Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 2, No. 2, February : 2024

SECURING RPL BASED IOT NETWORKS: MACHINE LEARNING APPROACHES FOR DETECTING ROUTING ATTACKS

Dr M.V.R Jyothisree, K.V Srilakshmi Asharani, T.Lakshmi, K. Kavitha Lakshmi, Assistant Professor, Department of Computer Science and Engineering Maturi Venkata Subba Rao(MVSR) Engineering College, Hyderabad ,Telangana State, India jyothisree_cse@mvsrec.edu.in
Dr S.Sreekanth, Professor, Dept.of CSE(DS) Institute of Aeronautical Engineering, Dundigal, Hyderabad-500043 pranavsree_2000@yahoo.co.in

Abstract: The Internet of Things (IoT) represents a transformative concept aimed at enhancing ecosystem intelligence. However, its susceptibility to various attacks poses significant challenges, primarily due to its dynamic and diverse nature.This study focuses on Wireless Sensor Networks (WSNs), a foundational component of IoT, and investigates the vulnerabilities associated with routing attacks targeting the Routing Protocol for Low Power and Lossy Network (RPL).Furthermore, we propose a novel approach for the detection of three distinct types of attacks on RPL utilizing Machine Learning (ML) techniques. To create a realistic dataset of IoT-based features, we conducted simulations using Contiki OS. This dataset comprises a blend of data generated by both malicious and normal nodes within the IoT network. We employed the Cooja simulator to simulate four network scenarios: one representing normal behavior and three simulating various malicious attacks. These scenarios served as the basis for generating training and testing datasets for our machine learning phase.In the machine learning phase, we leveraged the WEKA platform to analyze the dataset and determine whether the behaviour exhibited is normal or malicious. This process involved the utilization of diverse classification algorithms. The results were noteworthy, with precision values consistently exceeding 96% across all cases.
Keywords: Internet of Things (IoT), Wireless Sensor Networks (WSNs), RPL attacks, Machine Learning, Classification algorithms, Cooja simulator, Contiki OS.

1. INTRODUCTION

The concept of the Internet of Things (IoT) is poised to usher in a new era of intelligence in the real world by enabling the seamless connection of objects without the need for human intervention. This paradigm has found application in a multitude of fields, including energy management, transportation systems, home and building automation, industrial processes, and healthcare. As a result, the world is now teeming with billions of sensors that gather data from physical objects and transmit it to the Internet.This proliferation of IoT devices necessitates the utilization of Wireless Sensor Networks (WSNs) and Radio Frequency Identification (RFID) systems, operating on various communication protocols such as WiFi, Bluetooth, Zigbee, and others. These protocols enable efficient communication among these diverse devices.

In this paper, we delve into the unique challenges posed by WSNs, which form the foundation of the IoT ecosystem. WSNs are characterized as ad-hoc networks comprising a vast number of low-cost, battery-powered sensor nodes. These nodes play a pivotal role in sensing and monitoring environmental conditions, including parameters like temperature, humidity, pressure, and movement. Typically, WSNs consist of sink nodes, sensor nodes, and clients. Sensor nodes, distributed randomly, collect data and transmit it to the network's base station, often referred to as the sink node. Data traverses multiple nodes through multi-hop routing before reaching its destination, which may be an end-user, satellite, or the Internet. According to the IEEE 802.15.4[1] specifications, WSNs employ 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks) and RPL (Routing Protocol for Low Power and Lossy Networks) at the Network layer, while protocols like CoAP (Constrained Application Protocol) or MQTT (Message Queuing Telemetry Transport) operate at the Application layer.

RPL, a distance vector routing protocol designed for constrained 6LoWPAN networks, facilitates mesh topology[2] communication among nodes. In this paper, we introduce a novel anomaly-based detection

method for identifying specific RPL attacks within WSNs. Our approach relies on classification algorithms, with the WEKA machine learning tool aiding in the determination of whether observed behavior is normal or malicious. We acquire datasets by monitoring various network scenarios simulated using Contiki-Cooja.

Our proposed solution offers several advantages, including the ability to detect three distinct types of attacks using different classification algorithms and ensemble learning. Unlike previous approaches, our method minimizes the burden on the wireless sensor network and optimizes power consumption. Furthermore, we prioritize lightweight detection by selecting only the most relevant features to uncover irregularities.

The structure of this paper unfolds as follows: In Section II, we delve into routing attacks targeting RPL. Section III provides an overview of related work in this domain. The implementation of our proposed system is detailed in Section IV, with results showcased in Section V. Finally, we conclude this paper in Section VI.

2.INTRODUCTION TO RPL PROTOCOL AND ITS NOTABLE CHARACTERISTICS

This section provides an overview of the RPL protocol, its topological aspects, and highlights well-known attacks associated with it .Routing protocols serve as the fundamental framework for communication within any network. However, in the context of wireless sensor networks, we encounter distinct challenges marked by resource constraints, including limitations in energy, memory, and computing power. These resource limitations render traditional routing protocols designed for wired and ad-hoc networks ill-suited for the unique characteristics of wireless sensor networks. Consequently, a novel routing protocol known as "IPv6 Routing Protocol for Low Power and Lossy Networks" (RPL) [3] has been introduced to address these challenges. The core concept behind RPL revolves around the creation of an efficient network topology that can adapt to the constraints of wireless sensor networks. This adaptation is achieved through the utilization of Destination Oriented Dynamic Acyclic Graphs (DODAGs). A DODAG is essentially a graph structure that organizes network nodes into a hierarchical arrangement centered on a single destination, which acts as the root node of the network. The hierarchy further includes leaf nodes.RPL facilitates the establishment of DODAGs through the deployment of specific ICMPv6 control messages. These messages play a pivotal role in the dynamic formation and maintenance of RPL DODAGs. By embracing the principles of DODAGs and leveraging ICMPv6 control messages, RPL empowers wireless sensor networks with the ability to gain real-time insights into the network's state, making it a critical component in the realm of IoT and wireless sensor communications.

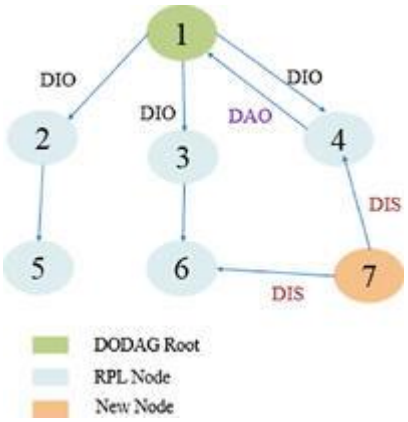

Figure 1: Overview of RPL Control Messages

The above Figure 1 illustrates the key control messages within the RPL protocol, namely the DODAG Information Solicitation (DIS), DODAG Information Object (DIO), and Destination Advertisement Object (DAO).

DODAG Information Solicitation (DIS):

In this communication process, each node plays a role in network access by initiating DIS messages. These DIS messages are broadcasted to neighboring nodes to request the transmission of DIO messages. The primary purpose of DIS messages is to facilitate the exchange of critical routing information required for the discovery of RPL instances.

DODAG Information Object (DIO):

DIO messages contain essential information vital for other nodes to identify and establish RPL instances. These messages are initially broadcasted by the root node of the network. Upon receiving DIO packets, nodes in the network employ the data within them to construct and update their routing tables. This dynamic process allows nodes to adapt and respond to changes within the network, ensuring efficient routing.

Dynamic Network Adaptation:

If a new node enters the network topology, a mechanism is in place to ensure network stability and adaptability. In such cases, all nodes within the network transmit DIO packets once more. This action aims to reconstruct a more stable and updated DODAG, aligning with the evolving network dynamics.

Destination Advertisement Object (DAO):

DAO packets serve the purpose of seeking permission to connect to a parent node within the network. When a node intends to establish a connection, it sends DAO packets to its chosen parent node, formally requesting consent. Subsequently, the parent node responds with a DIO-ACK (DODAG Information Object Acknowledgment) packet to acknowledge and accept the connection request.This intricate interplay of DIS, DIO, and DAO messages constitutes a fundamental aspect of RPL's functionality, ensuring efficient communication and adaptability within wireless sensor networks.

Types of Attacks Targeting RPL

The security landscape of the RPL protocol reveals three distinct categories of attacks [4][6], each with its unique characteristics and implications:

1. **HELLO Flooding Attack:**

The HELLO flooding attack[8] is notorious for creating communication bottlenecks within the network. It achieves this by inundating the channels with an excessive number of messages, specifically HELLO packets. These packets are typically exchanged between neighboring nodes to establish their presence and availability within the network. An attacker exploits this by employing an effective tool to flood the network with HELLO packets, convincing unsuspecting neighbor nodes to treat it as one of their own. Consequently, data transmission is redirected towards the malicious tool, congesting communication channels and causing the loss of critical data packets. In the context of the RPL protocol, this attack is executed through the transmission of malicious HELLO messages within DIS (DODAG Information Solicitation) packets.

2. **Version Number Modification Attack:**

The version number plays a pivotal role within each DIO (DODAG Information Object) message. Importantly, this number is incremented solely by the root node. In a version number modification attack[5][7], a malicious node manipulates this essential field, artificially increasing the version number. This seemingly minor alteration triggers an unnecessary reconstruction of the network graph, prompting the root node to reset its trickle timer and issue a new DIO packet. The ramifications of such an attack can disrupt network stability and resource efficiency.

3**. Black Hole Attack:**

The black hole attack[9][10] scenario involves one or more malicious nodes that intentionally discard data packets either partially or entirely, thus disrupting the normal flow of data within the network. These malicious nodes engage in deceptive practices, providing fraudulent routing information and positioning themselves as the optimal route toward the sink node. As a result, data is rerouted through the malicious node, impeding the network's usual data flow. An increase in the number of DIO messages exchanged among nodes serves as a tell-tale sign of this attack, primarily due to rank manipulation executed by the malicious node.

These three categories of attacks underscore the importance of security measures within the RPL

protocol, as they can severely impact the reliability and efficiency of communication within wireless sensor networks.

3 METHODOLOGY

Proposed System

Leveraging Machine Learning for Routing Attack Detection in WSNs.

The vast volume of network and sensory data generated by sensors within Wireless Sensor Networks (WSNs) presents a ripe opportunity for the application of machine learning techniques in intrusion detection. In this research endeavor, our focus was on identifying three distinct types of routing attacks: black hole attacks, hello flooding attacks, and version number modification attacks. Our approach concentrates on the establishment of a normal network profile, which serves as a baseline for comparison against observed network behavior.

To construct our detection system, we initiated the process by simulating four network scenarios employing the ContikiOS

and Cooja simulator. ContikiOS, recognized for its flexibility and lightweight design, is an open-source operating system tailored for sensor networks. It is implemented in C and serves applications across both commercial and non-commercial domains. A key utility within ContikiOS is Cooja simulator, a software simulator explicitly designed for wireless sensor networks.

The first scenario, aptly named the "normal network," served as our baseline reference, devoid of any malicious activity. This scenario featured a configuration consisting of one sink node and 24 sender nodes, strategically positioned across a 200x200-meter grid. Communication within this scenario was facilitated using the 6LoWPAN protocol, with RPL as the designated routing protocol.

The subsequent scenarios represented the malicious networks, where one of the 24 sender nodes was randomly designated to exhibit malicious behavior. These simulations allowed us to evaluate the performance of our intrusion detection system under adversarial conditions.

Following the simulation phase, we transitioned to pre-processing and feature selection as we embarked on constructing our dataset.

Pre-processing and Feature Selection

In this section, we outline our approach to selecting and extracting the appropriate attributes required for constructing datasets used in the machine learning phase.

Data Extraction

The choice of data features significantly influences the performance of our machine learning models. Selecting the right attributes is a pivotal challenge in achieving effective detection. To acquire data relevant to the detection of malicious nodes, we implemented continuous network traffic capture with observation windows of a fixed duration, denoted as "t" (in our case, t = 5 seconds). Messages, including DIS, DIO, and DAO, were captured using the "Radio messages" tool in COOJA, enabling the generation of PCAP files for subsequent analysis using Wire-shark.

Our selection of data attributes was informed by an understanding of the various attacks. We considered the following metrics:

Number of DIS Messages: This attribute quantifies the number of DIS messages exchanged among nodes within a 5-second window.

Number of DIO Messages: Similar to DIS, this metric counts the number of DIO messages exchanged among nodes within a 5-second window.

Number of DAO Messages: This attribute captures the count of DAO messages exchanged among nodes during a 5-second interval.

Version Number Modification: The version number, a crucial field in DIO packets, represents the version of a DODAG graph. Ideally, this field remains unchanged by nodes other than the root node, which is responsible for incrementing it. To detect version number modification attacks, we introduced the attribute "version_modification," set to 0 if the version number remains stable and 1 if it's modified.

Rank Value Average: The rank of a node, as indicated in DIO packets, reflects the node's position within a DODAG relative to the root. Some attacks, such as the black hole attack, involve malicious

nodes artificially lowering their rank to manipulate the DODAG. To assess this, we calculated the rank average from DIO messages within a 5-second window.

Power Consumption: Evaluating the impact of attacks on energy consumption is critical. We considered the average power consumption across all motes, a metric facilitated by the Power Tracker tool in COOJA.

Sample data captured from simulations, monitoring both normal and malicious behaviors (specifically, black hole, Hello flooding, and version modification attacks), are presented in Tables 1, 2, 3, and 4. Notably, the presence of malicious activity introduced by node 11 in the malicious networks resulted in network topology instability. For the network subjected to a black hole attack, a distinct increase in the number of DIO messages was evident during each 5-second interval.

Table 1 : Trace of the normal scenario

DIS_nbr	DIO_nbr	DAO_nbr	POWER_CONSP%	Rank_avg	Version_modification
507	20	0	1.52	120	0
0	221	1570	4.19	441.11	0
20	639	2175	6.79	762.42	0
0	1548	615	7.80	814.15	0
0	509	636	7.12	915.17	0
0	1029	1211	6.73	901.12	0
0	615	356	7.3	1132.49	0

Table 2 : Trace of Black hole attack scenario

DIS_nbr	DIO_nbr	DAO_nbr	POWER_CONSP%	Rank_avg	Version_modification
507	101	0	1.54	120	0
0	1153	2313	3.48	401.12	0
20	1432	2465	7.43	612.13	0
0	1505	2154	8.05	748.45	0
0	870	1151	10.32	701.25	0
0	1346	1621	11.25	843.11	0
0	1916	1202	11.6	930.17	0

Table 3: Trace of Hello Flooding attack scenario

DIS_nbr	DIO_nbr	DAO_nbr	POWER_CONSP%	Rank_avg	Version_modification
1165	20	0	8.10	120	0
932	125	773	9.52	413.12	0
928	163	1036	11.15	774.34	0
619	812	725	11.56	779.74	0
845	896	1220	11.59	931.13	0
520	425	2215	12.05	907.34	0
511	861	210	12.10	1001.52	0

Table 4: Trace version modification attack scenario

DIS_nbr	DIO_nbr	DAO_nbr	POWER_CONSP%	Rank_avg	Version_modification
507	20	0	1.52	120	0
0	1035	1004	3.13	435.73	0
20	2054	2301	7.01	732.14	0
0	2470	744	9.07	886.71	0
0	1012	1013	8.41	941.13	1
0	701	722	7.10	974.16	1
0	740	751	7.87	1152.37	1

When compared to normal scenario, we observe the following from

Hello Flooding Attack:

A decrease in the rank average is noted in comparison to a normal network scenario.

The attack involves broadcasting a large number of DIS packets (Neighbour Discovery Protocol packets).
This behaviour is reflected in the dataset.

Version Number Modification Attack:

The dataset shows that the attribute version_modification is set to 1, indicating a modification in the version number.
This type of attack is likely to disrupt the normal functioning of the system.

Power Transmission Consumption:

Malicious activity has led to an increase in energy consumption throughout the simulation in all attack scenarios.
The power consumption of each node in the active mode (ON) in the transmission radio (Tx) and the reception radio (Rx) is presented in Figures 2, 3, 4, and 5.

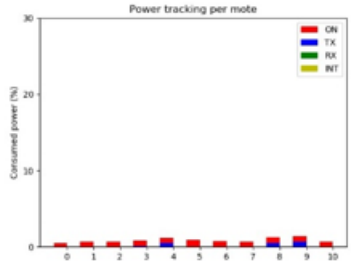

Fig 2 : Power tracking per mote in the normal scenario

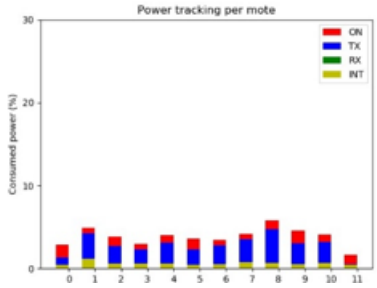

Fig 3: Power tracking per mote in the Black hole scenario

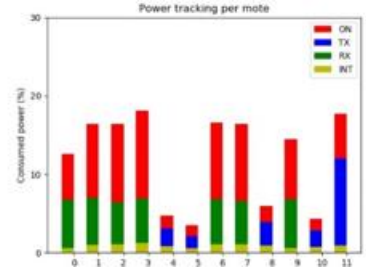

Fig 4 : Power tracking per mote in the Hello flooding scenario

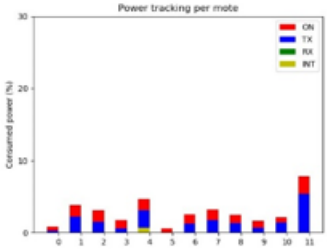

Fig 5 : Power tracking per mote in the version modification scenario

The figures illustrate the impact of the attacks on power consumption.In summary, the attacks discussed have negative effects on the network, causing disruptions in the form of decreased rank averages, version number modifications, and increased power consumption. The figures provided help visualize the impact on power consumption in different attack scenarios.

4 RESULTS AND PERFORMANCE

In our case, we have done the process of evaluating classification algorithms using Weka, particularly focusing on metrics such as the confusion matrix and precision. Data used for the learning phase were captured and recorded by the simulator and they are divided into two sets. The first phase is training set is used to train the classification model. The algorithm learns patterns and relationships from this set.

Second phase is test set , which is used to evaluate the model's performance on new, unseen data. It helps to assess how well the model generalizes to new instances.

Confusion Matrix:

True Positive (TP): Instances correctly predicted as positive.

True Negative (TN): Instances correctly predicted as negative.

False Positive (FP): Instances wrongly predicted as positive.

False Negative (FN): Instances wrongly predicted as negative.

The confusion matrix provides a comprehensive view of the model's performance, allowing you to assess how well it is classifying instances.

A fundamental evaluation technique applicable to all types of classification problems. Following table Displays four key values: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).Provides a comprehensive understanding of the relationship between these values.

	Predicted Positive	**Predicted Negative**
	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Precision:

- Precision measures the proportion of predicted intrusions that are true intrusions.
- A high-precision Intrusion Detection System (IDS) aims to minimize false alarms.

Recall:

- Recall, also known as Sensitivity or True Positive Rate, indicates the percentage of actual intrusions correctly predicted.
- A desirable IDS should exhibit a high recall value.

F-Measure:

- The F-Measure, or F1 Score, is a harmonic mean of precision and recall.
- It provides a balanced assessment of the classifier's performance.

Classification Algorithms and Results:

Tables below present the results of the learning phase with Weka using three classification algorithms: Support Vector Machine (SVM), Naïve Bayes, and Decision Tree.

Table 6: Results and performance of SVM classification algorithm

Class/Metric	**True Positive rate**	**False Positive rate**	**Recall**	**F-Measure**	**Precision**
Normal	1	0.05	1	0.952	0.909
Black hole	0.9	0.014	0.9	0.923	0.947
Hello Flooding	1	0	1	1	1
Version Modification	0.9	0	0.9	0.947	1
Weighted average	0.956	0.02	0.956	0.955	0.958

Table 7: Results and performance of Naïve Bayes classification algorithm

Class/Metric	True Positive rate	False Positive rate	Recall	F-Measure	Precision
Normal	0.967	0.017	0.967	0.967	0.967
Black hole	0.975	0.036	0.975	0.929	0.886
Hello Flooding	1	0	1	1	1
Version Modification	0.9	0	0.9	0.947	1
Weighted average	0.961	0.013	0.961	0.961	0.964

Table 8: Results and performance of Decision tree classification algorithm

Class/Metric	True Positive rate	False Positive rate	Recall	F-Measure	Precision
Normal	1	0.05	1	0.952	0.909
Black hole	0.925	0.007	0.925	0.949	0.974
Hello Flooding	1	0	1	1	1
Version Modification	0.9	0	0.9	0.947	1
Weighted average	0.961	0.018	0.961	0.961	0.964

Observations:

In our case, the results demonstrate high performance across all classification algorithms. Notably, a very low false positive rate and high precision were achieved, indicating the effectiveness of the selected algorithms for intrusion detection.

5 CONCLUSION

In conclusion, our research focused on addressing the vulnerabilities of the RPL protocol within WSN networks by detecting three specific types of attacks: black hole, Hello flooding, and version number modification. Recognizing the resource limitations of WSN devices, our goal was to develop robust security countermeasures to safeguard the integrity and performance of these networks.

Our approach involved simulating four network scenarios using the Contiki Cooja simulator, differentiating between a normal scenario and those with specific attacks introduced by a malicious sensor node. We meticulously built training sets to facilitate the learning phase of our detection method. Emphasizing efficiency and accuracy, we employed the Weka feature selection tool to identify the optimal attribute set crucial for successful classification.

In the learning phase, a combination of single classification models and an ensemble learning model was leveraged, resulting in an impressive precision value exceeding 96% across all scenarios. This robust performance indicates the effectiveness of our proposed approach in accurately identifying and mitigating the specified attacks on the RPL protocol.

Looking ahead, our future work aims to enhance the sophistication of our intrusion detection system (IDS) by simulating diverse scenarios with varying rates of malicious and normal nodes, along with an increased number of nodes. We recognize the need to broaden the scope of our investigation by incorporating additional routing metrics such as delay, hop count, throughput, and bandwidth. This expanded analysis will contribute to a more comprehensive understanding of potential threats and strengthen the overall security posture of WSN networks. In summary, our research not only highlights the vulnerabilities of the RPL protocol in the face of specific attacks but also proposes a promising detection mechanism that demonstrates high precision. As we continue to evolve our work, we anticipate that these findings will contribute significantly to the ongoing efforts in securing WSN networks against emerging threats.

REFERENCES

1. Granjal, J., Monteiro, E., & Silva, J. S. (2015). Security for the internet of things: a survey of existing protocols and open research issues. IEEE Communications Surveys & Tutorials., 17(3), 1294–1312.
2. Kfoury, E., Saab, J., Younes, P., &Achkar, R. (2019). A self organizing map intrusion detection system for RPL protocol attacks. International Journal of Interdisciplinary Telecommunications and Networking (IJITN)., 11(1), 30–43.
3. Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., & Levis, P., et al. (2012). RPL: IPv6 routing protocol for low-power and lossy networks.
4. Wallgren, L., Raza, S., & Voigt, T. (2013). Routing attacks and countermeasures in the RPL-based internet of things. International Journal of Distributed Sensor Networks., 9(8), 794326.
5. Pongle, P., Chavan, G. A., & survey: Attacks on RPL and 6LoWPAN in IoT. In,. (2015). International conference on pervasive computing (ICPC). IEEE, 2015, 1–6.
6. Anderson, J. P. (1980). Computer security threat monitoring and surveillance.Introduced to network system by Heberlein in the year of 1990.
7. Heberlein, LT., Dias, GV., Levitt, KN., Mukherjee, B., Wood, J., & Wolber, D. (1989). The intrusion 1843IoT Routing Attacks Detection Using Machine Learning Algorithms 1 3detection system (IDS) is an active process that analyzes network activity and system by the gathering of tools, methods, and resources to identify and detect intruders or malevo-lent activities.
8. Gupta, A., Pandey, OJ., Shukla, M., Dadhich, A., Mathur, S., & Ingle, A. (2013). Computational intelligence based intrusion detection systems for wireless communication and pervasive computing networks.
9. Yavuz, F. Y., Devrim, Ü., & Ensar, G. (2018). Deep learning for detection of routing attacks in the internet of things. International Journal of Computational Intelligence Systems., 12(1), 39.
10. Ioulianou, P., Vasilakis, V., Moscholios, I., & Logothetis, M. (2018) A signature-based intrusion detection system for the internet of things. Information and Communication Technology Form. .