JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 35, 361-374 (2019)
DOI: 10.6688/JISE.201903_35(2).0007

IMBF – Counteracting Denial-of-Sleep Attacks in 6LowPAN Based Internet of Things

A. JAHIR HUSAIN[1] AND M. A. MALUK MOHAMED[2]
[1]*Software System Group*
[2]*Department of Computer Science and Engineering*
M.A.M. College of Engineering
Tiruchirappalli, 621105 India
E-mail: ssg_jahir@mamce.org; malukmd@gmail.com

Many efforts have been taken by the Internet Engineering Task Force to fill the gap between the Internet of Things and the real life, by means of the 6LoWPAN protocol. The 6LoWPAN protocol allows a huge number of smart objects to be connected using the large address space of IPv6. IoT becomes successful by 6LoWPAN and the new Routing Protocol RPL. However, these two protocols are less secured and very much vulnerable to Denial of Service attacks. In this paper we design, implement and evaluate a novel intrusion detection and prevent mechanism called IMBF to secure the IoT from Denial of Sleep Attack. The proposed mechanism utilizes a lightweight modularized system to detect and prevent the Denial of Sleep Attack by means of Malicious Node Alert messages. The simulation results establish that the proposed technique is successful in detecting and preventing Denial of Sleep attacks in IoT.

Keywords: 6LoWPAN, denial-of-service attacks, Internet of Things, intrusion prevention system, IPv6, RPL, sleep deprivation attack, malicious node alert message

1. INTRODUCTION

The advent of the Internet of Things (IoT) paradigm is a most remarkable phenomenon of the last decade. The IoT is a network of different types of objects or things formed using the standard Internet Protocols. It is expected that a significant number of new types of things such as home appliances, vehicles, medical equipments, industry apparatus, traffic control signals and many more will be connected dynamically to the Internet. The development of different communication protocols, along with the miniaturization of transceivers, transforms an isolated, independent device into a communicating thing connected with the Internet. Furthermore, computing power, storage capacity and energy efficiency of small sensing or computing devices have considerably enriched while their sizes have hugely decreased. These kinds of developments in electronics and computer science have originated the increase in the number of Internet-connected smart devices in terms of millions, which can provide a variety of services to the human beings [1].

The growing attractiveness of the IoT is established by its application in different areas such as the development of smart buildings, smart cities, the energy resources management and networks, smart home applications, transportation, logistics, mobility management, and healthcare systems *etc.* [2]. Attaining these goals has been explored, up to date, by various research groups. Five such well-known research groups are Inter-

Received August 26, 2017; revised May 22 & July 19, 2018; accepted August 18, 2018.
Communicated by Meng Chang Chen.

net of Things (IoT), Mobile Computing (MC), Pervasive Computing (PC), Wireless Sensor Networks (WSNs), and, very recently Cyber-Physical Systems (CPS) [3].

Three reference models of IoT have been widely discussed in the literature. They are: (1) Three-layer model, (2) Five-layer model and (3) Seven-layer model [1]. The three-layer model shows the IoT as an extension of wireless sensor networks (WSN) [4]. In this model, IoT is considered as the combination of Wireless Sensor Network (WSN) and cloud computing services. In the five-layer model, the complex system of an enterprise is subdivided into several services and therefore service management and service composition are given in two different middle level layers [5]. The seven-layer model suggested by CISCO is a comprehensive extension to the traditional three and five layer models [6].

In all the three reference models, WSN and the communication technologies are considered to be the building blocks of IoT. Smart sensors in WSN are used to observe the environmental conditions and the 6LoWPAN protocol is applied to communicate the sensed data to other devices. 6LoWPAN is the acronym of IPv6 over Low power Wireless Personal Area Network, a protocol created by the Internet Engineering Task Force in order to apply the Internet Protocol to the smallest devices compatible with IEEE 802.15.4 based wireless nodes [7]. In IoT, tiny sensor and actuator networks are connected to the Internet by integrating the two different protocols namely IP and 802.15.4, and thus create the combination of issues from both the networks. Because of the resource constrained devices and the new protocols the security of IoT becomes a big challenge [8]. The existing security protocols used in IPv6 are very heavy and energy consuming, while 802.15.4 does not provide any security solutions for IP communications.

Out of the numerous attacks focused on 6LoWPAN, Denial of Service (DoS) attacks targeting the accessibility of the network, make it busy or engaged for an indefinite period of time. The purpose of these classes of attacks is not just to eavesdrop or to alter the data but to degrade the performance or crumple the network by some sort of physical attacks [9]. A special kind of DoS attack is the Denial of Sleep (DS) attack whose aim is to decrease the availability of network resources by reducing the time, the nodes spent in sleep mode. There are three well-known types of DS attacks against the IoT edge devices: They are (i) Battery draining, (ii) Sleep Deprivation (SD), and (iii) Outage attacks [1]. The DS Attacks may happen either from insider malicious node or an adversary from the Internet side. Therefore a cryptography system alone cannot protect IoT from this attack. A malicious node can compromise some genuine nodes, learn the decryption mechanism from them and make damages. Therefore an Intrusion Detection System (IDS) is very much needed to watch the nodes activities and to alert the genuine nodes when the signs of attack are spotted.

Furthermore, energy management also remains a main issue in IoT, since the IoT nodes have to work with the limited battery power. IoT edge devices consume battery power for three major processes: (i) Computation, (ii) Communication, and (iii) Dormant or sleep mode [10]. The inspiration for the proposed research work is to detect and prevent the Sleep Deprivation attacks in 6LoWPAN based IoT networks, before network operations are interrupted by any malicious activity and to initiate the proper remedy to increase network availability. In this paper, we propose a system for detecting the intrusion attack which causes the IoT nodes suffering from denial of sleep torture and preventing the nodes from the attacks for extending the durability of the network.

Objectives: The main objectives of the suggested IMBF are as follows:

(i) To investigate the neighbor information in order to maintain the set of genuine nodes.
(ii) To alert the genuine nodes about the presence of the malicious node by providing the malicious node ID.
(iii) To broadcast an alert message to the other nodes in order to prevent the network from the malicious attack.

2. RELATED WORKS

A sleep deprivation attack is a dangerous attack in IoT, since changing or recharging the batteries is not possible in sensor fields. In this type of attack, the invader tries to send an undesired set of requests that seem to be legitimate. The attacker compels the nodes to be active for a longer period of time for spending their energy in useless jobs and thus they will be put in power lost condition and become selfish [11]. According to Stajano and Anderson, three types of energy drainage attacks are predicted to exhaust the battery [12]: (i) Malignant attack – application code or kernel binary is modified to increase power consumption (ii) Benign attack – the device is forced to execute a legitimate but energy consuming application without modifying the application; (iii) Service Request attack – the nodes are repeatedly requested to forward the packets over the network or to find the route *etc.*, A dynamic coding mechanism is proposed by Amin [13], which implements a distributed signature based IDS which uses a bloom filter for signature matching. In DEMO, Kasinathan *et al.* described a Network based DoS detection IDS architecture in project Ebbits [14]. Ebbits system operates a module to monitor the network traffic in order to analyze and detect misbehaving nodes. Although Ebbits detects DoS attacks in 6LoWPAN networks, mobility of nodes is not considered when detecting attacks. In the finite state machine based IDS system, [15] Le *et al.* suggested a distributed IDS system, which is capable of detecting rank based attacks and local repair attack for IP based WSNs. In SVELTE [16] Raza, *et al.* proposed a hybrid IDS system for IoT. The authors presented a Host based IDS, which constructs network topology at 6BR system to detect the attacks like Sinkhole, DODAG inconsistency, Rank and selective forwarding. Jun *et al.* [17] proposed an event based IDS to overcome the problems of real-time of IDS in IoT. They have designed an IDS architecture on the basis of Event Processing Model (EPM). It is a rule-based IDS in which rules are stored in Rule Pattern Repository. This approach consumes more CPU resources, and memory to degrade the network performance. In the Intrusion Detection System (IDS) framework for IoT empowered by 6LowPAN devices [18] Kasinathan *et al.* studied the various approaches for detecting denial-of-service (DOS) attacks, which can be applied to collapse a network. Since 6LoWPAN employs low bandwidth, it is very much sensitive of DOS attacks. In Real time intrusion and wormhole attack detection system [19] Pongle *et al.* presented a method, in which the worm hole attack is detected by using the location information and neighbor information of a node and the attacker node is identified by Received Signal Strength (RSS). The Comparison of strengths and weakness of the existing and the proposed approach is given in Table 1.

Table 1. Comparison of existing and proposed approaches.

Sl. No.	Approach	Strengths	Weakness
1	Specification Based IDS for RPL (IP Based WSNs) [15]	Finite state machine design to detect RPL based attack.	Low false rate alarm and it is suitable only for IP based WSN
2	SVELTE (IDS for IoT) [16]	Host based IDS	Consumes more memory in every node
3	RIDES (IP-Based WSNs) [26]	Bloom Filters, CUSUM Charts	Computational complexity is high
4	Novel Hybrid IDS (WSNs) [27]	Clustered approach to save Energy	Not suitable for 6LoWPAN based IoT
5	Energy Efficient Hybrid-IDS (WSNs) [28]	Cluster Based, Energy Efficient	Considers intrusion from cluster head side only
6	Proposed method	(i) It improves energy efficiency and network scalability by using sectorization and clusterization approach. (ii) Three level module based approach can efficiently detect SD attack by reducing the probability of false positives	Transmission of control packets and message overhead may become high in some cases

3. PROPOSED MODEL

In this work, we have proposed an "Improved Module Based Framework" (IMBF) for detecting and preventing Denial of Sleep (DS) attacks in 6LoWPAN IoT system. IMBF observes the dormant period of sensors in the 6LoWPAN network and prevents them from any attack which makes them to become selfish in order to save their battery power. We have developed a system for energy preservation of sensor nodes and to alert the gateway node through a Malicious Node Alert (MNA) message. Generally, each sensor node separately reads and directs the field data to the sink or head node, after that it goes into the dormant state during its sleeping time, and then it wakes up, senses the field data and communicates. This process is repeated until an external stimulus is occurred to alter the process. Energy saving and the extension of the IoT lifetime are achieved by both reducing the number of transmissions and also by keeping the nodes in idle mode during their sleeping periods. Sensor nodes are not depending on any sleeping commands from the sink node (cluster/head); but, the choice of active or sleep mode is selected by each node locally. Therefore, at the time of selecting the choice, two aspects have to be considered. First one is calculating the sleeping time (ST_i) of each node i, and second is the time of acting as relay node when the network is in multihop ad hoc mode. This means that the sleeping period of a sensor node must be taken into account by neighbor nodes, before forwarding their messages. In other words, when determining its sleeping time, each node must consider the consequences of the sleeping time on the performance of the ad hoc mode operation. Therefore in our proposed model, each sensor node (S_i) will broadcast the Sleeping Time (ST_i) with the node ID, and the field data, so that other nodes will be able to fix their routes when they send their data in the ad hoc network.

3.1 Architecture

The proposed model of IMBF is shown in Fig. 1. It encompasses the sensor network, functioning with 6LoWPAN protocol. The sensor network is divided into a probable number of sectors. Each sector is controlled by a Sector Head (SH) and all the SH nodes are supervised by a Cluster Head (CH). The entire network is connected to the Internet via a 6LBR router.

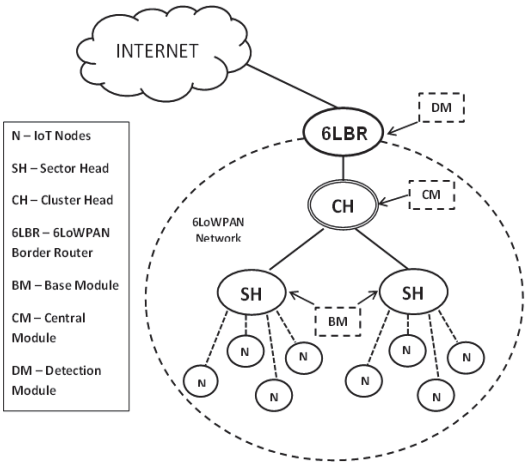

Fig. 1. Proposed IMBF intrusion prevention system for 6LoWPAN based IoT.

The IMBF system consists of three modules: The first one is the Base Module (BM), the second module is the Central Module (CM) and the third module is the Detection Module (DM). The BM is positioned in SH nodes, CM is placed in CH nodes and the DM is employed in 6LBR node. In the process of identifying the 'sleep deprivation torture', all the three modules work together to efficiently detect the anomalies and prevent the network from SD attacks. We have assumed the following key-points for the better understanding of the system:

(i) The topology is considered as a static one.
(ii) The location details of all the sensor nodes are known at the time of deployment.
(iii) The neighbor nodes are identified based on the Received Signal Strength Indicator (RSSI) value [20]
(iv) There is no attack at the time of initialization.
(v) Each node is assigned a unique ID.
(vi) Every sensor node (*N*) has its own duty cycle [21] and the same is advertised at the time of bootstrapping.
(vii) SH nodes have the knowledge of the duty cycle of the nodes.

BM and CM are working together to observe the abnormalities in the network. BM is working as the accumulator and distributor of vital data to the higher modules (CM and DM) in the process of detecting SD attack torture. The role of CM is to notice any abnormalities happening in the network and forward the same to the DM.

3.2 Distribution of Vital Data by Base Module

Some fundamental information are exchanged between IoT nodes (*N*) and SH node, at the time when 6LoWPAN is initiated. The information exchanged are (i) Node ID of every node (ii) The preliminary neighbor node IDs based on the RSSI value (iii) Duration of 'active' and 'dormant' period of the nodes as defined in the duty cycle. These data are delivered to the CM through a control message. Once the network is established the nodes will start reading the environment and forward the data to the SH nodes. The role of the SH node is to collect the data from the sensor nodes, scrutinize the data to avoid redundancy and forward the pure data to the CH node. The BM module at the SH nodes gathers the above mentioned vital information and forward them to the CM module at the CH nodes. The logical functioning architecture of BM and CM is depicted in Fig. 2. The process of identifying neighbor nodes is given in Algorithm 1. The CH node is responsible for computation over the data. The computation process varies for different applications.

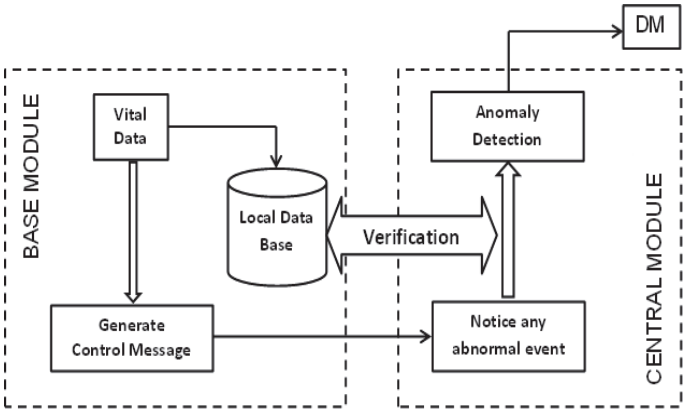

Fig. 2. Logical functioning architecture of BM and CM.

Algorithm 1 // Detection of neighbors

```
{
 For every node N do
 {
 Wait until network initiated
 After network initiated
 Receive the node_id and the duty
 cycle of other nodes
 If hop_count = 1 then
 {
 store node_id as neighbor node
 store the duty cycle with node_id
 }
 Else
 {
 Reject the data
 }
 Forward the stored data to SH
 }
}
```

Algorithm 2 // Detection of neighbor change

```
{

 For every active period do
 {
 Check if there is any change in neighbors
 If "Yes" then
 {
 Change in neighbor found
 Send nbr_chng (neighbor change) to CH
 }
 }
}
}
```

3.3 Forwarding Anomaly Detection Message by Central Module

Whenever an intruder attacks, an abnormality happens in the network. The attacker whose target is to drain the energy of a node, starts sending flood of data like continuous Route Request Packet (RREQ) or worthless control traffic. This induces the victim node to lose its sleep cycles. Hence the node becomes totally exhausted and stops working. In this condition, there will be a change in the neighbor ID or an alteration in the duty schedule or both. It is the responsibility of the CM at the CH node to detect this anomaly and send it to the DM. Algorithm 2 presents the process of detecting neighbor change.

3.4 Detection Module (DM)

In our model DM plays a crucial role in spotting and preventing "Denial of sleep torture" attack. This module has the following responsibilities:

(i) Genuine Neighborhood Authentication (GNA)
(ii) Duty Cycle Supervision (DCS) [21, 22]
(iii) Malicious Discovery (MD)
(iv) Generating and sending Malicious Node Alert (MNA)

The functional view of DM is depicted in Fig. 3. The working principles of these functions are given in Algorithms 3, 4, 5 and 6.

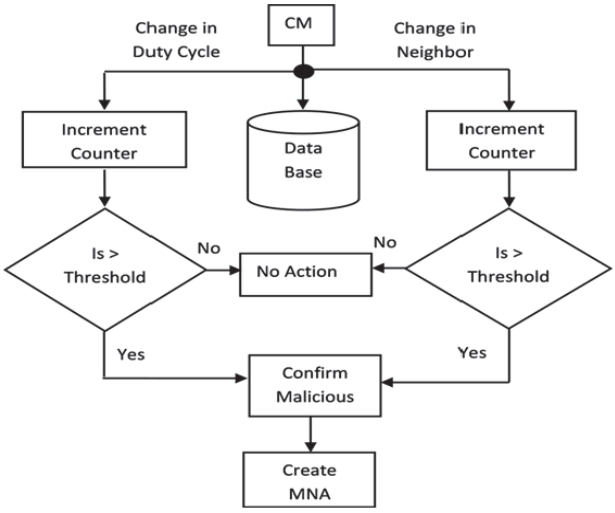

Fig. 3. The functional view of DM.

3.4.1 Genuine neighborhood authentication

In this part, the neighbor information collected from all sensor nodes (N) are investigated based on the distance between the N node and that neighbor. The distance is cal-

culated by the transmission range of every node by using RSSI value [20]. If it is found that the distance is more than the expected transmission range of nodes then the neighbor node is alleged as Malicious Node and a malicious counter (mal_cnt) is initiated for that neighbor. However, the distance measure is not only the required parameter for determining a node as an adversary node. Our intention is to detect and prevent the "denial of sleep torture". We require some more factors to decide the rival activity of a node.

Algorith 3 //Genuine Neighbor Authentication

```
{
Set NID_th=m;
Set nbr_change (node_id)=0
Set mal_cnt=0;
for every active period do
 {
 Calculate the distance between nodes
 // (using RSSI)
 If distance of node is greater than the
 assumed distance then
 {
 suspect as malicious node
 nbr_chng(node_id)=nbr_chng(node_id)+1;
 }
 else
 {
 confirm the neighbor as genuine
 send the nbr_confm packet to SH
 }
 If nbr_chang(node_id) >m then
 mal_cnt=mal_cnt+1;
 }
}
```

Algorithm 4 // Supervising Duty Cycle

```
{
Set DC_th=n;
Set dc_chng (node_id)=0;
For every active period do
 {
 If change in duty cycle (node_id) then
 {
 Suspect as malicious node;
 dc_chng (node_id)=dc_chng(node_id)+1;
 }
 If dc_chng(node_id)>n then
 mal_cnt=mal_cnt+1
 }
}
```

Algorithm 5 // Malicious Node Alert Message

```
{
 //Creating alert message
 Destination address = address of the SH
 nodes
 Affected node ID= Node_ID of the node
 whose duty cycle is altered
 Discovered Malicious Node_ID = New
 neighbor Node ID
 Send the message to the CH and SH nodes
}
```

Algorithm 6 //Malicious discovery

```
{
Set mal_cnt_threshold=k;
 //(k varies for application domain)
If mal_cnt>k then
 {
 node_id(affected) = T;
 // confirm the node is attacked
 create alert message:
 }
}
```

3.4.2 Duty cycle supervision

Every node has its own duty cycle and it is already publicized by the individual node at the time of initialization [21]. Hence, the CH nodes are already having the awareness of awake and sleep timings of the individual nodes. Observing the duty cycle of every node is one of the responsibilities of the CM, so that any alteration in the duty cycle can be predicted as a faulty activity due to an attack.

3.4.3 Malicious discovery

Malicious activity is confirmed by the following two parameters: (i) The change of neighbor ID and (ii) The alteration in the duty cycle of the nodes. We set threshold values for both the parameters: NID_th for the neighbor ID and DC_th for the duty cycle changes. In an IoT environment, the allowable changes in the neighbor ID are very less in a particular period of time. Similarly, alteration in duty cycle is also a very uncommon phenomenon. This is because the active and dormant period of sensor nodes are predefined at the time of initialization. The change is allowed only in rare cases. Every time the malicious counting variable mal_cnt is checked against the malicious count threshold. The threshold value is chosen based on the context of the IoT environment. If it is taken as a small value then the accuracy of detection will be higher whereas if the threshold is big then the accuracy will be less. If the malicious count reaches the threshold, then DM confirms that there is an SD attack at that particular node. This finding is immediately disclosed and an MNA message is sent back to the CH and SH nodes.

3.4.4 Malicious Node Alert (MNA) message

MNA serves a dual purpose in the 6LoWPAN environment: (i) Warning the SH node about the existence of malicious activity and (ii) Instruct the nodes to stop forwarding the packets from the victim node.

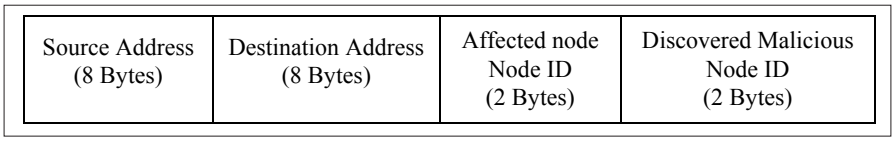

Fig. 4. MNA message format.

The MNA packet format is shown in Fig. 4. On seeing the node ID of the affected node all the other nodes get alerted about the malicious activity and they automatically stop forwarding the packets from the affected node. Hence the nodes will keep up their original duty cycle and hence the power drainage is stopped. This extends the lifespan of the nodes in the IoT environment.

4. EXPERIMENTAL SETUP

This section describes the performance analysis of the proposed IMBF system. The experiment setup is created in Contiki OS with the network simulator cooja. Contiki is an Open Source Operating System, which can be used to connect tiny, inexpensive, light weight microcontrollers to the Internet. For the simulation purpose, we used the emulated Tmote Sky nodes. The configuration setup is given in Table 2.

We run each experiment in a network of 5, 10, 20 and 50 emulated Tmote sky nodes. The following metrics are used to study the performance of our IMBF system.

Table 2. Contiki experimental setup.

Sl. No	Contiki Layer Configuration	Protocol/Interface
1	Radio Interface	CC2420
2	Radio Duty Cycling (RDC)	ContikiMAC
3	MAC	CSMA
4	Network	IPv6
5	Routing	RPL
6	Transport	UDP
7	Physical	IEEE 802.15.4

4.1 Number of Attacks Detected

This parameter is the measurement of actual number of detected attacks against the total number of attacks present, with different configuration of nodes like 5, 10, 20 and 50. The result is shown in Fig. 5 and it is illustrated that IMBF is performing well for all configurations.

4.2 IDS Energy Overhead

Since the nodes are generally battery powered In an IoT environment, and energy is a scarce resource, it is necessary to cut down the consumption of energy in every activity. The powertrace utility of Contiki is used to measure the power consumption by our IDS system [23]. The typical operating conditions of the Tmote sky are shown in Table 3. We calculate the energy consumption using the nominal values. We use 3V in our calculations. The state in which the Micro-Controller Unit (MCU) is idle while the radio is off is referred to as Low Power Mode or LPM. The CPU time is the state where the MCU is on and the radio is off. The states in which the MCU is on and the radio is receiving and transmitting is referred to as listen and transmit respectively.

Table 3. Typical operating conditions of Tmote sky nodes [24].

Sl. No.	Status	Min	Nom	Max	UNIT
1	Supply Voltage	2.1	–	3.6	V
2	Current Consumption: MCU on, Radio RX	–	21.8	23	mA
3	Current Consumption: MCU on, Radio TX	–	19.5	21	mA
4	Current Consumption: MCU on, Radio off	–	1800	2400	μA
5	Current Consumption: MCU idle, Radio off	–	54.5	1200	μA
6	Current Consumption: MCU standby	–	5.1	21.0	μA

We measure the energy consumption of IMBF running for 30 minutes. Fig. 6 shows the network-wide energy usage for 30 min by all the nodes. The network energy usage

and the node power consumption are calculated using the formula [25]:

$$\text{Energy (mJ)} = (\text{transmit} * 19.5\text{mA} + \text{listen} * 21.8\text{mA} + \text{CPU} * 1.8\text{mA} + \text{LPM} * 0.0545\text{mA}) * 3\text{V} / 4096 * 8 \quad (1)$$

$$\text{Power Consumption (mW)} = \text{Energy usage (mJ)} / \text{Time (s)} \quad (2)$$

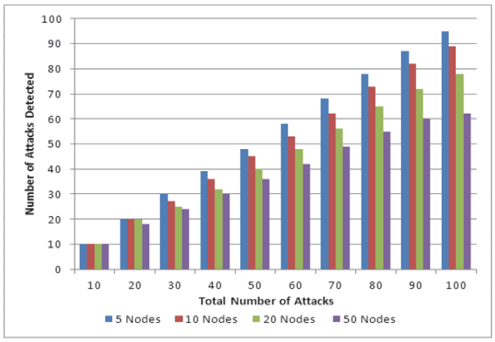

Fig. 5. Total numbers of attacks and the actual detected attacks.

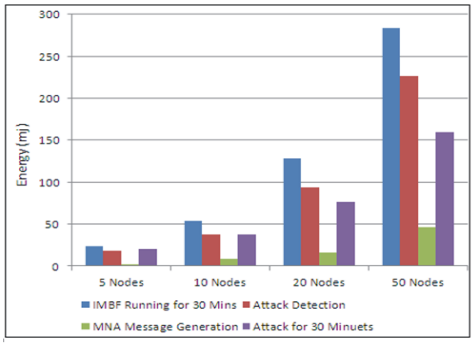

Fig. 6. Energy overhead.

4.3 Success and False Positive Rate

Fig. 7 shows the success and the false positive rate of the proposed IMBF approach. The formula to calculate these two behaviors are:

$$\text{Success rate or True Positive Rate (TPR)} = TP/(TP+FN) \quad (3)$$

$$\text{False Positive rate (FPR)} = FP/(FP+TN) \quad (4)$$

4.4 Memory Consumption

The memory usage by the proposed system is shown in Fig. 8. It shows the memory consumption of the system in KBs for the process of detection of attacks and the MNA message generation.

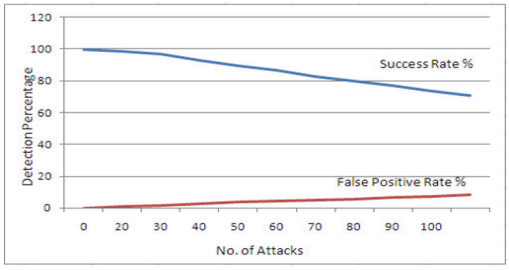

Fig. 7. The success and the false positive rate.

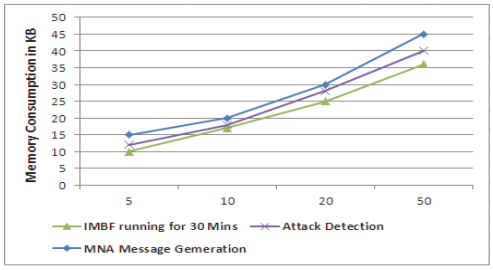

Fig. 8. Memory consumption by IMBF.

4.5 Delay and Delivery Ratio

Fig. 9 shows the results of average packet delivery ratio for the misbehaving attackers 5, 10, 15, 20, 25 nodes and Fig. 10 shows the average end-to-end delay with the network configuration of 10, 20 and 50 nodes.

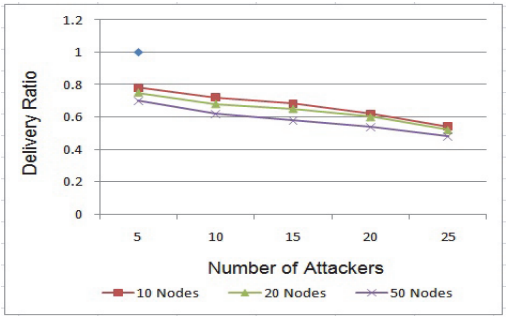

Fig. 9. Attackers vs delivery ratio.

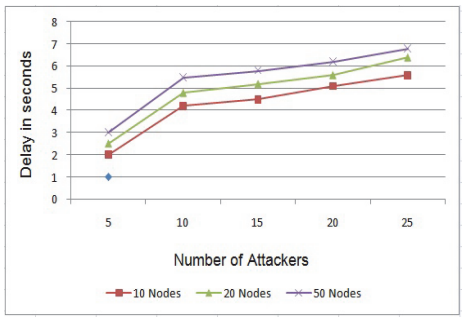

Fig. 10. Attackers vs delay in seconds.

The simulation result proves that the energy consumption is reasonably reduced since the sensing field is partitioned into sectors and clusters.

5. CONCLUSION AND FUTURE WORK

Since IoT has important and great applications in the real life environment, security is main concern while designing the network. In this work, we have designed a mechanism for detecting and preventing "Sleep Deprivation Torture" attacks. The module based approach is best suitable for resource-constrained nodes which run on 6LoWPAN protocol. This method consumes less amount of memory and has better success rates for detecting the attacks. The system has a wide range of flexibility in detecting and preventing SD attacks independent of the size of the network. In the future, we will implement and evaluate the performance of our proposed system for different real time applications.

REFERENCES

1. A. Mosenia and N. K. Jha, "A comprehensive study of security of internet-of-things," *IEEE Transactions on Emerging Topics in Computing*, Vol. 5, 2017, pp. 586-602.
2. M. Swan, "Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self-2.0," *Journal of Sensor and Actuator Networks*, Vol. 1, 2012, pp. 217-253.
3. J. A. Stankovic, "Research directions for the internet of things," *IEEE Internet of Things Journal*, Vol. 1, 2014, pp. 3-9.
4. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, Vol. 29, 2013, pp. 1645-1660.

5. L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, Vol. 54, 2010, pp. 2787-2805.
6. "The Internet of Things reference model," CISCO, http://cdn.iotwf.com/resources/71/IoT Reference Model White Paper June 4 2014.pdf.
7. G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, "Transmission of IPv6 packets over IEEE 802.15. 4 networks," No. RFC 4944, 2007.
8. R. Alexander, T. Tsao, M. Dohler, V. Daza, A. Lozano, and M. Richardson, "A security threat analysis for the routing protocol for low-power and lossy networks (RPLs)," No. RFC 7416, 2015.
9. A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, Vol. 35, 2002, pp. 54-62.
10. M. Pirretti, S. Zhu, N. Vijaykrishnan, P. McDaniel, M. Kandemir, and R. Brooks, "The sleep deprivation attack in sensor networks: Analysis and methods of defense", *International Journal of Distributed Sensor Networks*, Vol. 2, 2006, pp. 267-287.
11. T. Martin, M. Hsiao, D. Ha, and J. Krishnaswami, "Denial-of-service attacks on battery-powered mobile computers," in *Proceedings of IEEE 2nd Conference on Pervasive Computing and Communications*, 2004, pp. 309-318.
12. F. Stajano and R. J. Anderson "The resurrecting duckling," in *Proceedings of the 7th International Workshop on Security Protocols*, 2000, pp. 172-194.
13. S. O. Amin, M. S. Siddiqui, C. S. Hong, and J. Choe, "A novel coding scheme to implement signature based IDS in IP based sensor networks," in *Proceedings of IFIP/IEEE International Symposium on Integrated Network Management Workshops*, 2009, pp. 269-274.
14. P. Kasinathan, G. Costamagna, H. Khaleel, C. Pastrone, and M. A. Spirito, "DEMO: An IDS framework for internet of things empowered by 6LoWPAN," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, 2009, pp. 1337-1340.
15. A. Le, J. Loo, Y. Luo, and A. Lasebae, "Specification-based IDS for securing RPL from topology attacks," *IEEE/IFIP Wireless Days*, 2011, pp. 1-3.
16. S. Raza, L. Wallgren, and T. Voigt, "SVELTE: Real-time intrusion detection in the Internet of Things," *Ad Hoc Networks*, Vol. 11, 2013, pp. 2661-2674.
17. C. Jun and C. Chi, "Design of complex event-processing IDS in internet of things," in *Proceedings of IEEE 6th International Conference on Measuring Technology and Mechatronics Automation*, 2014, pp. 226-229.
18. P. Kasinathan, C. Pastrone, M. A. Spirito, and M. Vinkovits, "Denial-of-service detection in 6LoWPAN based Internet of Things," in *Proceedings of IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications*, 2013, pp. 600-607.
19. P. Pongle and G. Chavan, "Real time intrusion and wormhole attack detection in internet of things," *International Journal of Computer Applications*, Vol. 121, 2015, pp. 1-9.
20. J. Xu, W. Liu, F. Lang, Y. Zhang, and C. Wang, "Distance measurement model based on RSSI in WSN," *Wireless Sensor Network*, Vol. 2, 2010, pp. 606-611.
21. F. Wang and J. Liu, "Duty-cycle-aware broadcast in wireless sensor networks," in *Proceedings of IEEE 28th Conference on Computer Communications*, 2009, pp. 468-476.

22. D. C. Harrison, W. K. Seah, and R. Rayudu, "Rare event detection and propagation in wireless sensor networks," *ACM Computing Surveys*, Vol. 48, 2016, p. 58.
23. A. Dunkels, J. Eriksson, N. Finne, and N. Tsiftes, "Powertrace: Network-level power profiling for low-power wireless networks," SICS Technical Report No. T2011: 05, 2011.
24. http://www.eecs.harvard.edu/~konrad/projects/shimmer/references/tmote-sky-datasheet.pdf
25. H. Lamaazi, N. Benamar, and A. J. Jara, "RPL-based networks in static and mobile environment: a performance assessment analysis," *Journal of King Saud University-Computer and Information Sciences*, Vol. 30, 2017, pp. 320-333.
26. S. O. Amin, M. S. Siddiqui, C. S. Hong, and S. Lee, "Rides: Robust intrusion detection system for ip-based ubiquitous sensor networks," *Sensors*, Vol. 9, 2009, pp. 3447-3468.
27. H. Sedjelmaci and M. Feham, "Novel, hybrid intrusion detection system for clustered wireless sensor network," *International Journal of Network Security and Its Applications*, Vol. 3, 2011, pp. 1-14.
28. A. Abduvaliyev, S. Lee, and Y.-K. Lee, "Energy efficient hybrid intrusion detection system for wireless sensor networks," in *Proceedings of International Conference on Electronics and Information Engineering*, Vol. 2, 2010, pp. V2-25-V2-29.

A. Jahir Husain is a Research Scholar in Software System Group of M.A.M. College of Engineering, pursuing his Ph.D. in Information and Communication Engineering at Anna University, Chennai, Tamil Nadu, India. His research interests include distributed computing, mobile computing, mobile ad hoc networks and currently, his research focuses on intrusion detection in Internet of Things.

M. A. Maluk Mohamed received his Ph.D. degree from Indian Institute of Technology, Chennai at 2006. He is presently working as a Professor in Department of Computer Science and Engineering. His research interests include distributed computing, grid computing, wireless sensor networks, mobile computing and cluster computing. He is a member of IEEE, ACM, ISTE, CSI and IARCS.