International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 1, February 2020, pp. 467~476
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i1.pp467-476

Multiple intrusion detection in RPL based networks

Manjula C Belavagi, Balachandra Muniyal
Department of Information and Communication Technology, Manipal Institute of Technology,
Manipal Academy of Higher Education, Manipal, India

Article Info

Article history:

Received Mar 28, 2019
Revised Jul 3, 2019
Accepted Aug 29, 2019

Keywords:

Intrusion detection
Machine learning
Network security
RPL / 6LoWPAN networks
Wireless sensor networks

ABSTRACT

Routing Protocol for Low Power and Lossy Networks based networks consists of large number of tiny sensor nodes with limited resources. These nodes are directly connected to the Internet through the border router. Hence these nodes are susceptible to different types of attacks. The possible attacks are rank attack, selective forwarding, worm hole and Denial of service attack. These attacks can be effectively identified by intrusion detection system model. The paper focuses on identification of multiple intrusions by considering the network size as 10, 40 and 100 nodes and adding 10%, 20% and 30% of malicious nodes to the considered network. Experiments are simulated using Cooja simulator on Contiki operating system. Behavior of the network is observed based on the percentage of inconsistency achieved, energy consumption, accuracy and false positive rate. Experimental results show that multiple intrusions can be detected effectively by machine learning techniques.

Corresponding Author:

Balachandra Muniyal,
Department of Information & Communication Technology, Manipal Institute of Technology,
Manipal Academy of Higher Education,
Manipal, Karnataka, India 576104.
Email: bala.chandra@manipal.edu

1. INTRODUCTION

Advancement in wireless technology has lead to the development of small, low cost nodes having less power with sensing capability called sensor nodes. Which are used to monitor environmental and physical conditions. They are self-organized. Wireless Sensor Network (WSN) is made up of these nodes. WSN technology is fast developing and is an important research area too. Nowadays these nodes are part of the Internet of Things. Interconnection of billions of such nodes forms an Internet of Things. Available IPv4 address space is not sufficient to handle this. Hence IPv6 over Low-power Wireless Personal Area Network (6LoWPAN) [1, 2] is used for network of these nodes. Traditional routing protocols are not suitable for such networks. Hence the new routing protocol, Routing Protocol for Low Power and Lossy Networks (RPL) is standardized for such networks.

Routing protocol used in IPv6 for resource constrained for power and lossy Networks is RPL [3]. Linked state protocols which are normally used are not suitable for such scenarios as they require a significant amount of memory links state database. Hence RPL is designed as a proactive, distance vector, routing protocol. It starts identifying the routes immediately after the initialization of the RPL network. Topology of RPL based network is not fixed. It forms a tree-like topology, DODAG. Which shows preferred parent of each node each node of the RPL network, which acts as a gateway for that node. If a node’s entry is missing in its routing table for a packet, then the node simply forwards it to its preferred parent. This process continues until it reaches the destination or a common parent.

Applications of RPL based networks include military, health, business, environment surveillance and various other important applications such as the Internet of Things [4, 5]. These nodes are deployed in insecure environments. Hence they are susceptible to different security threats such as Rank attack, Sybil attack, Denial of Service (DoS) attacks, routing attacks, selective forwarding attack, etc [6-8]. Due to resource constraints available security mechanisms such as security protocols, Key-management protocols and authentication techniques are suitable for WSNs [9, 10]. Also the methods used by wireless and wired networks cannot be applied to WSNs [11, 12]. Hence WSNs requires a sophisticated Intrusion Detection System (IDS) which suits its properties.

Over recent years several articles available on IDS for wireless sensor networks [13, 14]. The papers refer to the WSN which is not exposed directly to the Internet. Whereas nowadays these sensor nodes are part of Internet of Things. Hence these nodes are globally available through the Internet via IPV6 BorderRouter (IPv6BR). So there is a requirement of more advanced IDS which can identify multiple attacks. These attacks can be detected effectively using machine learning algorithms [15].

Wallgren *et al* [16] and Le *et al* [17] have proven that RPL network is vulnerable to multiple attacks. In order to handle these attacks in RPL based networks anomaly-based IDS proposed by Raza *et al* [18]. This works in two phases. Initially, IDS data collected later it is analyzed. In the initial phase, the DODAG's root collects information about all its members and their neighbors. That is parent node, all neighbors and their corresponding ranks and the node's rank and a timestamp. Based on this information monitoring node verifies the rank of each node to identify any rank inconsistency to identify the malicious activity in the network. However, the behavior of the network will change with an increase in the number of malicious nodes. Hence there is a need to identify multiple attacks for varied network size.

RPL based networks DODAG construction is based on two objective functions, Expected Number of Transmissions (ETX) and Objective Function 0 (OF0) which works based on hop count [19]. From the literature, it can be identified that performance of ETX based 6LowPAN network is better compared to OF0 [20, 21]. Shreenivas *et al* [22] proposed ETX based IDS. But the number of nodes considered for the experimentation using simulation is very less. Existing literature focuses on a fixed number of malicious nodes with small network size. Multiple attacks have not been taken care of. This paper focuses on the following:

a. Simulate the RPL network using Cooja simulator on the Contiki Operating System using ETX as an objective function.
b. Identification of multiple attacks by scaling up the network size with the percentage increase in the malicious nodes
c. Comparison of network behavior for the above-mentioned network setup based on different parameters

The overall architecture of the RPL based network is shown in Figure 1. The router called IPv6BR is connected to the outside world. IDS module used is centralized and distributed called hybrid and is stored in IPv6BR as well as in individual nodes. DODAG root collects information about the RPL network such as DODAG ID, RPL Instance ID, the DODAG Version Number, the parent node ID, all neighbors and their corresponding ranks, and the node's rank and a timestamp. IDS module analyzes this data to identify the intrusions.

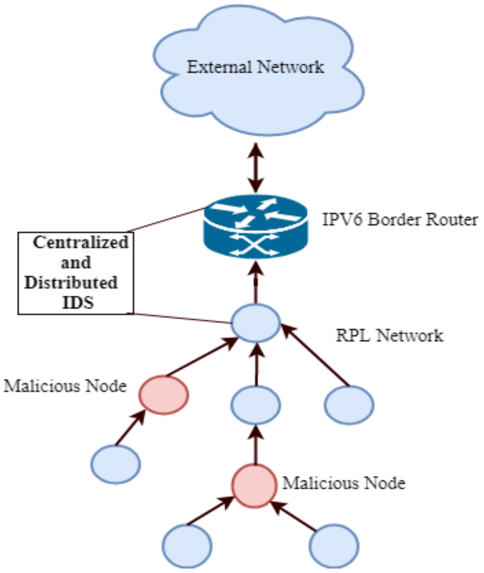


Figure 1. Overall architecture of IDS in RPL based networks

Rest of the paper is organized as follows. Section 2 describes the DODAG construction. In Section 3 research method used for intrusion detection in RPL based networks is discussed. Section 4 discusses results and analysis of RPL based network for different scenarios and Section 5 concludes the paper.

2. DODAG CONSTRUCTION

In this section RPL based network topology building process is discussed. RPL is a distance vector source routing protocol used by low power and lossy networks RPL networks do not have predefined topology. Figure 3 shows tree-like topology with parent and leaf nodes created by RPL based network. For the construction tree-like topology, RPL uses ICMPV6 control messages namely DODAG Advertisement Object (DAO), Information Object (DIO) and DODAG Information Solicitation (DIS). In RPL network topology (DODAG) has a single root node (R1) with no outgoing edges as shown in Figure 2. It is called as IPv6BR in 6LOWPAN networks. In a DODAG a rank is associated with each node. This value represents the position of the node in DODAG according to root. Rank of a node is always increasing in the downward direction, from root to leaf nodes as shown in Figure 3. Objective functions used for the construction of DODAG are:

a. Objective Function 0 (OF0)
b. Expected Transmission Count (ETX)
c. User-defined objective functions

It is identified by DODAGID and RPLSequenceID. Figure 3 shows DODAG and the rank associated with each node. The transmission path is selected based on the constructed DODAG.

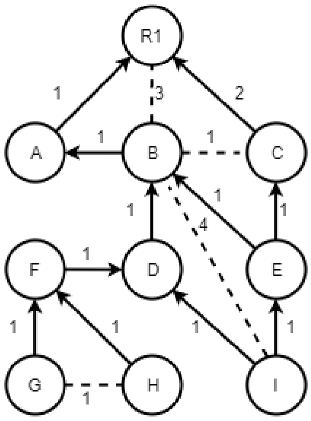


Figure 2. Destination oriented directed acyclic graph (DODAG) – construction

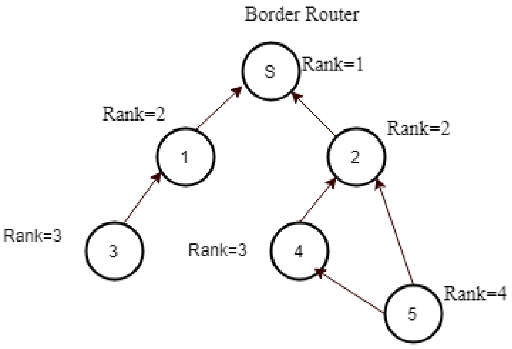


Figure 3. Destination oriented directed acyclic graph (DODAG)

Wireless sensor network traffic is simulated using Cooja [23] simulator on the Contiki operating system [24, 25]. Cooja is built using Java. Which makes use of 'C' programming to code the node. Simulation is stored as XML file with the extension CSC. All the simulation environment information such as types of nodes, the position of nodes, radio medium etc., for the network traffic is recorded. Communications between the nodes are observed using the following plugins:

a. Simulation Visualizer
b. Timeline
c. Radio logger

Using Cooja simulator RPL based network, 6LoWPAN is simulated, the DODAG for that appears as shown in Figure 3.

3. RESEARCH METHOD

The wireless sensor network with malicious activity is simulated to identify the network intrusions. The simulated network is analyzed to identify the intrusions using the packet transmissions, and IDS overhead is measured in terms of the power consumption and percentage of Inconsistencies Rate. Figure 4 shows the overall framework of the proposed method. Initially, the network is configured based on the parameters as shown in Table 1. Simulation is started by selecting the server node, client nodes and the speed limit. Some malicious nodes are also introduced in the network. Intrusions are identified based on the real-time communication between the nodes. Based on every node's parent and neighbor information mapper namely 6LoWPAN Mapper (IPV6Mapper) reconstructs the RPL DODAG. This is done by sending mapping requests in-terms of request packets to every node of at regular intervals. The request packet contains DODAG Identification number, DODAG version number, timestamp and RPL instance Identification number. Packet size of the mapping request is of 5 bytes. In response to the mapping request, each node sends the parent node, all neighbor nodes identification numbers and their ranks to the mapper.

In the case of RPL-based 6LoWPAN networks, the attacker sends wrong rank information about itself and of its neighbors using compromised nodes to the IPV6 Mapper. It is also possible to get inconsistent or incorrect information due to the lossy nature of the network. Hence it is necessary to detect the inconsistencies and to correct the invalid information.

Table 1. Parameters used for the simulation of WSN

Parameters Used	Value
Packet Reception Ratio	100 %
Transmission Ratio	100 %
Transmission Range	50 m
Interference Range	55 m
Simulation Time	20 minutes for each packet reception ratio
Client Nodes	10 – 50
DIO Min	12ms
DIO Doublings	8 ms
RPL MOP	NO-DOWNWARD-ROUTE
Objective Function	ETX

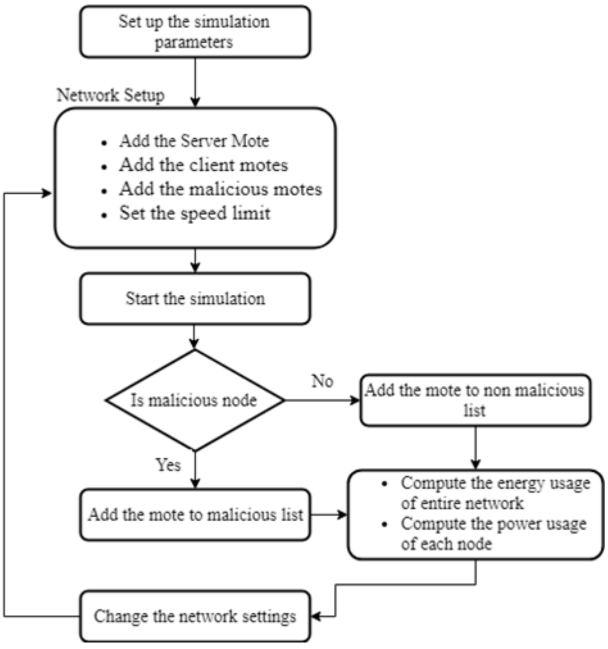

Set up the simulation parameters
Network Setup
Add the Server Mote
Add the client motes
Add the malicious motes
Set the speed limit
Start the simulation
Is malicious node
No
Add the mote to non malicious list
Yes
Add the mote to malicious list
Compute the energy usage of entire network
Compute the power usage of each node
Change the network settings

Figure 4. Workflow of intrusion detection in wireless sensor networks

Algorithm 1 DODAG Construction

```
procedure Build-DODAG (Nodes N)
    Initially Root sends DIO message.
    for all Nodes which are reachable from the Root Node
        Receives DIO message
        These nodes joins the network
        Selects Root as their preferred parent
    for all Nodes N do
```

```
8:              Send DIO message
9:              Nodes which are reachable from that Node receives DIO message and joins the network
10:             if The Timer of a node which is part of the network expires then
11:                     Send DIO message
12:             end if
13:             go to Step 6
14:     end for
15:     if Any of the node does not receive DIO then
16:             Sends DIS message
17:     end if
18:     Go to Step 3
19:     return
20: end procedure
```

An algorithm to detect and correct DODAG inconsistencies is described in Algorithm 2. Each edge of the DODAG is verified to identify the inconsistencies in the network. Rank of the parent node, Node ID and the rank of each node and of its neighbors is provided by the mapper. In order to detect correct information, the rank of both the nodes is verified.

If an attack leads to routing inconsistencies (rank attack), then it is handled as follows:

a. Faulty nodes are removed from the white-list of the mapper
b. Inconsistencies are corrected based on the rank of the neighbor node

It is necessary to identify properly working nodes and all available nodes. There is a possibility of multiple attacks from compromised nodes. In case of selective forwarding attack compromised node can drop CoAP traffic. RPL DODAG also maintains white-list of valid nodes. Actual nodes of RPL DODAG are compared with the white-listed nodes to identify the blacklisted nodes. This filtering of the nodes is described in Algorithm 3.

Algorithm 2 Identification and correction of RPL DODAG Inconsistencies

```
1: procedure check-inconsistency(NodesN)
2:      for All Nodes in the network do
3:              for each neighbor do
4:                      Compute  the difference in the rank
5:                      Compute the Average difference
6:                      if The computed Difference is > 0.2 then
7:                              Increase the number of faulty nodes
8:                      end if
9:              end for
10:     end for
11:     for Node in N do
12:             if The number of faulty nodes is > Fault-Threshold then
13:                     Update the rank based on the neighbor
14:             end if
15:     end for
16:     return
17: end procedure
```

Algorithm 3 Filtered Nodes

```
1: procedure Filter-nodes (Nodes white-list)
2:      for Node in white-list do
3:              if Node known to RPL DODAG then
4:                      Add the node to Filtered set.
5:              end if
6:      end for
7:      return
8: end procedure
```

In the case of RPL networks, the rank of the nodes is increasing towards the leaf as shown in Figure 3. Some times even topology follows the RPL parent. In child rank relationship there is a possibility of inconsistency. Inconsistency with respect to parent-child relationship can also be identified by comparing the rank of the host node and its parent. This need to be within the range Min-Hop-Rank-Increase (value set for the simulation) [26] as described in Algorithm 4.

Algorithm 4 Rank Inconsistency Identification

```
procedure Find-Rank-inconsistency (Nodes ALLN)
    for Every Node in ALLN do
        if Rank of a Node + Min-Hop-Rank-Increase < Parent Node's rank then
            update the fault-node count
        end if
    end for
    for Every Node in ALLN do
        if Node's fault count is > Fault-Threshold then
            Raise an alarm
        end if
    end for
    return
end procedure
```

Simulated data consists of traffic data which is in “pcap” format. It has both normal and malicious packets. This data is clustered using K-means to get clusters which are normal and clusters of other attacks. After getting clusters, these are labeled accordingly as the normal, rank attack, selective forwarding attacks, and DOS. This gives supervised dataset which can be used for building prediction models for identifying these attacks. Machine learning algorithm Random Forest Classifier is used to build a predictive model and further classification. Different models are built with different network size and malicious nodes. For each accuracy and False Positive Rate are recorded.

4. RESULTS AND ANALYSIS

Wireless sensor network having normal and malicious nodes is simulated on Contiki operating system using Cooja simulator. Series of simulations are carried out by varying the number of nodes. The behavior of the network is observed in three different cases:

a. 10% increase in malicious nodes out of 10, 40 and 100 normal nodes
b. 20% increase in malicious nodes out of 10, 40 and 100 normal nodes
c. 30% increase in malicious nodes with 10, 40 and 100 normal nodes

The server node acts as the root of the DODAG. Simulation timings are controlled by the Cooja plug-in namely Contiki test editor. The network scenario is shown in Figure 5. The green area resents the transmission rage of node 7 and the gray area indicates the interference range. Node 1 is the server, node 10 is the malicious node and the remaining are client nodes. To avoid the losses at the transmitter, Transmission ratio (TX) is set as 100%. Hence losses are allowed only at the receiving end. So the simulation parameter RPL-MOP is set as NO_DOWNWARD_ROUTE indicates that multipoint to point traffic is considered for the evaluation.

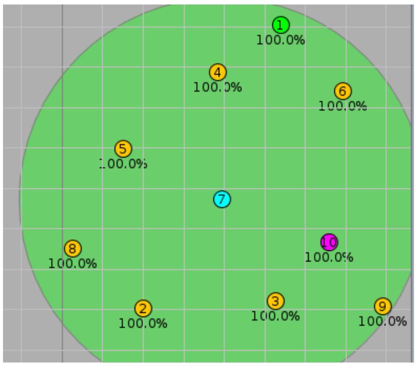
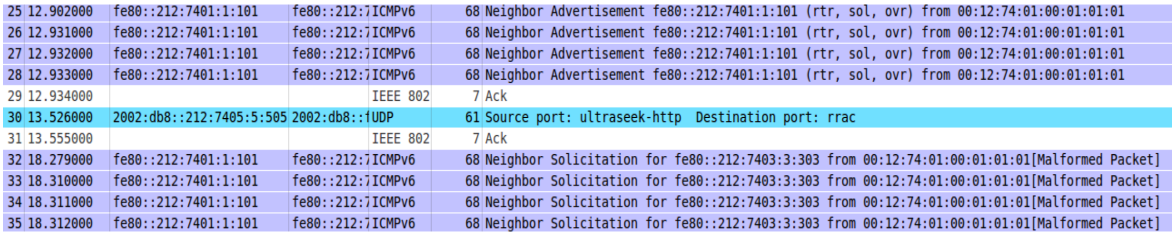


Figure 5. Simulated network with malicious and non-malicious motes

Figure 6 shows the result of the simulation that is the network traffic “pcap” with malicious activity. It can be observed from Figure 6 that the Packets with serial number 32 – 35 are the packets with malicious activities [malformed packets].



```
25 12.902000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01
26 12.931000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01
27 12.932000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01
28 12.933000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Advertisement fe80::212:7401:1:101 (rtr, sol, ovr) from 00:12:74:01:00:01:01:01
29 12.934000                                      IEEE 802   7 Ack
30 13.526000  2002:db8::212:7405:5:505 2002:db8::1 UDP      61 Source port: ultraseek-http  Destination port: rrac
31 13.555000                                      IEEE 802   7 Ack
32 18.279000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Solicitation for fe80::212:7403:3:303 from 00:12:74:01:00:01:01:01[Malformed Packet]
33 18.310000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Solicitation for fe80::212:7403:3:303 from 00:12:74:01:00:01:01:01[Malformed Packet]
34 18.311000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Solicitation for fe80::212:7403:3:303 from 00:12:74:01:00:01:01:01[Malformed Packet]
35 18.312000  fe80::212:7401:1:101    fe80::212:7 ICMPv6    68 Neighbor Solicitation for fe80::212:7403:3:303 from 00:12:74:01:00:01:01:01[Malformed Packet]
```

Figure 6. Sample traffic which shows malicious packet

4.1. Performance metrics

Performance of the network in case of intrusions is discussed in this section. Objective function considered is ETX, which works based on the hop count. The performance metrics considered are listed below:

a. Percentage of inconsistency
b. Energy consumed by the entire network
c. Average power consumed by the network
d. Accuracy and False Positive Rate for the different network scenarios

Table 2 shows the basic operational condition of Tmote-sky which is used for the experimentation. Based on these parameters energy and power consumed by the entire network for different test cases is computed using (1) and (2).

$$Energy(mJ) = \frac{(tr*19.5+lt*21.8+CPU*1.8+LPM*0.0545)mA*3V}{4096*8} \qquad (1)$$

where tr is transmission time, and lt is listen time.

$$Power(mW) = \frac{Energy(mJ)}{Time(ms)} \qquad (2)$$

Table 2. Basic operational conditions of Tmote-sky

MCU Mode	Operating Conditions	Minimum	Maximum	Normal
On	Radio RX	2.1 v	3.6 v	3.2 v
Standby	--		21.0 µA	5.1 µA
On	Radio TX		23 mA	21.8 mA
Idle	Radio On		2400 µA	1800 µA
On	Radio off		21 mA	19.5 mA

Figure 7 to Figure 10 shows the percentage of inconsistency rate for different network sizes with varying number of malicious nodes. From Figure 7 it can be observed that initially with less number of nodes the inconsistency observed is about 85%, whereas a decrease in inconsistency can be observed as the size of the network increases. For the larger networks, RPL network takes some time to become stable. Inconsistencies are identified after 15 minutes, for the network scenario with 40 nodes, whereas for10 nodes it takes nearly 10 minutes. Network-wide energy usage for different network sizes such as 10 nodes, 40 nodes and 100 nodes in a duty cycled RPL network is shown in Figure 10. From this figure, it can be identified that the network with less nodes results in less overhead. A duty cycling MAC protocol in Contiki namely ContikiMAC is used. The default ContikiMACs settings are used. It has that of 8 wakes up per second, and without traffic, the radio is on for 0.6% of the time. An experiment is carried out by considering a network of 10, 40 and 100 simulated Tmote-sky nodes. Figure 10 shows the energy used by the network of sensor nodes for 20 minutes. This Energy computation is carried out using the formula shown in (1).

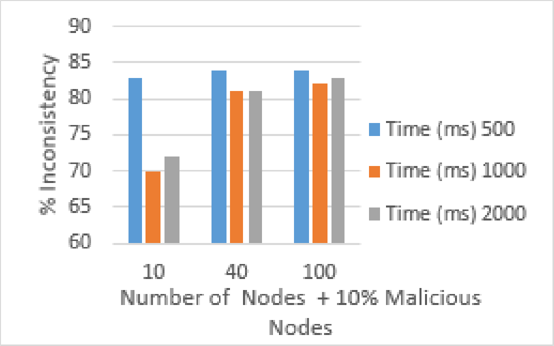


Figure 7. Percentage of inconsistency with 10% increase in malicious nodes

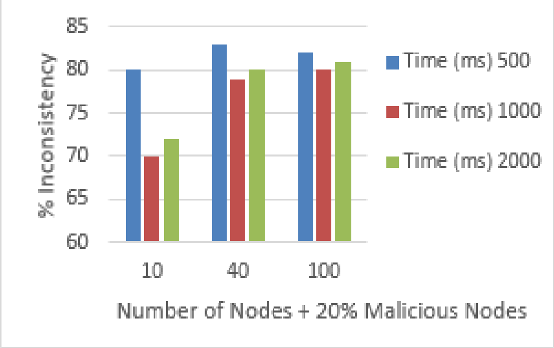


Figure 8. Percentage of inconsistency with 20% increase in malicious nodes

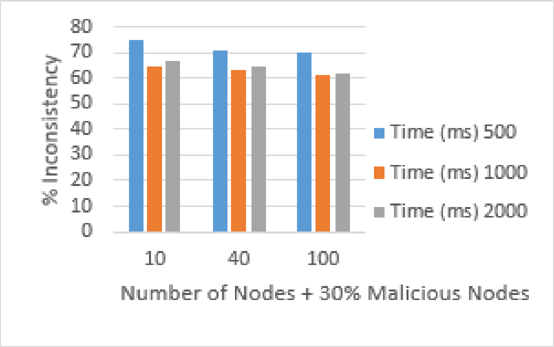


Figure 9. Percentage of inconsistency with 30% increase in malicious nodes

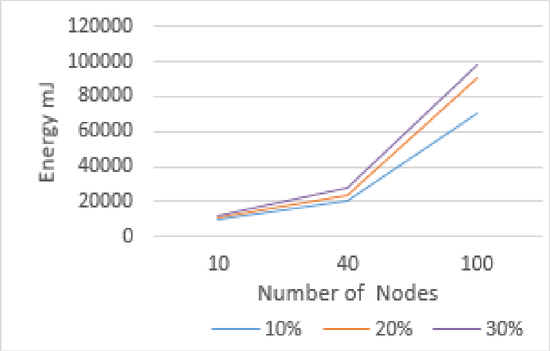


Figure 10. Network-wide energy usage with % increase in malicious nodes

The average power consumption of the network is calculated using (2). Table 3 shows the average power consumption of the network with the change in the network size. From Table 3, it can be observed that the average power consumption increases with the increase in the number of nodes. From Table 3, it can be observed that with the increase in malicious nodes, the power consumed by the network also increases. For 30% increase in malicious nodes the utilization of power throughout network increases drastically and hence the performance of the network starts decreasing.

Table 3. Average power consumption of WSN

Number of nodes	Average power consumption (mW)		
	10% Increase in Malicious node	20% Increase in Malicious node	30% Increase in Malicious node
10	0.76	0.79	0.92
40	0.86	0.90	1.2
100	1.2	1.4	1.8

4.2. Network performance for multiple attacks

Results of the experiment for selective forwarding attack, Rank attack and Denial Of service attack with the different number of malicious nodes and varied network sizes is shown in Figure 11 and Figure 12. From Figure 11 it can be observed that as the number of malicious nodes increases, the accuracy of the network starts decreasing. If the malicious nodes increase more than 30%, it is difficult to differentiate between the normal nodes and malicious nodes accurately. Hence the accuracy decreases drastically after this point. Similarly, from Figure 12, it can be observed that as the number of malicious nodes increases false positive rate decreases.

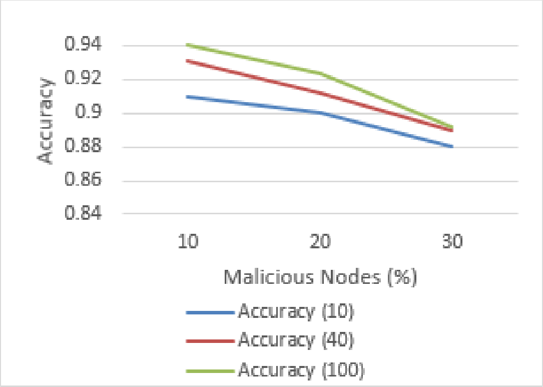


Figure 11. Accuracy of the RPL network with (%) increase in malicious nodes

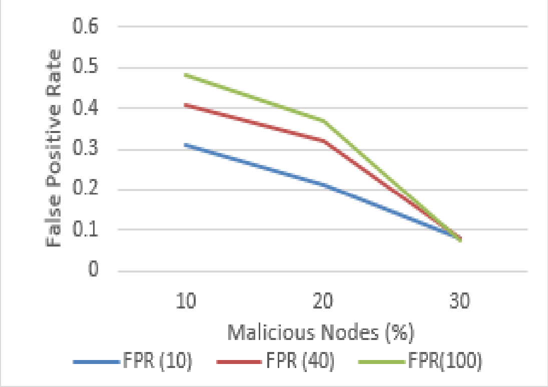


Figure 12. False positive rate of the RPL network with (%) increase in malicious nodes

5. CONCLUSION

In this paper intrusion detection in RPL based WSN is discussed. WSN is simulated using Cooja simulator on the Contiki operating system. WSN network is simulated by increasing the malicious nodes percentage by varying the network sizes. Multiple intrusions on such simulated network are identified and the behavior of the network is analyzed for the different parameters such as accuracy, false positive rate percentage of inconsistency and the energy consumption. Experimental results indicate that as the percentage of malicious nodes increase the performance of the network reduces drastically. Because it is difficult to differentiate between the normal node and malicious nodes effectively. Hence it can be concluded that multiple attacks can be identified effectively in RPL based networks using machine learning techniques.

REFERENCES

[1] N. Kushalnagar, *et al*., "IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals," RFC4919, 2007.

[2] T. Tsao, *et al*., "A Security Threat Analysis for the Routing Protocol for Low-Power and Lossy Networks (RPLs)," RFC 7416, 2015.

[3] T. Winter, *et al*., "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," RFC 6550, 2012.

[4] Jo Y., *et al*., "Design and implementation of heterogeneous surface gateway for underwater acoustic sensor network," *International Journal of Electrical and Computer Engineering*, vol. 9, pp. 1226-1231, 2019.

[5] Ali, *et al*., "Real-time heart pulse monitoring technique using wireless sensor network and mobile application," *International Journal of Electrical and Computer Engineering*, vol. 8, pp. 5118-5126, 2018.

[6] S. Zander, *et al*., "Automated traffic classification and application identification using machine learning," *Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary*, pp. 250-257, 2005.

[7] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks*, vol. 1, pp. 293-315, 2003.

[8] Y. Xu, *et al*., "Detecting wormhole attacks in wireless sensor networks," *Critical Infrastructure Protection, and S. Shenoi, Eds. Boston, MA: Springer US*, pp. 267-279, 2008.

[9] N. Jiang, *et al*., "Routing attacks prevention mechanism for RPL based on micropayment scheme," *Procedings of International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pp. 835-841, 2016.

[10] A. Mayzaud, *et al*., "Mitigation of topological inconsistency attacks in RPL-based low-power lossy networks," *International Journal of Network Management*, vol. 25, pp. 320-339, 2015.

[11] M. Nikravan, *et al*., "A Lightweight Defense Approach to Mitigate Version Number and Rank Attacks in Low-Power and Lossy Networks," *Wireless Personal Communications*, vol. 99, pp. 1035-1059, 2018.

[12] Fadele, *et al*., "Internet of Things security: A survey," *Journal of Network and Computer Applications*, vol. 88, 2017.

[13] H. Sedjelmaci, *et al*., "Novel Hybrid Intrusion Detection System for Clustered Wireless Sensor Network," *International Journal of Network Security & Its Applications*, vol. 3, 2011.

[14] Y. Maleh, *et al*., "A global hybrid intrusion detection system for wireless sensor networks," *Procedia Computer Science, 6th International Conference on Ambient Systems, Networks and Technologies (ANT)*, vol. 52, pp. 1047-1052, 2015.

[15] Belavagi M. C. and Muniyal B., "Multi Class Machine Learning Algorithms for Intrusion Detection - A Performance Study," *Security in Computing and Communications. SSCC 2017. Communications in Computer and Information Science, Springer*, vol. 746, 2017.

[16] L. Wallgren, *et al*., "Routing Attacks and Countermeasures in the RPL-based Internet of Things," *International Journal of Distributed Sensor Networks*, vol. 9, 2013.

[17] Le, *et al*., "The impact of rank attack on network topology of routing protocol for low-power and lossy networks," *IEEE Sensors Journal,* vol. 13, pp. 3685-3692, 2013.

[18] Raza S., *et al*., "SVELTE: Real-time intrusion detection in the Internet of Things," *Ad Hoc Networks,* vol. 1, pp. 2661-2674, 2013.

[19] H. Lamsaazi, *et al*., "Study of the Impact of Designed Objective Function on the RPL-Based Routing Protocol," *Chapter Advances in Ubiquitous Networking, Lecture Notes in Electrical Engineering*, vol. 2, pp. 67-80, 2016.

[20] W. Mardini, *et al*., "Comprehensive Performance Analysis of RPL Objective Functions in IoT Networks," *International Journal of Communication Networks and Information Security (IJCNIS),* vol. 9, pp. 323-332, 2017.

[21] W. Tang, *et al*., "Analysis and optimization strategy of multipath RPL- Based on the COOJA simulator," *Int. J. Comput. Sci.,* vol. 11, pp. 27-30, 2014.

[22] D. Shreenivas, *et al*., "Intrusion Detection in the RPL-connected 6LoWPAN Networks," *Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security*, pp. 31-38, 2017.

[23] Contiki. Available: http://www.contiki-os.org/

[24] A. Dunkels, *et al*., "Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors," *Proceedings of the 1st IEEE Workshop on Embedded Networked Sensors (Emnets-I),* Tampa, Florida, USA, 2004.

[25] A. Dunkels, *et al*., "The Contiki operating system," 2012. Available: http://www.sics.se/contiki/.

[26] S. Thombre, *et al*., "IP based Wireless Sensor Networks: Performance Analysis using Simulations and Experiments," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications,* vol. 7, pp. 53-76, 2016.

BIOGRAPHIES OF AUTHORS

Manjula C Belavagi has received the B.E. degree in Computer Science and Engineering from Karnatak University, Dharwad and M.Tech from Visvesvaraya Technological University, Belgaum. Currently doing PhD in Manipal Academy of Higher Education, Manipal, India. She is currently working as Assistant Professor in the Department of Information & Communication Technology, Manipal Institute of Technology, Manipal. Her area of research interests include Machine Learning, Game Theory and Network Security.

Balachandra Muniyal received the B.E. degree in computer science and engineering from Mysore University and the M.Tech. and Ph.D. degrees in computer science and engineering from the Manipal Academy of Higher Education, Manipal, India. He carried out his M.Tech. project work in T-Systems Nova GmbH, Bremen, Germany. He was deputed to Manipal International University, Malaysia, in 2014. He is currently a Professor and the Head with the Department of Information & Communication Technology, Manipal Institute of Technology, Manipal. He has 25 years of teaching experience in various Institutes. He has more than 30 publications in national and international conferences/journals. His research interest includes network security.