power and lossy networks (RPLs)," Internet Eng. Task Force (IETF), RFC 7416, 2015.

[8] 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.

[9] M. Zhao, A. Kumar, P. H. J. Chong, and R. Lu, "A comprehensive study of RPL and P2P-RPL routing protocols: Implementation, challenges and opportunities," *Peer-to-Peer Networking Applications,* vol. 10, no. 5, pp. 1232-1256, 2017.

[10] A. Le, J. Loo, K. K. Chai, and M. Aiash, "A specification-based IDS for detecting attacks on RPL-based network topology," *Information,* vol. 7, no. 2, pp. 1-25, 2016.

[11] A. Le, J. Loo, Y. Luo, and A. Lasebae, "Specification-based IDS for securing RPL from topology attacks," in *IFIP Wireless Days (WD)*, pp. 1-3, 2011.

[12] P. Pongle and G. Chavan, "A survey: Attacks on RPL and 6LoWPAN in IoT," in *International Conference on Pervasive Computing (ICPC)*, pp. 1-6, 2015.

[13] 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.

[14] 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*: Springer, pp. 1205-1213, 2014.

[15] 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, 2015.

[16] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *29th Annual IEEE International Conference on Local Computer Networks*, pp. 455-462, 2004.

[17] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *4th International Symposium on Information Processing in Sensor Networks*, pp. 364-369, 2005.