A Trust-Management-Based Intrusion Detection System for Routing Protocol Attacks in Internet of Things

Mahmoud Elbaradie [a], Anas Youssef [b], and Ashraf El-Sisi [b]
[a] Department of Math and Computer Science, Faculty of science, Tanta University
[b] Department of Computer Science, Faculty of Computers and Information, Menofia University
mahmoud.elbarady@science.tanta.edu.eg , anas.youssef@ci.menofia.edu.eg , ashraf.elsisi@ci.menofia.edu.eg

Abstract— The Internet of things is a pool of on-demand and configurable resources and services that are delivered across the usage of the internet. Providing privacy and security to protect their resources is considered a very challenging issue since the distributed architecture of the cloud makes it vulnerable to the intruders. To mitigate this issue, intrusion detection system plays an important role in detecting the attacks in the network. Intrusion detection system is a software or hardware component that implements monitoring and analysis processes of the system events or network activities. Once detecting any intrusion, an alert is raised to the administrator in order to take appropriate actions against such these intrusive events. In this paper an intrusion detection system is proposed for routing protocol for lossy and low power network attacks. The objective of the proposed system is to detect a variety of routing attacks namely sinkhole, selective forward and blackhole attacks. The detection algorithm uses trust management strategies that are based on a set of trust properties each of which is used for the detection of a specific type of routing attacks. The proposed attack detection algorithm was simulated using the Contiki Cooja simulator with centralized intrusion detection system placement strategy. The evaluation results show that in the proposed algorithm was able to detect the simulated attacks with 100% true positive detection rate in some scenarios.

Keywords— Trust Management, Internet of Things; RPL Attacks; Intrusion Detection System; Sinkhole; Blackhole.

I. INTRODUCTION

The Internet of things (IoT) systems are exposed to several attacks because of the huge number of connected devices and the nature of this devices which all operate in low power and lossy networks (LLNs). Traditional security techniques cannot be directly applied to overcome IoT security issues [1]. LLNs use a standard routing protocol named routing protocol for LLNs (RPL). There are three main types of RPL attacks namely: attacks against resources, attacks against topology and attacks against traffic [2]. The focus of this paper is to detect RPL attacks against topology specially sinkhole, selective-forward and blackhole attacks.

In this paper an intrusion detection system (IDS) for RPL attacks in IoT networks is proposed. The proposed approach uses trust management strategies for the detection a specific type of RPL attacks such as sinkhole, selective forward and blackhole attacks. The proposed IDS uses an attack detection algorithm that leverages a set of parameters to detect a variety of RPL attacks. Two main attack detection parameters are used by the detection algorithm which are the node-rank change rate in the network and the drop rate of node-forward-packets in the network.

A wide variety of research techniques have been proposed in the literature to design, implement, and evaluate an IDS for IoT networks [3, 4, 5, 6, 7]. Such techniques can detect and identify various types of RPL attacks such as sinkhole, blackhole and selective forward attacks. Each of these techniques is focused on a single type of attacks, while the proposed approach aims to develop an extendable IDS that can be

easily modified to detect types of attacks other than those used in the trust model that supports the proposed IDS.

Simulated networks with different configurations were implemented to evaluate the proposed IDS using Contiki Cooja simulator [8]. The true positive attack detection percentage ranges from 89% to 100% for sinkhole attack and ranges from 80% to 100% for both blackhole and selective-forward attacks. Also, simulations were performed with a combination of sinkhole and blackhole attacks. In such simulations the true positive attack detection percentage was in range from 92% to 100%. Such combination was implemented by merging two different attack detection parameters in a single trust management model. The proposed technique increases the detection percentage for the mentioned attacks and provides high detection percentage for combination of sinkhole and blackhole attacks.

The paper is organized as follows. Section II introduces background information about related RPL attacks and types of IDSs. Related work is briefly explained in section III. Section IV describes the proposed IDS. A brief explanation of the experimental setup is introduced in section V. The experimental results are presented and discussed in section VI. Finally, conclusions and future work are summarized in section VII.

II. BACKGROUND

Security issues in IoT are categorized based on IoT layers into three layers namely: perception layer, application layer and transportation layer [9]. In this paper the focus is on RPL attacks in the perception layer. The following subsections provide background information about such RPL attacks and a classification of different IDSs.

A. RPL Attacks

Figure 1 shows a taxonomy of attacks against RPL protocol [2]. The attacks are classified into three main categories based on the target of each attack. The first category of attacks is made against network resources in which illegitimate nodes create unnecessary actions to overload these resources. The second category of attacks is made against network topology while the third category attacks is made against the traffic flow in the network. Each of the three categories will be described in more detail as follows.

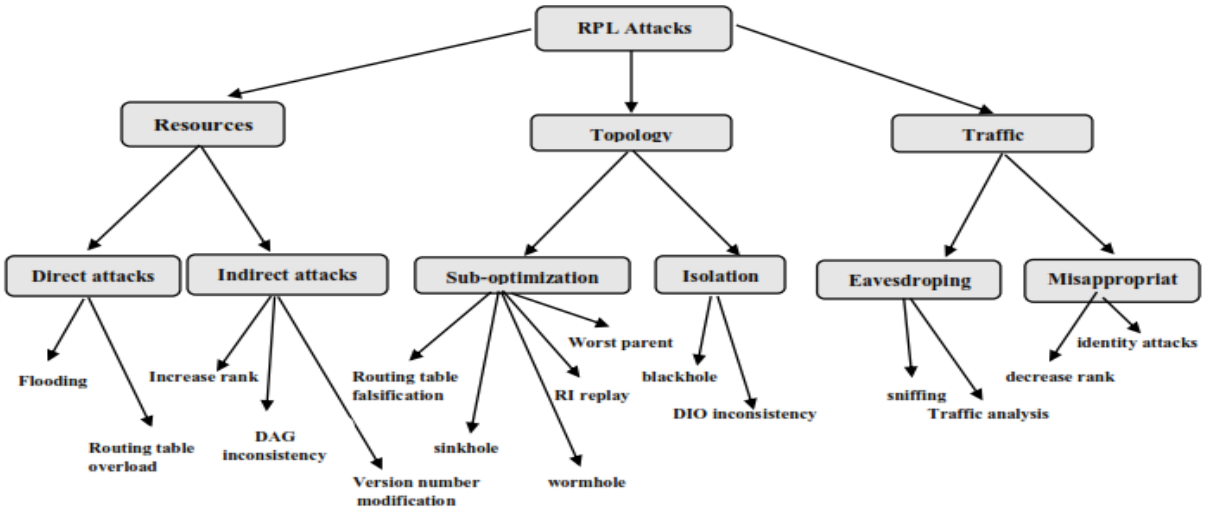

Fig. 1. Taxonomy of RPL attacks [2]

Resource attacks are divided into two subcategories namely: direct attacks and indirect attacks. In direct attacks, the malicious node breaks down the resource by creating a flow of unnecessary actions which include flooding and routing table overload. In the indirect attacks, the malicious node forces other nodes

to create unnecessary actions to overload the network resources. The indirect attacks include increased-rank attacks, direct acyclic graph (DAG) inconsistency attacks and version number modification attacks.

Topology attacks are divided into two subcategories namely: sub-optimization and isolation attacks. The sub-optimization attacks affect the network performance by diverging network traffic paths from the optimal path. They include routing table falsification attacks, sinkhole, wormhole, routing information replay and worst parent attacks. The isolation attacks aim to isolate a node or a subset of nodes in the network which makes these nodes idle and consequently, cannot communicate with their neighbours or with the root. The isolation attacks include blackhole attack and destination advertisement object (DAO) inconsistency attacks.

Topology attacks are the focus of this paper specially sinkhole, selective forward and blackhole attacks. Sinkhole is considered a node rank attack where the malicious node advertises a lower rank than its neighbors to attract their traffic to itself. In selective-forward attack, malicious nodes may refuse to forward certain messages and simply drop them. The malicious node selectively drops the packets coming from a particular node or from a group of nodes [10]. The blackhole attack drops all messages and never forwards anything, this attack when combined with the sinkhole, can have severe effects on the network performance [11].

Finally, traffic attacks include two subcategories namely: eavesdropping attacks and misappropriation. In eavesdropping attacks, malicious nodes perform eavesdropping activities such as sniffing on the traffic of the network. Eavesdropping attacks include sniffing and traffic analysis attacks. Misappropriation attacks work on changing the identity of a legitimate node. These attacks do not cause effective damage of the RPL network. These attacks are often used as a base for other types of attacks. Misappropriation attacks include decreased rank and identity attacks.

B. Intrusion Detection Systems

The IoT network can be secured by encryption but there are some limitations on applying encryption in IoT networks [8]. This is due to the lack of encryption and authentication standards developed specifically for IoT networks in addition to the limited computing power of IoT devices. Therefore, traditional security countermeasures like encryption could not work efficiently in IoT systems [12]. For this reason, developing specific security solutions for IoT networks is essential. Hence, an IDS is adopted with various IDS placement strategies are applied. The administrator of an IoT network can choose the strategy that is more appropriate for the network resource capacity. An IDS acts as a network observer which generates an alert before the attacker start attacking. It can detect both types of attacks namely: internal and external attacks. IDSs can be classified into three categories based on placement strategies, detection methods and validation strategies [12]. More details will be described in the subsequent subsections.

B.1 IDS Placement Strategies

There are three placement strategies based on network architectures [12]. The three strategies are described in more detail as follows:

- **Distributed IDS placement**: An IDS module is placed in every node in the LLN. Each node is responsible for monitoring its neighbors.
- **Centralized IDS placement:** An IDS module is placed in a centralized node, for example, in the border router, where all data that nodes send are gathered in that router. The centralized approach takes into consideration the LLN attribute of IoT nodes. The centralized node is the only node in the network that analyzes data and is responsible for the detection of attacks from any objects in the physical domain.

- **Hybrid IDS placement:** Hybrid IDS placement strategy combines the two previous placement strategies together to gain from the advantages of both and to avoid their draw backs. There are two approaches for hybrid IDS placement. In the first approach the network is divided into clusters where the IDS modules are placed in the head of each cluster. In the second approach, IDS modules are placed both in the centralized nodes and in the other network nodes. The main difference of this approach from the first one is the presence of a central component.

B.2 Intrusion Detection Methods

The intrusion detection methods are divided into four categories namely: anomaly-based, signature-based, specification-based and hybrid [12]. Details will be described as follows:

- **Signature-based approaches:** the IDS generates internal signature databases from the behaviors of the different attacks. If the activity of the network matches the signatures stored in the databases, then an alarm is fired. Signature-based IDSs are very effective and provide high accuracy of detecting attacks with known signatures. However, they are ineffective to detect new attacks and variants of known attacks, because a signature of these attacks is still unknown by the IDS.
- **Anomaly-based approaches:** the IDS studies and analyses the behavior of the network under attack and compares this behavior with the normal network behavior. If a mismatch based on a specified threshold occurs, the IDS fires an alarm. This approach is efficient to detect attacks against resources, because such attacks force a network node to create unnecessary actions which affect the behavior of the node.
- **Specification-based approaches:** the IDS specifies an exact behavior for each node, routing table or other network component. The approach then detects intrusion when a network component behavior deviates from the specified exact behavior with a specific threshold. Therefore, the concept of deviations from normal behavior is the same for anomaly-based and specification-based detection approaches. However, the main difference is the rules of specification-based approaches which are defined manually for each specification made by a human expert which makes this approach both tedious and error prone.
- **Hybrid approaches:** merge the concepts of the three previously described approaches to maximize their advantages and minimize the impact of their drawbacks.

B.3 IDS Validation Strategies

IDS validation strategies can be performed by use of data and experts. While the use of data means a quantitative and more objective validation, the use of experts provides a subjective and qualitative validation. A classification of validation methods was presented in [12]. This classification includes four main validation methods namely: hypothetical, empirical, simulation and theoretical. Hypothetical methods are used when there is unclear relation between the IDS and realism. Empirical methods are used for systematic experimental gathering of data from operational settings. In [13, 14] the authors developed experimental testbeds using a combination of specific IoT software/hardware components such as TinyOS [15] and Raspberry Pi [16] to evaluate their proposals. Simulation methods are used is simulating various IoT scenarios. The authors in [3,4,17] used simulation as their validation strategy. Theoretical methods are based on formal or precise theoretical arguments to support results [18].

III. RELATED WORK

SVELTE is a hybrid IDS where the authors proposed an anomaly- and signature-based IDS [3]. They proposed a hybrid placement strategy where they placed IDS modules both in the 6LoWPAN Border Router (6BR) and in any other constrained IoT nodes in the network. SVELTE IDS applied network simulations

on sinkhole and selective forward attacks using Contiki Cooja network simulator [8]. SVELTE IDS expect that the 6BR is not a constrained node and it can be a PC or a laptop.

SVELTE IDS defines three modules. The first module is the 6Mapper mapping module which is responsible for collecting information about the RPL network. The second module is for intrusion detection which analyses the data collected by the 6Mapper and detects the intrusion. The third module is a firewall which filters unwanted traffic before it enters the constrained IoT network. For sinkhole attacks, SVELTE IDS was able to achieve a maximum true positive attack detection rate of 90% which decreases for larger networks. While for selective forward attacks, SVELTE IDS was able to achieve 80% true positive attack detection rate in lossy networks.

The proposed IDS in this paper aims to detect the same attacks like SVELTE, but using a centralized approach to relieve the load from the burden of the constrained IoT nodes. Unlike SVELTE, the proposed IDS uses a set of trust management properties to detect different types of attacks and the 6BR is constrained node as any node in network not pc or laptop.

INTI is an IDS of sinkhole attacks in 6LoWPAN IoT networks that was proposed in [4]. INTI was proposed to identify sinkhole attacks on the routing services in IoT. INTI aimed to reduce the ratio of false positive and false negative attack detection rates. INTI used reputation and trust strategies for detection of attackers by analysing the behaviour of IoT devices. The results showed that INTI achieved a sinkhole attack detection rate up to 92% in a fixed scenario and 75% in a mobile scenario. The proposed IDS is similar to Cervantes IDS in its usage of reputation and trust strategies for detection of attackers. The difference between the two approaches is that the proposed IDS analyses the behaviour of the attack itself not the behaviour of the IoT devices.

A trust-based IDS was proposed in [5] to secure RPL routing protocol from blackhole attacks. This approach is based on a monitoring mechanism where each node in the network monitors the traffic of its neighbours. The proposed IDS used traffic analysis to detect malicious nodes. This method costs the constrained IoT devices extra energy consumption. The proposed IDs in this paper provides a similar work to secure RPL routing protocol from blackhole attacks in addition to other types of attacks.

SIEWE is an anomaly-based IDS to detect RPL blackhole attack [6]. SIEWE filters out suspect able nodes in the network and then verifies the behaviour of these nodes. SIEWE filters out the nodes that broadcast a comparatively high routing metric and appends their node IDs to a suspect list. The behaviour of the nodes in the suspect list is then analysed and the observations are sent to a border router node. The proposed approach is also used to detect RPL blackhole attacks. However, unlike SIEWE, the proposed IDS is signature-based and the detection mechanism is performed based on the behaviour of the attack not the behaviour of nodes in the network.

Trust based IDS to detect internal attacks in IoT Systems is proposed in [7]. This work focused on three attacks namely: blackhole, sinkhole and wormhole attacks. The proposed IDS in this work was placed in each node. Each node behaves as a monitor node and evaluates the trust score of all its neighbours. The trust scores are evaluated based on three different behaviours namely: honesty, reception of packets only from neighbouring nodes and lack of cooperation. This work is similar to the proposed approach in using trust management techniques and in detecting similar attacks. However, it differs from the proposed IDS in that the detection technique is based on the behaviour of the neighbour nodes not on the attack itself. Furthermore, our proposed IDS is placed in the boarder router not at each node to save energy consumption which the monitoring technique exhaust on constrained IoT devices.

Table 1 shows a summary of the related work versus the proposed IDS. The table shows a simple classification of the work discussed above based on IDS characteristics and RPL attacks. Specifically, for each related work the table shows the IDS placement strategy, intrusion detection method and the RPL attacks used to evaluate the IDS detection method. The characteristics of the proposed IDS is shown in the last row of the table.

Table 1. Summary of related work versus proposed IDS

Reference Paper	IDS Placement Strategy		Intrusion Detection Method		RPL Attacks used in evaluation		
	Distributed	Centralized	Anomaly based	Signature based	Sinkhole attack	Blackhole attack	Selective-forward attack
Raza et al. [3]	X	X	X	X	X		X
Cervantes et al. [4]		X		X	X		
Airehrour et al. [5]	X		X			X	
Patel et al. [6]	X		X			X	
Ambili et al. [7]	X		X		X	X	
Proposed IDS		X		X	X	X	X

IV. PROPOSED IDS

This section describes the proposed IDS. The proposed IDS uses an intrusion detection trust management model to detect RPL attacks against network topology. It works as a network observer that generates an alert when the attackers begin to attack both from inside and from outside of the network. It applies a centralized strategy where the IDS is placed in the broader router as shown in Figure 2. All nodes in the network act as data collection components that send their data to a border router node. The border router node collects data for analysis purposes. The proposed IDS applies a signature-based strategy where there are specific signatures that define the different attacks. If any match happens with the stored patterns/signatures, the IDS fires an alarm.

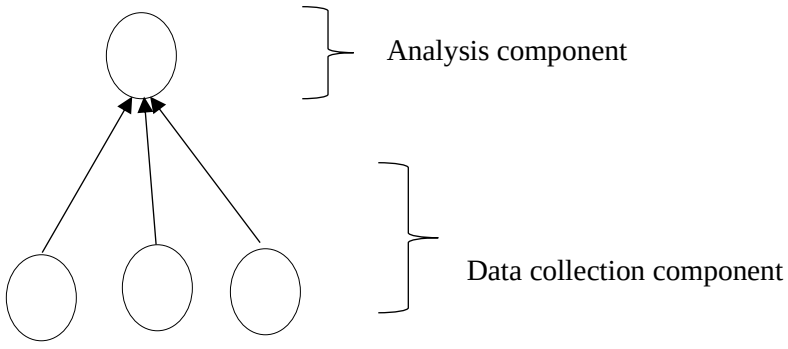

Fig. 2. Centralized Based IDS

A. Proposed Trust Management Model

The proposed IDS is based on a trust management model where the trust assessment of node j by node i at time t is denoted by $T_{ij}(t)$. The time t is the time at which node j sends data to node i . $T_{ij}(t)$ is a real number in the range of [0, 1] where 0 indicates no trust, and 1 indicates full trust. $T_{ij}(t)$ is formally defined as follows:

$$T_{ij}(t) = 1 - D_{ij}(t) \qquad (1)$$

Where $D_{ij}(t)$ is the probability of detecting the attack made by node j to node i at time t. $D_{ij}(t)$ takes a value from 0 to 1 that defines the probability of considering the node to be an attacking node. For example, if $D_{ij}(t)$ is 0.7, this means node i considers node j an attacking node with probability 0.7. Therefore, the trust value $T_{ij}(t)$ will be 0.3. $D_{ij}(t)$ is formally defined as follows:

$$D_{ij}(t) = w_1 \times p_1 + w_2 \times p_2 \quad (2)$$

Where w_1, w_2 are two weights associated with two attack detection parameters, p_1 and p_2 where the sum of w_1 and w_2 is equal to 1. The weight values are used to assign more effect to one of the two attack detection properties on the value of $D_{ij}(t)$ than the other trust property. Changing the weight values provides flexibility in the detection of more than one type of RPL attacks. Each of the two properties are described as follows.

The first attack detection parameter, p_1, is used to detect change-in-node-rank family attacks like sinkhole attack. It is used to calculate the change in rank compared with its neighbours' average rank as shown in the following equation:

$$p_1 = \frac{(\text{average of neighbours ranks} - \text{current node rank})}{\text{MinHopRankIncrease}} \quad (3)$$

Where the MinHopRankIncrease refers to the minimum allowed increase in rank between a node and any of its parents [19]. MinHopRankIncrease is a constant variable whose default value is 256 which is considered the minimum rank in the IoT network, i.e., rank of the root node [20. 21]. The Contiki Cooja simulator which is used in this work assigns a 16-bit rank value to each node. The node rank value changes in units of 256 that allows a maximum of 255 hops. Each node calculates its rank with respect to its parent rank using the summation of the parent rank and the MinHopRankIncrease value. The rank calculation is based on hop count objective function which is defined as follows:

$$R(n) = R(P) + \text{MinHopRankIncrease} \quad (4)$$

where R(n) represents the rank of node n and R(P) is the rank of the parent node. Any node selects one of the neighbour nodes with the least R(n) value to be its parent node.

Figure 3 shows a sample IoT network with sample rank values. This figure shows how the parameter, p_1, is used to detect a sinkhole attack. Root node 1 calculates the change in rank for the other three neighbour nodes, 2, 3 and 4 based on the sent rank from the three nodes. Node 1 then calculates the paramter p1 and $T_{1j}(t)$ for each node j, based on the following steps. First of all, node 1 collects all neighbors' data and calculates the average rank of all neighbor nodes. Then node 1 uses equation (3) to calculate p*1* followed by equation (2) to calculate $D_{1j}(t)$ where $w_1 = 1$, $w_2 = 0$ because only p_1 is used to detect sinkhole attacks, while p_2 is ignored. Finally, trust value $T_{1j}(t)$ is calculated using equation (1). These steps are repeated for each node for each send interval period, t.

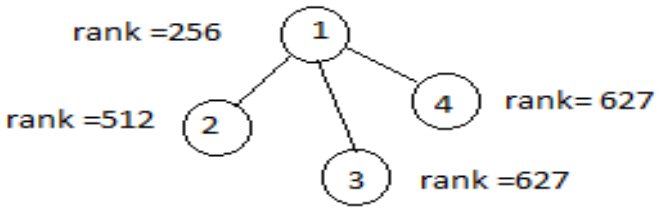

Fig. 3. An IoT network with sample rank values

The values that result from applying the above calculations are as follows. For node 2, $p_{1=}$ 0.1, D_{12} (t) = 0.1 and T_{12} (t) = 0.9. For nodes 3 and 4, p_1 = 0.4 , D_{13} (t) = D_{14} (t) = 0.4 and T_{13} (t) = T_{14} (t) = 0.6. In general, when a sinkhole attack occurs, a decrease in the node rank happens which results in the decrease in the value of T_{ij} (t).

The second attack detection parameter, p_2, is used to detect packet-drop family attacks like selective-forward and blackhole attacks. These attacks work by dropping the forwarded data packet which is received by a node from its neighbours. This data packet is a Destination Oriented Directed Acyclic Graph (DODAG) Information Object (IO) message. This DODAG Information Object (DIO) message stores rank and DAG information about the node's neighbour as shown in figure 4 [22]. DIO message is the first message that each node sends to all its neighbours in the DODAG. The trust property, p_2, is formally defined by following equation:

$$p_2 = 1 - \frac{received\ forwardpacket}{maxdatapacket} \quad (5)$$

where the received forward packet is the summation of the values of all the objects in DIO message like node ID, node rank and number of neighbours as shown in figure 5 [3] and maxdatapacket is the sum of max values of these objects i.e. the sum of max node ID, max rank, max parent ID, max parent rank and max number of neighbours.

The structure of the dropping attacks like blackhole or selective-forward attacks is based on dropping of the received data packet of their children. For example, if a node x, i.e. a blackhole malicious node, receives a data packet from its child node y, the node x neglects the received packet. This means that the received forward packet at the root node becomes always zero.

RPL Instance ID				Version Number	Rank	
G	0	MOP	Prf	DTSN	Flags	Reserved

Fig. 4. DIO message for a node

Node ID	Node Rank	Parent ID	Parent Rank	Number of Neighbors

Fig. 5. Format of the forward data packet

B. Proposed Algorithm

The proposed attack detection algorithm is shown in figure 6. It takes as input a set of nodes N and a threshold value TH. The algorithm calculates and returns a final trust value for all nodes in N and raises an

alarm when the trust value becomes less than or equal to TH. The proposed algorithm performs one of two tasks for all nodes in N at each transmission time. The two tasks are described as follows.

The first task is applied if the node has a direct link with the root node. This task is applied by the proposed algorithm by testing the equality between the node's parent ranks with the root rank as shown in line 3. If this equality is true, the two parameters, p_1 and p_2, are computed as shown in lines 4 and 5.

The second task is applied if the node has no direct link with the root node. This node only sends its data packets to its parent node. In this case the parameter, p_2, is not effective and the detection is only based on parameter p_1. Since there is no forwarding of data packets and consequently, there is no packet dropping that can occur, parameter p_2 is ignored. The weight values, w_1 and w_2, are set based on which of the two parameters has the higher value. The parameter with the higher value is multiplied by the higher weight while the other parameter is multiplied by the lower weight. This is shown in lines 10 to 14. Finally, the root calculates the trust value $T_{ij}(t)$, for each node in N as shown in line 16. The root node then raises an alarm for each node in N if $T_{ij}(t)$ value is less than or equal to TH as shown in lines 18 to 24.

Input: N - a list of nodes and threshold TH (the minimum trust value is equal to 0.5)

Output: trust value for each node in N at the time of sending t, i.e. $T_{ij}(t)$

Begin

```
1.    foreach Node in N do
2.      diff_rank=average_neighbour_rank - node_rank
3.      if  (parent_rank = root_rank)  then
4.          p1← |diff_rank/MinHopRankIncrease|
5.          p2← |1- (received forward packet /maxdatapacket)|
6.      else
7.          p1← |(diff.rank/MinHopRankIncrease)|
8.          p2← 0
9.      end if
10.     If (p1>p2) then
11.         w1 is set to a value greater than w2
12.     else
13.         w2 is set to a value greater than w1
14.      end if
15.    Dij (t) ←w1×p1+w2×p2
16.    Tij (t) ←1- Dij (t)
17.   end for
18.   foreach Node in N do
19.    if  Tij (t) ≤ TH then
20.      Fire alarm
21.      Return Tij (t)
22.    end if
23.    Return Tij (t)
24.   end for
```

End

Fig. 6. Attack Detection Algorithm

Figure 7 describes by an example how the attack detection algorithm shown in figure 6 can detect sinkhole, blackhole and selective-forward attacks. As shown in figure 7, node 1 is the root node, node 4 is a malicious node that generates an attack and the other eight nodes are normal nodes. When the algorithm starts, each of the nodes 2, 3 and 4 sends two packets of data. The first data packet contains the node's own data which contains its rank value and its parent rank value. The second data packet contains its children data. The root node receives all data packets from the three nodes and calculates the two parameters, p_1 and p_2, in addition to the trust values of the three nodes.

As shown in figure 7, node 2 has five neighbors, i.e. nodes 1, 3, 4, 5 and 7. The average of node 2 neighbor ranks is equal to 706.4. According to step 2 in figure 6, the diff_rank for node 2 will be equal to 52.4. The comparison between the parent rank and the root rank is done as shown in line 3 of figure 6 to decide if the node has a direct link with the root node or not, then according to line 4 $p_1 = 0.2047$.

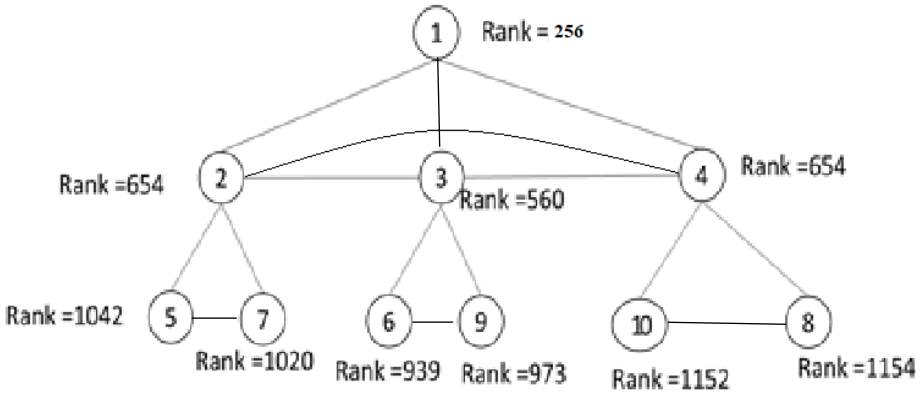

Fig. 7. An IoT network with 9 normal nodes and 1 malicious node (node 2)

The received forward packet from node 5 to node 2 is evaluated to 1705. This is computed by adding node id, node rank, node parent id, node parent rank and number of neighbors for node 5. The maxdatapacket is evaluated to 1822. This value is the maximum received forward packet among all nodes at the root. Afterwards, according to line 5, p_2 is 0.064. The weights values are selected to be 0.8 and 0.2, where the more effective parameter, i.e., with the higher value, is 0.8 and the other parameter is 0.2. Therefore, $w_1 = 0.8$ and $w_2 = 0.2$. Finally, according to equation (1) at line 15, $D_{12}(t) = 0.1766$ and at line 16 $T_{12}(t) = 0.823$. Similarly, the trust values for nodes 3 and 4 are calculated. For nodes 5, 6,7,8,9 and 10 the root calculates only parameter p_1 based on the data forwarded by their parents. Parameter p_2 is set to zero because of all these nodes are leaf nodes with no children.

Suppose that node 2 becomes a malicious node that generates sinkhole attack and suppose that its rank decreases to 500 to attract neighbor node to be their parent. Consequently, $p_1 = 0.806$, $p_2 = 0.1487$, $D_{12}(t) = 0.674$ and $T_{12}(t) = 0.325$.

Suppose that node 2 becomes a malicious node that generates blackhole or selective-forward. This means that node 2 will drop the received forward packet, i.e., the received forward packet =0. Therefore, $p_1 = 0.2047$, $p_2 = 1$, $D_{12}(t) = 0.841$ and $T_{12}(t) = 0.159$.

V. EXPERMANTAL SETUP

The experiments used to evaluate the proposed approach are implemented using Contiki Cooja network simulator [8]. A set of experiments are run to evaluate each of the following attacks: sinkhole, blackhole, selective-forward and a combination of sinkhole and blackhole attacks. The number of nodes used in the experiments are either 8, 16 or 24 nodes. Each single experiment is run five times and every

time the true positive attack detection rate is calculated. The true positive attack detection percentage (TAP) is defined by the following equation.

$$\text{TAP} = \left(\frac{\text{number of true alarms } (attack\ events)}{total\ number\ of\ alarms}\right) \text{x } 100\% \quad (5)$$

The simulation time of each of the five run times is selected to be one of the following: 5, 10, 15 and 20 minutes. Finally, the average TAP is calculated across the five different run times for each simulation time. The selected weights are 0.8 and 0.2 where the attack detection parameter which gives the higher TAP value is multiplied by 0.8 while the other detection parameter is multiplied by 0.2.

Figures 8, 9 and 10 are a set of figures that show the structure of the simulated networks that were built to evaluate sinkhole attacks. The border router, i.e. the root node, is always shown in green colour, the normal nodes are shown in yellow colour and the attack nodes are shown in red colour.

Figures 11, 12 and 13 are a set of figures that that show the structure of the simulated networks that were built to evaluate blackhole and selective-forward attacks. In case of blackhole and selective-forward attacks the structure of the network differs from that of the sinkhole attack network to decrease the adverse effect of lossy IOT network. The former is divided into clusters to reduce the reverse effect of lossy networks by reducing the root child. Which led to reduce the amount of request root that happen at same time. The root node contains the IDS. Any normal node sends a list of data to root. The data sent includes node's rank, node's parent ID, node's parent rank, node's neighbour ID, node's neighbour rank, node's neighbour parent and DIO message which is received from its children. At least the attack node which send their data and drop the data for specific child in selective forward attack and data of all child in blackhole attack.

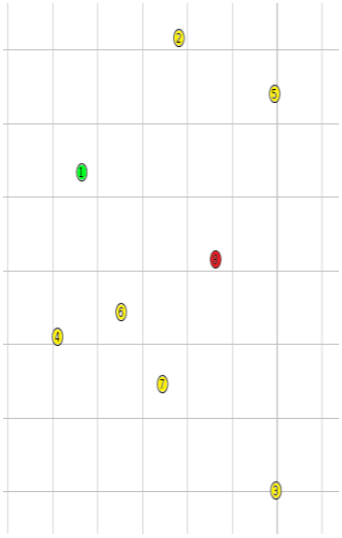

Fig. 8. 8 nodes with 1 sinkhole node attack

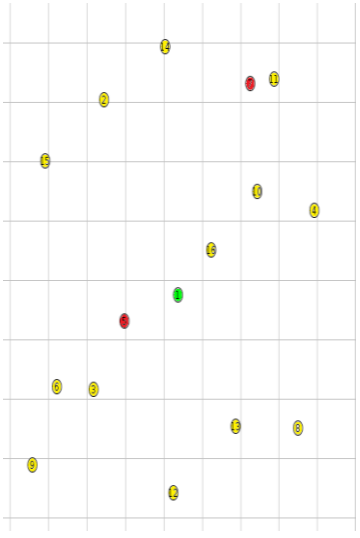

Fig. 9. 16 nodes with 2 sinkhole node attacks

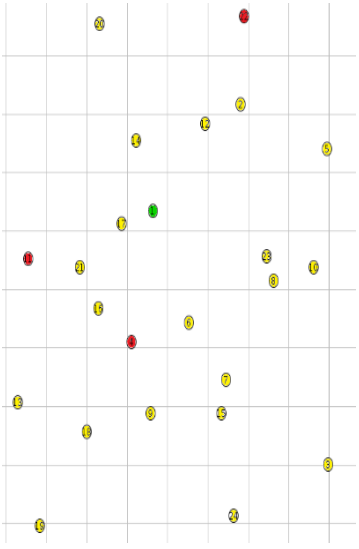

Fig. 10. 24 nodes with 3 sinkhole node attacks

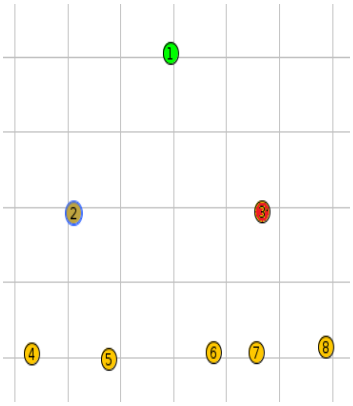

Fig. 11. 8 nodes with 1 blackhole or selective-forward node attack

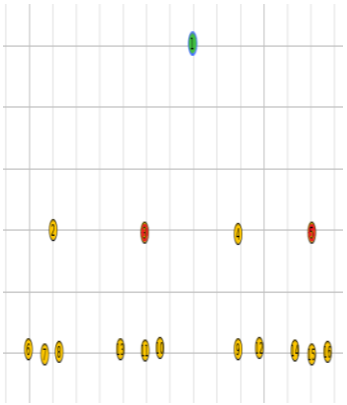

Fig. 12. 16 nodes with 2 blackhole or selective-forward node attacks

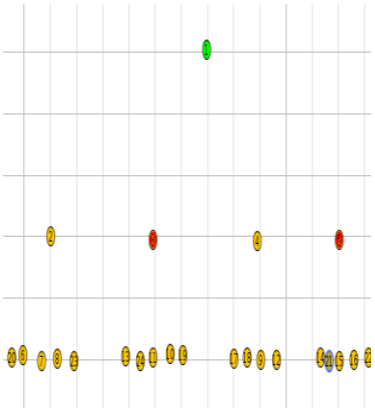

Fig. 13. 24 nodes with 2 blackhole or selective-forward node attacks

VI. EVALUATION RESULTS

In this section evaluation results of the proposed attack detection algorithm are presented and discussed. The attack detection results are shown for sinkhole, blackhole, selective-forward attacks. Furthermore, attack detection results are shown when sinkhole and blackhole attacks are combined in the same node.

A. Sinkhole Attack

This section presents and discusses the attack detection results obtained for sinkhole attack. The results include two variations of the proposed trust management model namely: single weighted and two-weighted trust management models. The single-weighted trust management model is used when only one of the two weights shown in equation (2) is set to one while the other weight is set to zero. This allows only one of the two parameters, p_1 and p_2, in equation (2) to have full effect on the calculated node trust value while the other one does not have any effect at all. The two-weighted trust management model is used when it is required to explore the effect of both parameters as a weighted effect on the calculated node trust value. The higher weight is given to the parameter which provides the higher detection ratio.

A.1 Single-Weighted Trust Management Model

In this subsection we present the attack detection results that are only obtained using the change-in-node-rank attack detection parameter, p_1, without any effect from the packet-drop attack detection parameter, p_2. To obtain such detection results, weight, w_1, is set to one while weight, w_2, is set to zero in equation (2).

Figure 14 shows the average TAP for sinkhole attack using the single-weighted trust management model. As described previously, the experiments were simulated with different run times as shown in the figure. Different IoT network configurations were used in the experiments including 8, 16 and 24 nodes. Figure 14 shows an average TAP of 100% in all simulation scenarios except for the IoT network with 24 nodes. The reason for such finding is that when the number of nodes increases, the lossy property inherited in IoT networks becomes more effective. This clearly affects the average TAP observed in IoT networks with 24 nodes which shows lower average TAP values in the range from 82% to 100%. Also, as the simulation time increases, the lossy property inherited in IoT networks becomes more apparent in IoT networks with large number of nodes. To overcome such limitation, the IoT network can be divided into clusters where the IDS can be distributed across the root node and the head of each cluster. If the head of cluster becomes a malicious node, it can be detected by the root.

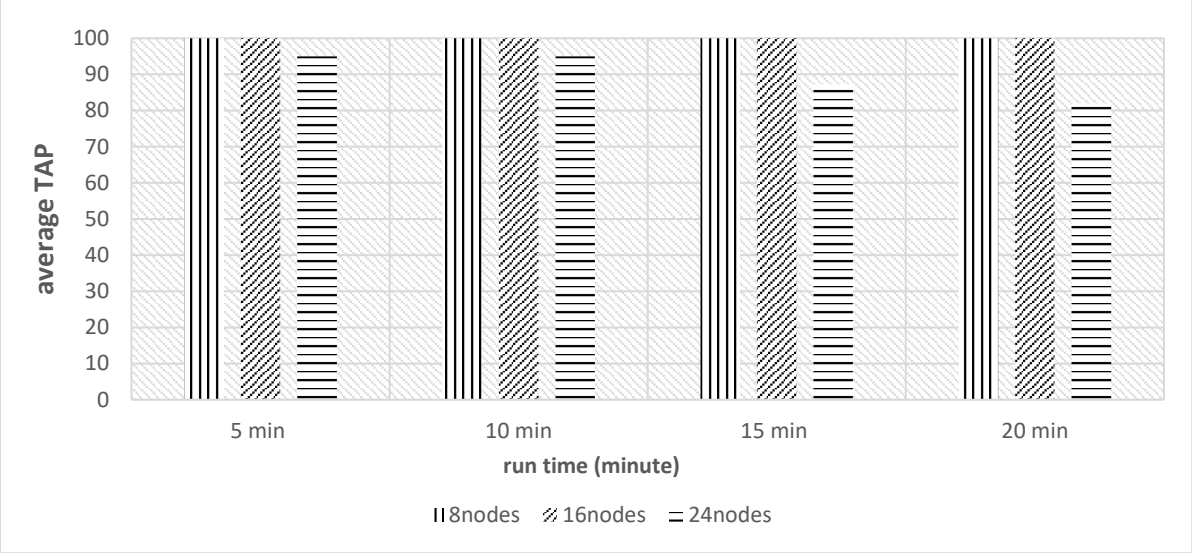

Fig. 14. Average TAP for sinkhole attack using single-weighted trust model

A.2 Two-Weighted Trust Management Model

In this subsection we present the detection results using both the change-in-node-rank attack detection parameter, p_1, and the packet-drop attack detection parameter, p_2. To obtain such detection results, the weights, w_1, and w_2, in equation (2) are set to 0.8 and 0.2 depending on the average TAP values obtained from the two parameters, p_1 and p_2. As described previously, the attack detection parameter which gives the higher TAP value is multiplied by 0.8 while the other detection parameter is multiplied by 0.2.

Figure 15 shows the average TAP for sinkhole attack using the two-weighted trust management model. Like the results illustrated in figure 14, an average TAP of 100% is obtained in all simulation scenarios except for the IoT network with 24 nodes. However, there is slight improvement in the average TAP values in the IoT networks with 24 nodes when compared with their counterparts shown in figure 14. The average TAP values in the IoT networks with 24 nodes range from 89% to 100%. This slight improvement shows the effect of merging the two parameters, p_1 and p_2, together in the trust model.

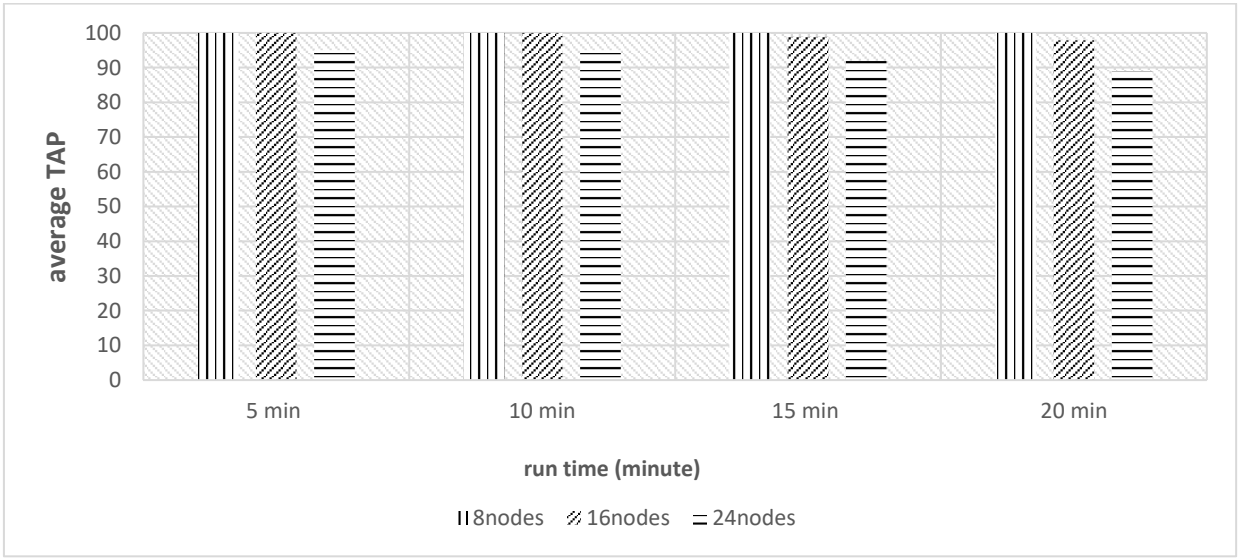

Fig. 15. Average TAP for sinkhole attack using two-weighted trust model

B. Blackhole and Selective-forward Attacks

This section presents and discusses the attack detection results obtained for blackhole and selective-forward attacks. The results include the same two variations of the proposed trust management model like those shown in figures 15 and 16.

B.1 Single-Weighted Trust Management Model

Unlike subsection A.1, in this subsection we present the attack detection results only obtained using the effect of parameter, p_2, without the effect of parameter, p_1. To obtain such detection results, weight, w_1, is set to zero while weight, w_2, is set to one in equation (2).

Figure 16 shows the average TAP for both blackhole and selective-forward attacks using the single-weighted trust management model. The two attacks show the same result because the both blackhole and selective-forward attacks have the same malicious effect of dropping packet data. Like the results obtained in figures 14 and 16 for sinkhole attack, an average TAP value of 100% was obtained in all simulation scenarios except for the IoT network with 24 nodes. However, the average TAP values for the IoT network with 24 nodes are better than their counterparts in figure 14. The average TAP values shown in figure 17 and 18 shows a minimum of 92% across all simulation scenarios. The reason for this is that when the number of nodes increases, the IoT network is divided into clusters as described previously in section V. The flooding of messages forwarded by nodes are managed which allows the lossy property inherited in IoT networks to become less effective.

B.2 Two-Weighted Trust Management Model

Like subsection A.2, in this subsection we present the attack detection results using both parameters, p_1, and p_2. Figures 17 and 18 show the average TAP for blackhole and selective-forward attacks, respectively, using the two-weighted trust management model. Figure 18 shows that there is a slight increase in the average TAP values obtained for the selective-forward attack in the network with 24 nodes when compared with their corresponding values in Figure 17 for the blackhole attack. This is because in selective-forward attack the malicious node drops forward packet to a selected child not all children like the case of blackhole attack.

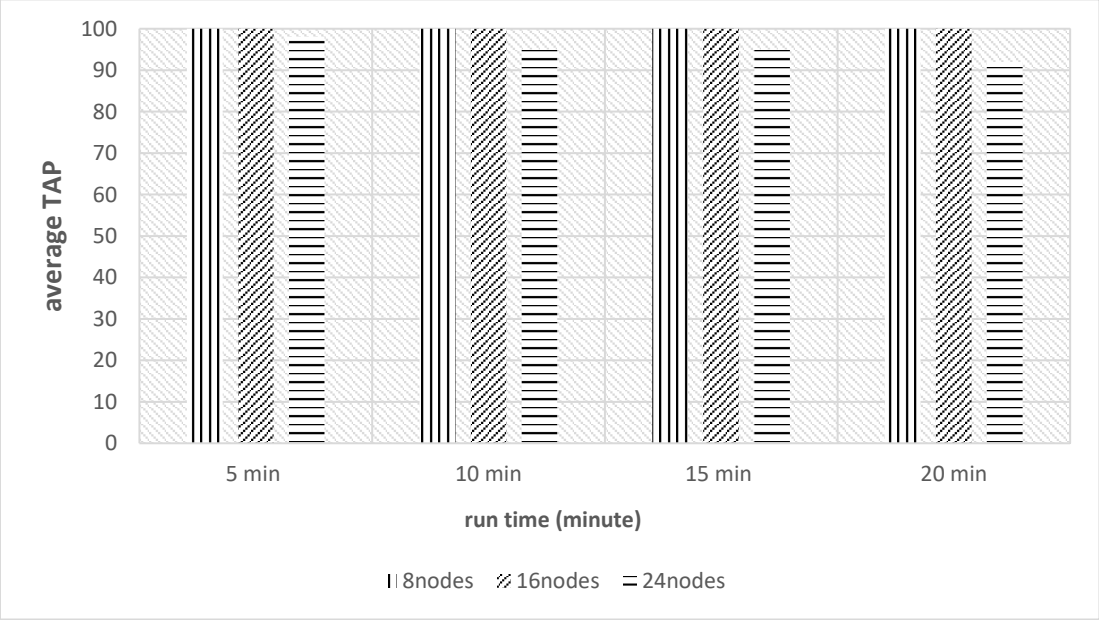

Fig. 16. Average TAP for blackhole/selective-forward attacks using single-weighted trust model

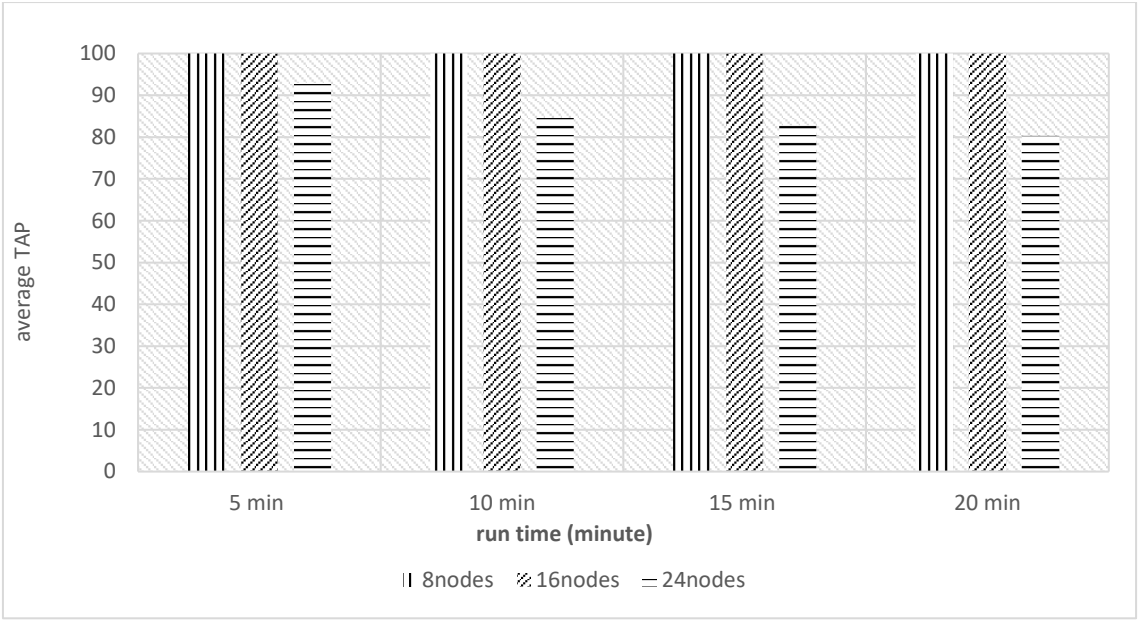

Fig. 17. Average TAP for blackhole attack using two-weighted trust model

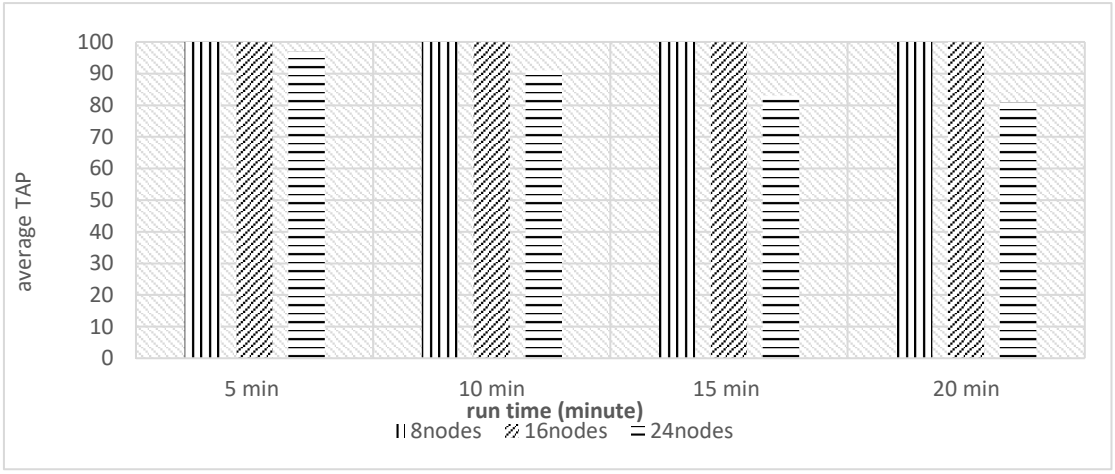

Fig. 18. Average TAP for selective-forward attack using two-weighted trust model

Another notice in figures 17 and 18 is that both figures show the average TAP value decreases in the 24 nodes network with the increase in the run time. The observed decrease happens because the constrained root node misses received data during calculating the two parameters when more nodes send at the same time.

C. Combining Sinkhole and Blackhole Attacks

In this subsection we present the attack detection results that are obtained when a combination of sinkhole and blackhole attacks are applied in the same node. Such type of combinations can have very adverse effects on the performance of the IoT network and harm its performance significantly [23]. This combination will lead to make unexpected change in node ranks in addition to dropping packet data in the network. The proposed trust management model with its two parameters, p_1 and p_2, is very useful in detecting such combination of attacks since each of the two parameters is used to detect only one of the two attacks, i.e., sinkhole and blackhole. To the best of our knowledge, such combination of attacks has not been investigated before while detecting RPL attacks in IoT networks.

Figure 19 shows the average TAP for the combined attacks using the two-weighted trust management model. The figure shows average TAP values that range from 91% to 100%. These results show the positive

effect of the proposed trust management model on the detection accuracy of different types of attacks with different properties when combined in a single attack.

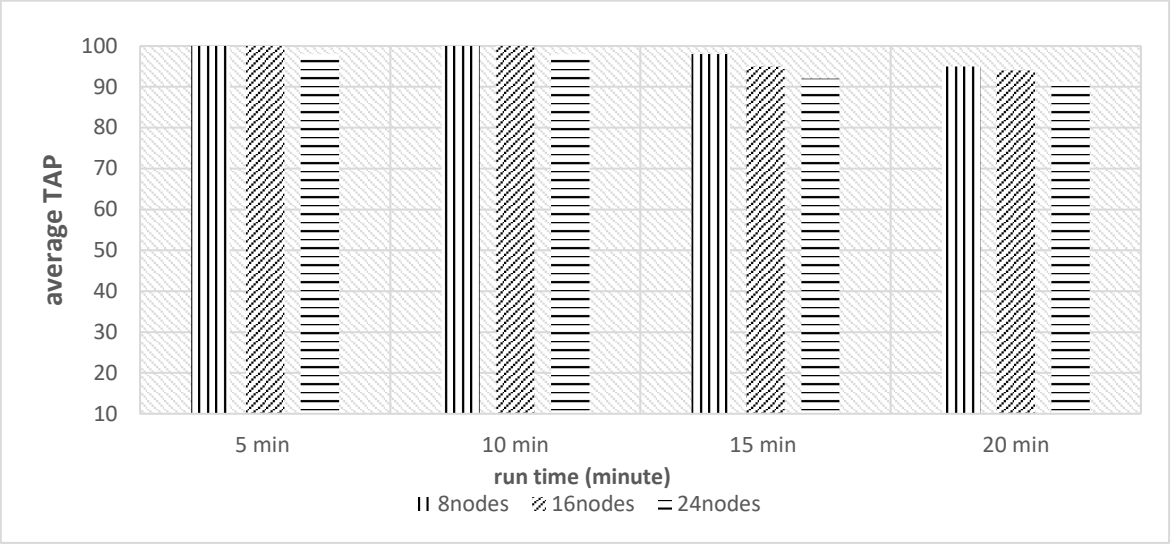

Fig. 19. Average TAP for combined attacks using two-weighted trust model

VII. CONCLUSIONS AND FUTURE WORK

This paper proposed an IDS to detect a set of RPL attacks in IoT networks. The proposed IDS is based on an attack detection algorithm that uses a novel intrusion detection trust management model. The proposed trust management model is based on two attack detection parameters to detect a set of different RPL attacks. The first parameter is used to measure the change of node rank to detect rank family attacks like sinkhole attacks. The second parameter is used to measure the dropped ratio of forward data packets that a network node receives from its parents. The second parameter is effective in the detection of packet drop family attacks like blackhole and selective-forward attacks.

Simulated IoT networks were implemented to evaluate the proposed attack detection algorithm. The obtained average TAP ranges from 82% to 100% for sinkhole attack and ranges from 80% to 100% for both blackhole and selective-forward attacks. Also, simulations were performed with a combination of sinkhole and blackhole attacks. In such simulations the average TAP was in the range from 91% to 100%.

Future work of this research is twofold. The first is testing the proposed IDS with different types of attacks. The second is the extension of the proposed approach to detect a wider range of attacks using additional trust properties.

REFERENCES

[1] M.A. Iqbal, O.G. Olaleye, M.A. Bayoumi, A review on Internet of Things (Iot): security and privacy requirements and the solution approaches, Glob. J. Comput. Sci. Technol. E Network, Web Secure. Vol.16, no.7, 2016.

[2] Isabelle Chrismant, and Remi Badonnel, “A Taxonomy of Attacks in RPL-based Internet of Things,” International Journal of Network Security, IJNS, vol. 18, no.3, pp.459 - 473, 2016.

[3] Raza, Shahid, Linus Wallgren, and Thiemo Voigt. "SVELTE: Real-time intrusion detection in the Internet of Things." Ad hoc networks vol.11, no.8, pp. 2661-2674, 2013.

[4] Cervantes, Christian , Poplade, Diego & Nogueira, Michele & Santos, Aldri,“Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things”. Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management. pp.606-611, 2015.

[5] David Airehrour, Jairo Gutierrez and Sayan Kumar Ray, “Securing RPL routing protocol from blackhole attacks using a trust-based mechanism,” Telecommunication Networks and Applications Conference (ITNAC), 2016 26th International, pp. 115-120, 2016.

[6] Patel, Himanshu B., and Devesh C. Jinwala. "Blackhole detection in 6LoWPAN based internet of things: an anomaly-based approach." TENCON 2019-2019 IEEE Region 10 Conference (TENCON). IEEE, pp. 947-954,2019.

[7] Ambili, K. N., and Jimmy Jose. "Trust Based Intrusion Detection System to Detect Insider Attacks in IoT Systems." Information Science and Applications. Springer, Singapore, pp.631-638, 2020.

[8] Mehmood, Tayyab. "Cooja network simulator: Exploring the infinite possible ways to compute the performance metrics of iot based smart devices to understand the working of iot based compression & routing protocols." pp.1-7, 2017.

[9] Chen, Ray, Fenye Bao, and Jia Guo. "Trust-based service management for social internet of things systems." IEEE transactions on dependable and secure computing vol.13, no.6, pp. 684-696, 2015.

[10] Wazir Zada Khan et.al “Comprehensive Study of Selective Forwarding Attack in Wireless Sensor Networks”in I.J. Computer Network and Information Security, vol. 1, pp. 1-10,2011.

[11] 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.

[12] Bruno Bogaz Zarpelão , odrigo Sanches Mianiand and Cláudio Toshio Kawakani, “A Survey: Intrusion detection system for internet of things .” Journal of Network and Computer Applications, vol. 84, pp. 25–37, 2017.

[13] Kasinathan, P., Costamagna, G., Khaleel, H., Pastrone, C., Spirito, M.A. 2013b. DEMO: an IDS framework for internet of things empowered by 6LoWPAN. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security,CCS '13, ACM, New York, NY, USA, pp. 1337–1340,2013.

[14] Amaral, J., Oliveira, L., Rodrigues, J., Han, G., Shu, L. Policy and network-based intrusion detection system for IPv6-enabled wireless sensor networks. In: Communications (ICC), 2014 IEEE International Conference on, pp. 1796–1801, 2014.

[15] Levis P. et al. “TinyOS: An Operating System for Sensor Networks.” In: Weber W., Rabaey J.M., Aarts E. (eds) Ambient Intelligence. Springer, Berlin, Heidelberg, pp 115-148, 2005.

[16] Maksimovic, Mirjana & Vujovic, Vladimir & Davidović, Nikola & Milosevic, Vladimir & Perisic, Branko. “Raspberry Pi as Internet of Things hardware: Performances and Constraints.” 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), vol.3, no.8 , pp. 1013–1018, 2014.

[17] Pongle, Pavan, and Gurunath Chavan,“Real Time Intrusion and Wormhole Attack Detection in Internet of Things”, International Journal of Computer Applications (0975 - 8887), Volume 121 - No. 9, 2015

[18] Kasinathan, Prabhakaran, et al. “Denial-of-Service detection in 6LoWPAN based Internet of Things.”, 9th international conference on wireless and mobile computing, networking and communications (WiMob) (pp. 600-607), 2013.

[19] Pongle, Pavan, and Gurunath Chavan,“Real Time Intrusion and Wormhole Attack Detection in Internet of Things”, International Journal of Computer Applications, vol. 121, no. 9,2015

[20] Sousa, Natanael, et al. "ERAOF: A new RPL protocol objective function for Internet of Things applications." 2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), IEEE, pp.1-5, 2017.

[21] Winter, T.; Thubert, P.; Brandt, A.; Hui, J.; Kelsey, R.; Levis, P.; Pister, K.; Struik, R.; Vasseur, J.P.; Alexander, R. RPL: IPv6 routing protocol for low-power and lossy networks. No. RFC 6550, 2012. Available online: https://tools.ietf.org/html/rfc6550/ (accessed on 10 October 2018).

[22] N. Djedjig, D. Tandjaoui, F. Medjek and I. Romdhani, "New trust metric for the RPL routing protocol," 2017 8th International Conference on Information and Communication Systems (ICICS), Irbid, pp. 328-335, 2017.

[23] 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.