INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Secured Technique to Detect and Avoid Malicious Nodes in Internet of Things

P Calduwel Newton[1], **F Jabez Samuel**[2]*

1 Department of Computer Science, Government Arts College (Affiliated to Bharathidasan University, Tiruchirappalli), Tiruchirappalli, 22, Tamil Nadu, India
2 Department of Information Technology, Bishop Heber College (Autonomous) (Affiliated to Bharathidasan University, Tiruchirappalli), Tiruchirappalli, 17, Tamil Nadu, India

OPEN ACCESS

Received: 04-07-2022

Accepted: 16-10-2022

Published: 30-11-2022

Citation: Newton PC, Samuel FJ (2022) Secured Technique to Detect and Avoid Malicious Nodes in Internet of Things. Indian Journal of Science and Technology 15(44): 2363-2374. https://doi.org/10.17485/IJST/v15i44.1387

* **Corresponding author**.

samueljabez2@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: To detect rank attacks during topology establishment and updated the RPL Destination Oriented Directed Acyclic Graph (DODAG) formation algorithm. The algorithm's distributed module runs across all participating nodes, while the centralized module runs in the sink. **Methods:** The integrity and authenticity of control messages transmitted among two nodes and the sink are verified using a lightweight Hashed Message Authentication Code - Light-weight One-way Cryptographic Hash Algorithm (HMAC-LOCHA). The Secured Technique to Detect and Avoid Malicious Nodes (STDAMN) technique is proposed to overcome the rank attack of the nodes. **Findings:** The proposed scheme STDAMN outperforms the LEADER and SBIDS schemes when considering 50% malicious nodes, the accuracy rate of STDAMN is 3% higher than LEADER and 17% higher than SBIDS in Security mode whereas it is 4% and 14% higher in non-security mode respectively in the decreased rank attack. Again considering 50% malicious nodes, the accuracy rate of STDAMN is 2% higher than LEADER and 13% higher than SBIDS in with-Security mode whereas it is 2.2% and 16.1% higher in without-security mode respectively in the increased rank attack. Also indeed, the false positive rate for STDAMN is lower by 72.5% and 72%, 15.3% and 21.2% whereas the false negative rate for STDAMN is lower by 77.2% and 62.1%, 32.5%, and 39.5% on average for with-security and without-security respectively than LEADER and SBIDS in the decreased rank attack. **Novelty:** This paper presents a rank attack detection approach for non-storing mode RPL used in IoT to cope with both increased and decreased rank attacks to address this issue. The performance of the suggested technique is assessed both conceptually and through simulation using the Contiki-based Cooja simulator. The proposed technique surpasses state-of-the-art rank attack detection techniques in terms of detection accuracy and false positive/negative rate while maintaining acceptable network performance, according to simulation results.

Keywords: Security; RPL; Rank attack; HMAC; Malicious Node; DODAG

1 Introduction

RPL is a fundamental routing system for IPv6 over low-power wireless personal area networks that have been suggested (6LoWPAN). The IETF suggested RPL as a standard protocol for Low power and Lossy Networks LLNs in March 2012, and it was recently modified. IPv6 provides Internet connectivity, as well as lowers the cost of contacting the root (base station) from any LLN node. RPL creates a topology based on a directed acyclic graph (DAG), which is a mathematical graph model with no directed cycles. The distance vector rules were used to generate this graph. The data and DAG nodes converge on a single sink node. RPL is a multi-hop routing protocol in which each node can connect to a large number of other nodes.

In the existing work, researchers have proposed Various security enhancement-related techniques as follows: In [1] proposed a Rank attack detection technique to detect increased and decreased rank attacks in IoT networks based on RPL. The author modified existing RPL by adding security mechanisms and modifying the Destination Advertisement Object (DAO) control messages which results in a low overhead of messages. The disadvantage of this technique is it underperforms during the mobility of nodes and storage can be implemented. In [2] proposed a scheme to avoid copycat attacks in IPv6-based Low-power Wireless Personal Area Networks (6LoWPANs). The copycat attack degraded network performance concerning Packet Delivery Ratio (PDR) and end-to-end delay of packets. The drawbacks in this scheme are undetected spoofed copycat attacks, maintenance of neighboring node tables for storing adjacent nodes.

In [3] suggested a scheme to optimize node to node data forwarding, also to avoid redundant message transmission, and backward compatibility with traditional RPL. This scheme resulted in high PDR and successful node-to-node communication. This scheme is not suitable for dynamic nodes and Scalability. In [4] reviewed different types of network attacks in the RPL protocol. In [5] proposed a mechanism based on the divide and conquer method to identify pre-defined malicious nodes. This mechanism identified the various vulnerabilities in the traditional RPL protocol. The flaw in this mechanism is the malicious nodes are less identified which leads to an unstable network.

In [6] proposed a technique for multipath and multi-nodes heterogeneous LLN. This technique guaranteed protection and reliability against network attacks at a low cost. The disadvantage of this technique is the amount of packet loss is not taken into consideration. In [7] proposed a technique for the detection of DAO, DODAG Information Object (DIO), and DODAG Information Solicitation (DIS) message attacks. This technique suggested a solution for DODAG root and neighboring node attacks. The disadvantage of this technique is the PDR ratio decreases with an increase in the number of nodes. Also, the latency increases concerning the number of nodes.

In [8] suggested a technique to identify false positives, communication overhead, and firmware modification. This technique added rules to avoid unknown attacks. This technique proposed the solution without the need for device updates. In [9] reviewed that most techniques do not provide solutions for multiple attacks, particularly in rank attack and version number at the same time. Also, most techniques have lower performance and detection accuracy.

In [10] reviewed various attacks in RPL protocol and their detection techniques out of which selected a few trusted models and classified them into categories. In [11] surveyed various techniques to address numerous issues in the Rank attack and stated that RPL still suffers from different security threats which were not addressed in recent research papers. In [12] compared possible attacks and classified based on Confidentiality, Integrity, and Availability. Also, studied their countermeasures with trust-based solutions.

In [13] proposed an algorithm to secure the RPL protocol with trust parameters and mobility parameters that allows only the trusted nodes to take part in data transmission. This algorithm failed to identify the behavior of the malicious node and also to identify the rank attacks. In [14] Proposed an algorithm to defend against a replay-based attack that reduces the quantity of RPL control messages and also increases the number of optimal routes. This algorithm also substantially increases the network performance. In [15] developed a protocol for LLN communication with objective functions Received Signal Strength Indicator (RSSI) and Expected Transmission Count (ETX) for selecting Parent Node. This protocol offers reliability and less end-to-end delay.

In [16] proposed a technique for the detection of FDI attacks this technique provides stability to the nodes. This algorithm also reduced the node fault transmission and improved accuracy of detection of malicious nodes. The limitation of this technique is with an increased proportion of malicious nodes, the recall is also increased. [17] provided a survey on different security-based routing protocols for IoT. And investigated RPL attacks and their impact concerning control overhead and network performance. [18] proposed a lightweight model using Machine Learning (ML) based on a one-class classifier to detect IoT botnets with high accuracy.

In [19] proposed a scheme to avoid clone attack and attains high residual energy and throughput. In [20] proposed two models to detect malicious nodes and find the optimal route to avoid a few routing attacks. [21] reviewed different strategies for intrusion detection systems in IoT and addressed the classification of attacks in IoT. In [22] proposed a model to identify misbehaving nodes in RPL black hole attacks to ensure authentication. In [23] provides all the information about RPL with formulas and definitions. In[24] provided the information of the simulation model with a description

Detecting and avoiding the malicious nodes in RPL become a security challenge and addressing these areas of research helps ensure the security of IoT. So, there is a need to improvise the accuracy rate in terms of false positive and false negative rate with respect to increased and decreased rank attacks

2 Methodology

2.1 STDAMN: A Proposed Work

The proposed Secured Technique to Detect and Avoid Malicious Nodes (STDAMN) specifies a distributed model that supports finding and storing the disrupted wireless nodes that are already registered in a DODAG to maintain the state information. By saving this information in a separate table, it helps to identify the intruder node by verifying the details stored in the state information about the registered node in the DODAG. The process is to verify the stored rank information about the node which is interrupted and compare it with the new node trying to register as a parent node to complete the DODAG for message transfer. The parent rank and the rank of the child node can be received through the DIO message as per the RPL procedure. It is sent through the DAO ack message for successful registration or rejection of the new request from registering node. The intruder node is identified by comparing the rank sent from the newly registering node with the help of solicitation and the stored information of the previous disrupted node. This saves the delay in identifying the intruder node and further the information about the intruder node is stored for further verifications.

Packet sequence numbers from the source node are analyzed and if found legitimate they are transmitted to the destination, otherwise, the packets are considered as malicious packets and ignored. The various steps involved in STDAMN are as follows,

```
Input: DAO Control Message
Output: Malicious Node IDs and Packet Sequence number
Triggers: Node Compromise Detection when sink node receives DAO control messages
in the network
Begin
  Read DAO control message from the child node
  Retrieve CNID, CNR, PNID and PNR from DAO control message
  Generate MAC = HMAC ((CNID ||CNR ||PNID ||PNR), SK) from the received values
  Insert < CNID , CNR, PNID, PNR > into Information Table
  if (Node N is a parent node) then
    CPNR = getChildParentRank(CNID)
    if ($CNR \neq CPNR$ ) then
      Insert CNID into Malicious_Nodes // a set to store malicious node IDs
    End if
  End if
  If (CNR < PNR+ MinHopRankIncrease) then
    insert MNID into Malicious_Nodes_Table ;
    Discard the DAO control message;
    Reject node with NODE_ID = MNID
  Else If(CNR > PNR + MinHopRankIncrease) then
    insert MNID into Malicious_Nodes_Table;
    Discard the DAO control message;
    Reject node with NODE_ID = MNID
  End if
  End if
  If (Node is Child) then
    getPacketNumber(CNID)
  else if (Node is Parent) then
    getPacketNumber(PNID)
  End if
  End if
End
Function getChildParentRank(CNodeID):
```

```
Begin
for (Entry i in the Information_Table) do
if (CNodeID is the parent node) then
return ParentRank
end if
end
Function getPacketNumber(Node N)
Begin
If (Packet_ID(MNID) $\neq$ Packet_ID(PNID) ) then
Process packet from $PN_{ID}$ and forward packet to sink
Reject packet from $MN_{ID}$
End if
If (Packet_ID(MNID) $\neq$ Packet_ID(CNID) ) then
Process packet from CNIDand forward packet to sink
Reject packet from MNID
End if
End
where,
CNID – Child node ID
CNR – Child Node Rank
PNID – Parent Node ID
PNR – Parent Node Rank
SK – Secret Key
CPNR – Child parent node rank
```

3 Results and Discussion

The simulation results of both the techniques STDAMN and the LEADER in with-basic-security and without basic-security modes using HMAC. This is done to measure the additional overheads incurred by each of the schemes for incorporating security features. In maintaining such securities, the STDAMN and LEADER use a cryptographic MAC (HMAC using LOCHA) and the SBIDS uses cryptographic encryption and decryption (AES-128).

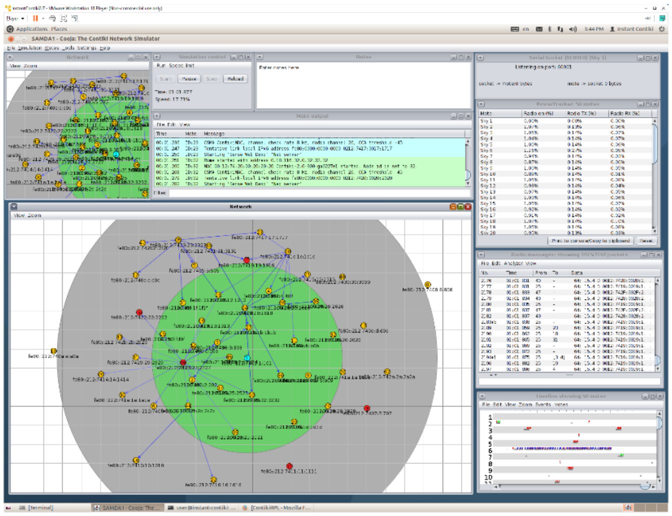


Fig 1. A Sample scenario of simulation with 50 nodes

Figure 1 represents the Snapshot of the simulation with 50 nodes in Cooja Simulator. The simulation is carried out in Cooja with 10% malicious nodes (i.e., 5 nodes) where the blue node is the sink and yellow nodes are the legitimate nodes. The malicious nodes are represented in red and located randomly in the network. Also conducted two sets of experiments to measure the extent to which design objectives have been attained for all the techniques.

3.1 Accuracy Rate

The detection accuracy is defined as the ratio of the number of nodes that are detected correctly (legitimate or attacker) to the total number of nodes present in the network.

$$Accuracy = (TP+TN)/(TP+FP+TN+FN) \tag{1}$$

where TP is the number of true positive samples that are classified as positive, FP is the number of negative samples that are classified as positive, TN is the number of negative samples classified as negative, and FN is the number of positive samples classified as negative.

Table 1 indicates the accuracy percentage of STDAMN for 10, 25, and 50 nodes. The values of TP, TN, FP, and FN are generated by STDAMN.

Table 1. Accuracy rate of STDAMN in Decreased Rank Attack Detection with security

	Nodes	TP	TN	FP	FN	Accuracy %
STDAMN	10	5	5	0.02	0.03	100
	25	17	8	0.13	0.37	98
	50	37	13	0.4	1.1	97

Scenario for 25 nodes in STDAMN

Accuracy = (17+8)/(17+0.13+8+0.37) = 0.9803

Table 1 shows the accuracy rate of malicious node detection with a varying number of malicious nodes in the network.

Table 2 indicates the accuracy % of LEADER for 10, 25, and 50 nodes. The values of TP, TN, FP, and FN are generated by the LEADER Technique.

Table 2. Accuracy rate of LEADER in Decreased Rank Attack Detection with security

	Nodes	TP	TN	FP	FN	Accuracy %
LEADER	10	7	3	0.03	0.02	100
	25	17	8	0.25	0.52	97
	50	30	20	0.85	2.25	94

Scenario for 25 nodes in LEADER

Accuracy = (17+8)/(17+0.25+8+0.515) = 0.9703

Table 3. Comparison of STDAMN with existing techniques based on Accuracy rate in Decreased Rank Attack Detection

Algorithm	No of Malicious Nodes (%)	Accuracy (%)
STDAMN (With Security)	10	100
	25	98
	50	97
STDAMN (Without Security)	10	100
	25	98
	50	96
LEADER (With Security)	10	100
	25	97
	50	94
LEADER (Without Security)	10	100
	25	96.5
	50	92
SBIDS (With Security)	10	98
	25	95
	50	80

Continued on next page

Table 3 continued

SBIDS (Without Security)	10	98.5
	25	91
	50	82

It is observed that the accuracy rate decreases with the increasing number of malicious nodes in both without security and with-security cases for all the techniques. The reason for such results is sometimes a part of the network gets disconnected from the rest of the network due to rank attacks. In the network partitioning scenario, malicious nodes present in one part may not be detected by the sink as DAO messages from such nodes do not reach the sink. However, this decreasing rate of accuracy for STDAMN is very less compared to LEADER and SBIDS. For example, considering 50% malicious nodes, the accuracy rate of STDAMN is 3% higher than LEADER and 17% higher than SBIDS in with-Security mode, whereas the accuracy rate of STDAMN is 4% higher than LEADER and 14% higher than SBIDS in the without-security mode in the decreased rank attack.

Table 4. Comparison of STDAMN with existing techniques based on Accuracy rate in Increased Rank Attack Detection

Algorithm	No of Malicious Nodes (%)	Accuracy (%)
STDAMN (With Security)	10	99.5
	25	98.2
	50	97
STDAMN (Without Security)	10	99.4
	25	97.9
	50	96.1
LEADER (With Security)	10	99
	25	97.3
	50	95
LEADER (Without Security)	10	98.8
	25	95.5
	50	93.9
SBIDS (With Security)	10	98
	25	94
	50	84
SBIDS (Without Security)	10	97.6
	25	90
	50	80

However, this decreasing rate of accuracy for STDAMN is very less compared to LEADER and SBIDS. For example, considering 50% malicious nodes, the accuracy rate of STDAMN is 2% higher than LEADER and 13% higher than SBIDS in with-Security mode whereas it is 2.2% and 16.1% higher in without-security mode respectively in the increased rank attack.

STDAMN performs an additional rank inconsistency checking by retrieving malicious nodes' rank from its parent and child nodes as well in both increased and decreased attacks. Also, the packets from the malicious nodes are analyzed and rejected. These results show that STDAMN performs better than LEADER and SBIDS.

3.2 FALSE POSITIVE RATE AND FALSE NEGATIVE RATE

False Positive Rate (FPR) is defined as the ratio of the number of legitimate nodes that are incorrectly detected as attacker nodes to the total number of legitimate nodes present in the network. False Negative Rate (FNR) is defined as the ratio of the number of attacker nodes that are incorrectly detected as legitimate nodes to the total number of attacker nodes present in the network.

$$FPR = FP/\,(TN + FP)\,*100 \tag{2}$$

$$FNR = FN/(FN + TN)*100 \tag{3}$$

Table 5. False Positive / Negative rate of STDAMN in Decreased Rank Attack Detection

	Nodes	TP	TN	FP	FN	FPR	FNR
STDAMN	10	5	5	0.02	0.03	0.39	0.59
	25	17	8	0.13	0.37	1.59	4.4
	50	37	13	0.4	1.1	2.9	7.8

Table 5 indicates the False Positive / Negative rate of STDAMN for 10, 25, and 50 nodes. The values of TP, TN, FP, and FN are generated by STDAMN.

Scenario for 25 nodes in STDAMN

FPR = 0.13/(8+0.13)*100 =1.5990

FNR = 0.37/(0.37+8)*100 = 4.4205

Table 5 shows the false positive and false negative rates in rank attack detection with the varying number of malicious nodes.

Table 6 indicates the False Positive / Negative rate of LEADER for 10, 25, and 50 nodes. The values of TP, TN, FP, and FN are generated by the LEADER Technique.

Table 6. False Positive / Negative rate of LEADER in Decreased Rank Attack Detection

	Nodes	TP	TN	FP	FN	FPR	FNR
LEADER	10	7	3	0.03	0.02	1	0.8
	25	17	8	0.25	0.52	3	6
	50	30	20	0.85	2.25	4	10.1

Scenario for 25 nodes in LEADER

FPR = 0.25/(8+0.25)*100 = 3.0303

FNR = 0.515/(0.515+8)*100 = 6.0481

Table 7. Comparison of STDAMN with existing techniques based on False Positive Rate and False Negative rate in Decreased Rank Attack Detection

Algorithm	No of Malicious Nodes (%)	False Positive rate (%)	False Negative rate (%)
STDAMN (With Security)	10	0.39	0.59
	25	1.59	4.4
	50	2.9	7.8
STDAMN (Without Security)	10	0.5	0.6
	25	2.6	4.9
	50	3.6	8.7
LEADER (With Security)	10	1	0.8
	25	3	6
	50	4	10.1
LEADER (Without Security)	10	1	0.8
	25	4	5.9
	50	5	14
SBIDS (With Security)	10	2	4
	25	7	15
	50	19	24
SBIDS (Without Security)	10	3	0
	25	4	9
	50	17	22

It is observed that both False Positive Rate and False Negative rate are increased with the increasing number of malicious nodes in the network for both the schemes such as with-security and without-security. Similar to the accuracy rate, here also due to occasional network partitioning, the false positive/negative rates increase with the increasing number of malicious nodes.

However, this increase in rates is much lower in STDAMN compared to LEADER and SBIDS. Precisely, the false positive rate for STDAMN is lower by 72.5% and 72%, 15.3% and 21.2% for LEADER and SBIDS respectively, whereas the false negative rate for STDAMN is lower by 77.2% and 62.1%, 32.5%, and 39.5% on average for with-security and without-security respectively than LEADER and SBIDS in the decreased rank attack.

Table 8. Comparison of STDAMN with existing techniques based on False Positive Rate and False Negative rate in Increased Rank Attack Detection

Algorithm	No of Malicious Nodes (%)	False Positive rate (%)	False Negative rate (%)
STDAMN (With Security)	10	0.7	0.7
	25	2.6	4.8
	50	4.2	8.2
STDAMN (Without Security)	10	0.6	0.7
	25	3.1	5.6
	50	4.4	9.7
LEADER (With Security)	10	1.7	1.6
	25	4.1	8
	50	6.6	12.8
LEADER (Without Security)	10	1.9	1.9
	25	5.2	7.1
	50	6.8	18
SBIDS (With Security)	10	2.6	6
	25	8.6	18
	50	23	29.2
SBIDS (Without Security)	10	3.6	0.8
	25	5.4	11.8
	50	21	26

However, this increase in rates is much lower in STDAMN compared to LEADER and SBIDS. Precisely, the false positive rate for STDAMN is lower by 63.6% and 64.7%, 18.3% and 20.9% whereas the false negative rate for STDAMN is lower by 64% and 53.9%, 28%, and 37.3% on average for with-security and without-security respectively than LEADER and SBIDS in the increased rank attack. Here also the reason for LEADER's better performance is similar to the first set of experiments

The proposed technique STDAMN is implemented using the Cooja network simulator by modifying the code of the RPL protocol. The simulation parameters for this scenario is given in Table 9

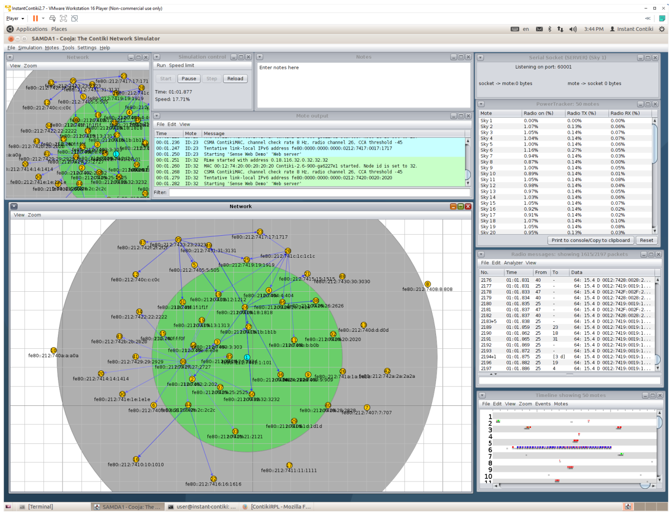
Table 9. Simulation Parameters

Parameter	Values
Routing protocol	RPL
Simulator	Cooja
MAC Layer	IEEE 802.15.4
Area(m)	200 * 200
Radio environment	UDGM
Number of nodes	10,25,50
Mote Type	Tmote Sky
Simulation time(s)	3000
Transmission range(m)	50

Figure 2 shows the simulation environment of STDAMN with 50 nodes in the Cooja Simulator.

Figure 3 shows the accuracy rate in Decreased rank attack detection. It is observed from Figure 4 that the STDAMN is having a higher accuracy rate compared to LEADER and SBIDS.

Figure 4 shows the accuracy rate in Increased rank attack detection. It is observed from Figure 5 that the STDAMN is having a higher accuracy rate compared to LEADER and SBIDS.



Time: 01:01.877
Speed: 17.71%

```
Time       Mote   Message
00:01.236  ID:23  CSMA ContikiMAC, channel check rate 8 Hz, radio channel 26, CCA threshold -45
00:01.247  ID:23  Tentative link-local IPv6 address fe80:0000:0000:0000:0212:7417:0017:1717
00:01.250  ID:23  Starting 'Sense Web Demo' 'Web server'
00:01.251  ID:32  Rime started with address 0.18.116.32.0.32.32.32
00:01.260  ID:32  MAC 00:12:74:20:00:20:20:20 Contiki-2.6-900-ga6227e1 started. Node id is set to 32.
00:01.268  ID:32  CSMA ContikiMAC, channel check rate 8 Hz, radio channel 26, CCA threshold -45
00:01.279  ID:32  Tentative link-local IPv6 address fe80:0000:0000:0000:0212:7420:0020:2020
00:01.282  ID:32  Starting 'Sense Web Demo' 'Web server'
```

Listening on port: 60001

socket -> mote:0 bytes mote -> socket 0 bytes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	0.00%	0.00%	0.00%
Sky 2	1.07%	0.13%	0.06%
Sky 3	1.05%	0.14%	0.07%
Sky 4	1.04%	0.14%	0.07%
Sky 5	1.00%	0.14%	0.06%
Sky 6	1.16%	0.27%	0.05%
Sky 7	0.94%	0.14%	0.03%
Sky 8	0.87%	0.14%	0.00%
Sky 9	1.00%	0.14%	0.05%
Sky 10	0.89%	0.14%	0.01%
Sky 11	1.05%	0.14%	0.08%
Sky 12	0.98%	0.14%	0.04%
Sky 13	0.97%	0.14%	0.05%
Sky 14	1.03%	0.14%	0.06%
Sky 15	1.05%	0.14%	0.07%
Sky 16	0.92%	0.14%	0.02%
Sky 17	0.91%	0.14%	0.02%
Sky 18	1.07%	0.14%	0.10%
Sky 19	1.05%	0.14%	0.08%
Sky 20	0.95%	0.13%	0.03%

No.	Time	From	To	Data
2176	01:01.831	40	-	64: 15.4 D 0012:7428:0028:2...
2177	01:01.831	25	-	64: 15.4 D 0012:7419:0019:1...
2178	01:01.833	47	-	64: 15.4 D 0012:742F:002F:2...
2179	01:01.834	40	-	64: 15.4 D 0012:7428:0028:2...
2180	01:01.835	25	-	64: 15.4 D 0012:7419:0019:1...
2181	01:01.837	47	-	64: 15.4 D 0012:742F:002F:2...
2182	01:01.837	40	-	64: 15.4 D 0012:7428:0028:2...
2183+5	01:01.838	25	-	64: 15.4 D 0012:7419:0019:1...
2189	01:01.859	25	23	64: 15.4 D 0012:7419:0019:1...
2190	01:01.862	25	18	64: 15.4 D 0012:7419:0019:1...
2191	01:01.865	25	31	64: 15.4 D 0012:7419:0019:1...
2192	01:01.869	25	-	64: 15.4 D 0012:7419:0019:1...
2193	01:01.872	25	-	64: 15.4 D 0012:7419:0019:1...
2194+1	01:01.875	25	[3 d]	64: 15.4 D 0012:7419:0019:1...
2196	01:01.882	25	19	64: 15.4 D 0012:7419:0019:1...
2197	01:01.886	25	4	64: 15.4 D 0012:7419:0019:1...

Fig 2. A Simulation in Cooja Simulator

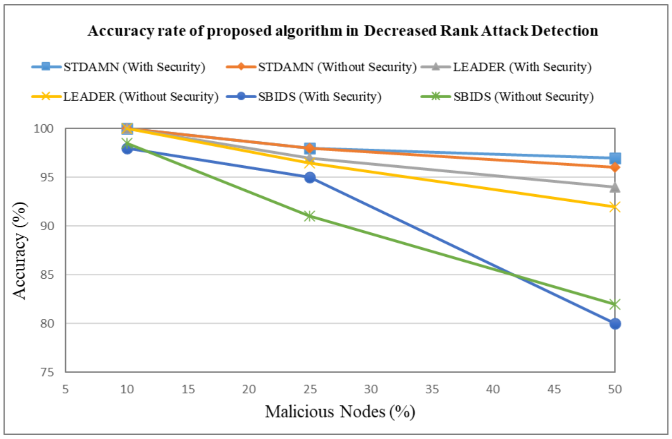


Fig 3. Accuracy Rate in Decreased Rank Attack Detection

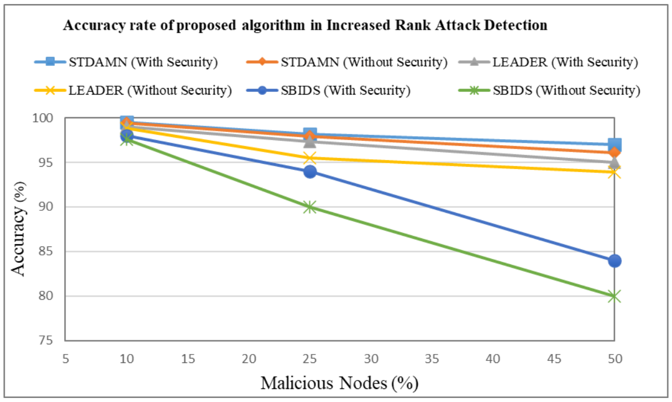


Fig 4. Accuracy Rate in Increased Rank Attack Detection

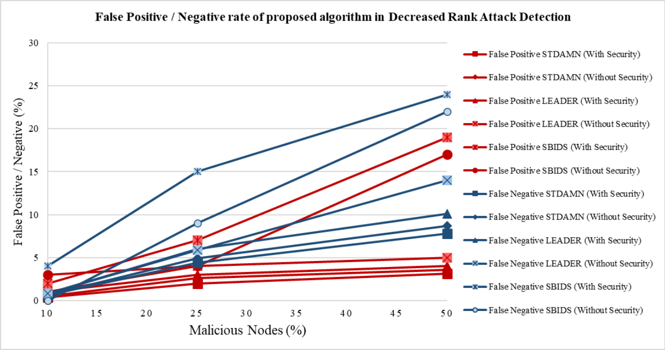


Fig 5. False Positive and Negative Rates in Decreased Rank Attack Detection

Figure 5 shows the False positive rate and the False negative rate in Decreased rank attack detection. It is observed from Figure 6, that the STDAMN is better at having a lower false positive rate and the false negative rate in the case of with-security and without-security compared to LEADER and SBIDS.

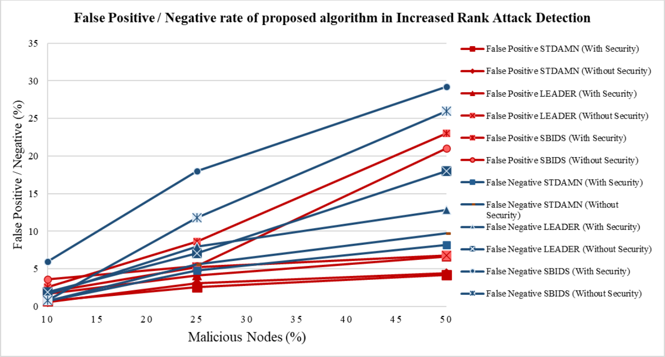


Fig 6. False Positive and Negative Rates in Increased Rank Attack Detection

Figure 6 shows the False positive rate and False negative rate in Increased rank attack detection. It is observed from Figure 6 , that the STDAMN is having a lower false positive rate and false negative rate in the case of with-security and without-security compared to LEADER and SBIDS.

Table 10. Comparitive table

Algorithm	Characteristics	Pros	Cons
LEADER[1]	DAO Modification, Rank Attack Detection,	Detection Accuracy, Security hashing	Non storing mode, Non mobility of nodes
CoSec-RPL[2]	Copycat attack, 6LoWPAN	Increase in PDR, Impact on AE2ED	DIS flooding attacker can be detected, not capable of detecting a spoofed copycat attack
SAMP-RPL[6]	Heterogeneous IoT, Multi-path routing security	Secure multipath routing, Secure loss-driven multipath routing	machine learning approach can be adopted to predict packet loss events, network conditions may lead to packet loss
DETONAR[8]	Detection of Routing Attacks, Intusion Detection System	Routing Attacks Dataset For RPL, Decrease in packet loss	Yet to Investigate in dynamic networks, Yet tp test its performance on large-scale networks

Continued on next page

Table 10 continued

SMTrust [13]	Trust-Based Secure Routing Protocol, Mobility metrics	better network performance, increased scalability	Yet to evaluate the critical trust metrics including mobility metrics, Optimize detection

4 Conclusion

Because rank attack intrusions are critical in RPL, STDAMN is proposed, which is a rank attack detection technique that ensures detection of both decreased and increased rank attacks in an RPL-based IoT network while maintaining basic security throughout control message exchange.

A rank attack detection method and basic security features are incorporated into the traditional RPL algorithm. The distributed module of the method, which runs on the participating nodes, is in charge of adding some extra information to the DAO control message.

This aids the centralized module of the sink in effectively detecting network rank attacks. This technique is a simple extension of the DAO message and the adoption of a hashed MAC method.

The proposed technique STDAMN's performance is compared to that of a state-of-the-art rank attack detection scheme, LEADER and SBIDS, both simulated and experimental, revealing that STDAMN outperforms LEADER and SBIDS in both with and without essential security scenarios. The simulation result shows that the proposed scheme STDAMN outperforms the LEADER and SBIDS schemes in terms of both sets of parameters, including design target parameters and network performance metrics. When considering 50% malicious nodes, the accuracy rate of STDAMN is 3% higher than LEADER and 17% higher than SBIDS in with-Security mode whereas it is 4% and 14% higher in without-security mode respectively in the decreased rank attack. Again considering 50% malicious nodes, the accuracy rate of STDAMN is 2% higher than LEADER and 13% higher than SBIDS in with-Security mode whereas it is 2.2% and 16.1% higher in without-security mode respectively in the increased rank attack. Also indeed, the false positive rate for STDAMN is lower by 72.5% and 72%, 15.3% and 21.2% whereas the false negative rate for STDAMN is lower by 77.2% and 62.1%, 32.5%, and 39.5% on average for with-security and without-security respectively than LEADER and SBIDS in the decreased rank attack. The attack detection technique can be extended in the future to work in many RPL sink nodes and to take into account node mobility.

References

1) Karmakar S, Sengupta J, Bit SD. LEADER: Low Overhead Rank Attack Detection for Securing RPL based IoT. *International Conference on COMmunication Systems & NETworkS (COMSNETS)*. 2021. Available from: https://doi:10.1109/COMSNETS51098.2021.9352937.
2) Verma A, Ranga V. CoSec-RPL: detection of copycat attacks in RPL based 6LoWPANs using outlier analysis. *Telecommunication Systems*. 2020;75(1):43–61. Available from: https://doi.org/10.1007/s11235-020-00674-w.
3) Mahyoub M, Mahmoud ASH, Abu-Amara M, Sheltami TR. An Efficient RPL-Based Mechanism for Node-to-Node Communications in IoT. *IEEE Internet of Things Journal*. 2021;8(9):7152–7169. Available from: https://doi:10.1109/JIOT.2020.3038696.
4) Pasikhani AM, Clark JA, Gope P, Alshahrani A. Intrusion Detection Systems in RPL-Based 6LoWPAN: A Systematic Literature Review. *IEEE Sensors Journal*. 2021;21(11):12940–12968. Available from: https://doi:10.1109/JSEN.2021.3068240.
5) Boudouaia MA, Abouaissa A, Benayache A, Lorenz P. Divide and Conquer-based Attack against RPL Routing Protocol. *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. 2020. Available from: https://doi:10.1109/GLOBECOM42002.2020.9322275.
6) Sahraoui S, Henni N. SAMP-RPL: secure and adaptive multipath RPL for enhanced security and reliability in heterogeneous IoT-connected low power and lossy networks. *Journal of Ambient Intelligence and Humanized Computing*. 2021. Available from: https://doi.org/10.1007/s12652-021-03303-9.
7) Wadhaj I, Ghaleb B, Thomson C, Al-Dubai A, Buchanan WJ. Mitigation Mechanisms Against the DAO Attack on the Routing Protocol for Low Power and Lossy Networks (RPL). *IEEE Access*. 2020;8:43665–43675. Available from: https://doi:10.1109/ACCESS.2020.2977476.
8) Agiollo A, Conti M, Kaliyar P, Lin TN, Pajola L. DETONAR: Detection of Routing Attacks in RPL-Based IoT. *IEEE Transactions on Network and Service Management*. 2021;18(2):1178–1190. Available from: https://doi:10.1109/TNSM.2021.3075496.
9) Almusaylim ZA, Alhumam A, Jhanjhi NZ. Proposing a Secure RPL based Internet of Things Routing Protocol: A Review. *Ad Hoc Networks*. 2020;101:102096. Available from: https://doi.org/10.1016/j.adhoc.2020.102096.
10) Avila K, Jabba D, Gomez J. Security Aspects for Rpl-Based Protocols: A Systematic Review in IoT. *Applied Sciences*. 2020;10(18):6472. Available from: https://doi.org/10.3390/app10186472.
11) Boudouaia MA, Ali-Pacha A, Abouaissa A, Lorenz P. Security Against Rank Attack in RPL Protocol. *IEEE Network*. 2020;34(4):133–139. Available from: https://doi:10.1109/MNET.011.1900651.
12) Mangelkar S, Dhage SN, Nimkar AV. A comparative study on RPL attacks and security solutions. *International Conference on Intelligent Computing and Control (I2C2)*. 2017. Available from: https://doi:10.1109/I2C2.2017.8321851.
13) Muzammal SM, Murugesan RK, Jhanjhi NZ, Jung LT. SMTrust: Proposing Trust-Based Secure Routing Protocol for RPL Attacks for IoT Applications. *International Conference on Computational Intelligence (ICCI)*. 2020. Available from: https://doi:10.1109/ICCI51257.2020.9247818.
14) Arena A, Perazzo P, Vallati C, Dini G, Anastasi G. Evaluating and improving the scalability of RPL security in the Internet of Things. *Computer Communications*. 2020;151:119–132. Available from: https://doi.org/10.1016/j.comcom.2019.12.062.
15) Ganesh DR, Patil KK, Suresh L. A Multicast Transmission Routing Protocol for Low Power Lossy Network Based IoT Ecosystem. *Intelligent Data Communication Technologies and Internet of Things*. 2019;p. 569–582. Available from: https://doi.org/10.1007/978-3-030-34080-3_65.

16) Lai Y, Tong L, Liu J, Wang Y, Tang T, Zhao Z, et al. Identifying malicious nodes in wireless sensor networks based on correlation detection. *Computers & Security*. 2022;113:102540. Available from: https://doi.org/10.1016/j.cose.2021.102540.

17) Simoglou G, Violettas G, Petridou S, Mamatas L. Intrusion detection systems for RPL security: A comparative analysis. *Computers & Security*. 2021;104:102219. Available from: https://doi.org/10.1016/j.cose.2021.102219.

18) Malik K, Rehman F, Maqsood T, Mustafa S, Khalid O, Akhunzada A. Lightweight Internet of Things Botnet Detection Using One-Class Classification. *Sensors*;22(10):3646. Available from: https://doi.org/10.3390/s22103646.

19) Vaishnavi S, Sethukarasi T. Detection and Avoidance of Clone Attack in IoT Based Smart Health Application. *Intelligent Automation & Soft Computing*. 2022;31(3):1919–1937. Available from: https://doi.org/10.32604/iasc.2022.021006.

20) Bint M, Sajid E, Ullah S, Javaid N, Ullah I, Qamar AM, et al. Exploiting Machine Learning to Detect Malicious Nodes in Intelligent Sensor-Based Systems Using Blockchain. *Wireless Communications and Mobile Computing 2022 73860491- 16*. 2022. Available from: https://doi.org/10.1155/2022/7386049.

21) Khraisat A, Alazab A. A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. *Cybersecurity*. 2021;4(1). Available from: https://doi.org/10.1186/s42400-021-00077-7.

22) Prathapchandran K, Janani T. A Trust-Based Security Model to Detect Misbehaving Nodes in Internet of Things (IoT) Environment using Logistic Regression. *Journal of Physics: Conference Series*. 2021;1850(1):012031. Available from: https://doi:10.1088/1742-6596/1850/1/012031.

23) RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. 2012. Available from: https://tools.ietf.org/html/rfc6550.

24) MSP430 Microcontrollers MSP430F1611, 8 MHz MCU with 48KB Flash, 10KB SRAM, 12-bit ADC, Dual 12-bit DAC, comparator, DMA, I2C/SPI/UART. . Available from: https://www.ti.com/product/MSP430F1611.