SN Computer Science (2024) 5:249
https://doi.org/10.1007/s42979-023-02568-5

ORIGINAL RESEARCH

Unveiling the Energy-Based Validation and Verification (EVV) Method for Perceiving and Averting Rank Inconsistency Attacks (RIA) for Guarding IoT Routing

K. Ramu[1] · N. Gomathi[2] · Sanjay Kumar Suman[3] · P. Joel Josephson[4] · M. Vadivukarassi[5] · Narasimha Swamy Lavudiya[6] · L. Bhagyalakshmi[7]

Received: 24 August 2023 / Accepted: 17 December 2023

Abstract

The need for connected devices is increasing rapidly in response to the expanding demand for internet connectivity and related services. Due to the high need for IoT applications, new methods and tools have been developed. RPL is a protocol suite used in IoT networks that facilitates communication and movement between nodes. Commercial implementation of the Internet of Thing (IoT) is hampered by a small number of security problems, despite the fact that there are many benefits to adopting IoT. The EVV approach is what the authors suggest utilizing in order to locate the rank node in an RPL topology that has been incorrectly assigned. A rank value is a numerical representation of each node's position in the tree in relation to the root node. To identify the malicious hub, the proposed EVV method is implemented at the root hub. Attackers in RPL use the energy meter to their advantage and launch a variety of attacks by moving up the RPL directed attack graph (DODAG). This work proposes an energy-based intrusion detection module to identify these attacks and the malicious nodes. Against a rank attack, also known as a rank inconsistency attack (RIA), this EVV module can hold its own. Select network parameters are used to evaluate the proposed EVV method against the current systems. Thus, compared to prior methods, the EVV significantly decreased the time required for both attacker identification and network convergence.

Keywords IoT · EVV · RIA · DODAG · RPL

This article is part of the topical collection "Machine Intelligence and Smart Systems" guest edited by Manish Gupta and Shikha Agrawal.

✉ Sanjay Kumar Suman
Prof.dr.sanjaykumarsuman@gmail.com

K. Ramu
K.ramu147@gmail.com

N. Gomathi
gomathin@veltech.edu.in

P. Joel Josephson
pjoelece@mrec.ac.in

M. Vadivukarassi
vadivume28@gmail.com

Narasimha Swamy Lavudiya
narasimhaswamyrl@gmail.com

L. Bhagyalakshmi
Prof.Dr.L.Bhagyalakshmi@gmail.com

[1] IBM KYNDRYL LLC, Chennai, TN, India

[2] Department of CSE, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai, TN, India

[3] Department of ECE, St. Martin's Engineering College, Secunderabad, Telangana, India

[4] Department of ECE, Malla Reddy Engineering College, Secunderabad, Telangana, India

[5] Department of Computer Science and Engineering, St Martin's Engineering College, Secunderabad, Telangana, India

[6] Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, AP, India

[7] Department of ECE, Rajalakshmi Engineering College, Chennai, TN, India

Published online: 03 February 2024

Introduction

Security in the IoTs is crucial for private data exchange between sensor nodes in an IoT network. To ensure that IoT devices can communicate securely, it is necessary to create a system for safe routing between sensor nodes. In IoT, hackers use limited devices to conduct a variety of routing attacks such the rank attack, selective forwarding, sinkhole attack, denial of service, sybil, wormhole attack, hello flood, and so on [1]. However, the solutions provided by these methods for detecting and identifying routing attacks in the IoT are insufficient.

Many IoT devices connect to low-power, lossy networks (LLNs), which severely restrict their adaptability and make it difficult for them to use sophisticated cryptographic security mechanisms. As a result, a strong security mechanism among IoT sensor nodes requires consideration of the requirements of constrained devices [2]. IPv6 could make it easy to connect everyday objects to the Internet of Things.

The proliferation of Internet of Things (IoT) technology, which may be applied in numerous contexts and is already seeing widespread adoption. Security during adoption of the Internet of Things is challenging because of the wide variety of IoT devices. Also important [3] is the use of end-to-end (E2E) encryption for all data sent between IoT nodes. Because of this, safeguards for the confidentiality and integrity of transmitted data must be implemented on both ends of the connection. Security methods already in place are necessary for end-to-end message encryption in the IoT.

Due to resource limitations, the network is vulnerable to many different types of assaults from both within and outside. In particular, internal attacks open up more entry points and hinder the efficiency of the network [4]. The detection and defence against the rank inconsistency assault are difficult tasks. Therefore, a standardized security mechanism is required for IoT to identify and counteract such threats.

To trick its neighbours, an adversarial node creates a fake ranking metric. Since many IoT devices run on batteries, power is a precious commodity for isolated nodes [5]. When a node's energy supply or other resources are low, it is disconnected from the network until it can be evaluated for readmission [6].

Most IoT nodes have limited battery life, thus they must use some sort of energy-efficient strategy. There may be further delays in packet delivery [7] if a node has a weak network connection, limited bandwidth, or slow processing performance.

Depending on the use case, insecure RPL control communications could be used as security primitives. When nodes join an RPL instance, they already have the necessary keys pre-installed to process and secure RPL messages in pre-installed mode [8]. In authenticated mode, the nodes have the required keys pre-installed just like they do in pre-installed mode. These keys can only be used to generate a leaf node for an RPL instance.

The suggested method to determine rankings, EVV focuses on the energy metric. The chosen parent determines each child node's Child Rank (CR). Preferred parents are aware of their children's beginning energy and packet transmission details [9]. By comparing the kid node's initial and consumed energy, the preferred parent can determine the node's available energy. In the verification step, the Self Rank (SR) and the Criticality Rating (CR) are utilized to pinpoint the offending node. Rank data from nearby nodes is checked and double-checked. The ranking of child nodes may be trusted since it is double-checked against other pieces of ranking data [10]. The malicious node removal mechanism expels the identified attacker node from the DODAG. The essential function of an intrusion handling mechanism is the removal of malicious nodes. Either local repair or global repair can be used to carry out the process. As a result, the EVV decreased the time it took to identify attackers and for the network to converge [11].

The Related Work Done

Researchers in this study modelled and shared the outcomes of many RPL-based assault scenarios. These findings were shown. The rank assault, local repair attack, neighbour attack, and DODAG Information Solicitation (DIS) attack [12] are only some of the novel internal threats to RPL that were considered in this study. Important performance measures like end-to-end delay, delivery ratio, and control overhead were broken down in this article to show how such attacks affected the network. This article delves into the inner workings of four distinct attacks on RPL operations—the rank attack, the local repair attack, the neighbour attack, and the DIS attack. Among these assaults was the rank one. The simulation results showed that internal threats have a significant impact on RPL network performances, such as decreasing the delivery ratio, increasing the end-to-end delay, and developing additional control overhead, which depletes network resources [13].

The simulation results showed that internal threats have a major effect on the performance of RPL networks. The authors' attention was concentrated on the robustness of

RPL's routing. Trust Anchor Interconnection Loop (TRAIL) is a service proposed by the authors [14] to increase authentication security in RPL topologies. It was a flexible approach to checking RPL topologies. Each node checked its ascending path to the root to prevent topological issues and detect rank faking. Nodes and messages in the network required less time and energy thanks to TRAIL [15].

The authors' suggested specification-based Intrusion Detection System (IDS) [16] is able to recognize both the rank assault as well as the local repair attack. The authors constructed an intrusion detection system (IDS) using a network monitor as its key component and outlined its monitoring operations. They did this by utilizing an RPL finite state machine at each monitor node.

The 6LoWPAN network uses a number of different security techniques. However, these would only prevent attacks from the outside on the 6LoWPAN. When individual nodes are compromised, however, they turn into the internal adversary. Attacks including sinkhole attacks, selective forwarding attacks, rank assaults, black hole attacks, and more can be launched from hacked nodes [17]. These may cause problems in determining alternate routes or forwarding data. An intrusion detection system can be used to keep an eye out for any hiccups in the system and sound the alert so that future attacks can be avoided. The authors analysed the simulation results to determine how long it takes for the network to reach a stable, loop-free state [18]. The Directed Acyclic Graph (DAG) experienced a routing loop as a result of the node's rank increase operation. Loop avoidance, loop detection, and loop prevention were the mechanisms evaluated. Authors described how varying rank value improved network performance. After a certain amount of time had passed, the node in question would raise its rank until it was at par with its neighbours' greatest rank [19].

Researchers came up with a new rank computation approach and a loop-free local route repair mechanism so that they could eliminate the problem of routing loops in RPL. The results of the simulation show that the loop free routing protocol RPL greatly beat the baseline routing protocols in terms of packet delivery rate, end-to-end packet delay, and routing overhead [20]. This can be seen by looking at the results of the simulation. To counter the rank-increase attack, the authors developed a novel node-count security technique [21]. Due to a Rank Increased Attack (RIA), a resource-draining loop has formed between the network's nodes. The other nodes in the architecture were also impacted, and the IoT devices' lifespan was shortened as a result [22].

The focus of the authors was on the safety of RPL's routing. Trust Anchor Interconnection Loop (TRAIL) is an authenticating security service suggested for RPL architecture. It was an open-ended method for verifying RPL topologies. It recognized rank spoofing and stopped topological errors by having each node verify its upward path to the root. TRAIL also reduced the amount of time and energy used by the network's nodes and messages [23].

The authors proposed an objective function in RPL based rank attack for low power and lossy networks. The network's Objective Function (OF) was used to determine the optimal path to the DODAG root. The intended rank attack provided a false routing path, delaying network throughput and increasing transmission and reception times [24]. The authors presented a hybrid specification-based intrusion detection system (IDS) with centralized and distributed modules placed on the sink and RPL nodes, respectively, to prevent nodes from choosing an intruder as their successor. The proposed technique also eliminated the risk of an attacker masquerading as a time source to throw off the synchronization of IPv6 Time Slotted Channel Hopping (6TiSCH) networks. In conclusion, extensive modelling proved that the suggested IDS effectively protected RPL topologies with low network management overhead.

Internet of Things Intrusion Detection over 6LoWPAN (INTI). The goal of INTI is to detect, isolate, and avoid the negative impacts of attack sinkholes in routing. By examining the actions of each node, it integrates watchdog, reputation, and trust tactics to identify malicious actors. The attackers' ability to assume a variety of roles in the network as free node, member node, and leader node is taken into account by the INTI system. It allows the network to self-organize and fix itself if something goes wrong. The first characteristic emphasizes devices working together in harmony with the established network topology. The second feature enables the identification of potentially malicious nodes and the regrouping of clusters to keep the network operational.

The Objective of the Work

In this research work, we will focus on developing a safe routing framework that uses validation and verification as its foundational procedures. This method is employed to identify malicious nodes that have artificially inflated rank values. The methods to accomplish the goal are as follows:

- To offer a protected routing system for the IoTs.
- To quickly explain the various techniques that make up the architecture.

The Projected Work

The EVV methods under consideration mostly focus on attacking the system from within. The typical security solutions could be optimized for lightweight design to accommodate the minimal resource restrictions imposed by the devices. The Rank Inconsistency Attack (RInA), which generates a path that is not ideal and has the potential to result in a loop, will be the focus of this investigation, and its objective will be to locate it and put a stop to it. As a consequence of this, RInA has an impact on the performance of the network, reducing the percentage of successfully delivered packets while simultaneously increasing packet latency and lowering throughput. As a result of this work, energy-based verification and validation (EVV) approaches have been developed for the Internet of Things (IoT) to detect and avoid rank inconsistency attacks. This EVV (Energy-based Validation and Verification) method integrates three processes to analyse rank inconsistencies and offer protected communication in an IoT setting. Those processes are determining ranks, providing evidence, and removing malicious nodes. The method's intent is to safeguard IoT networks that use RPL-based routing protocols.

Figure 1 depicts the overall design and workflow of the EVV method. Different methods are presented, each with their own set of steps and objectives. The ultimate goal of this method is to enable secure connection and identify rogue nodes that cause rank irregularities through the use of validation and verification procedures. The proposed method is applied to the IoT to detect the rank inconsistency attack. In an IPv6-based low-power, lossy network, the rogue node can be identified by its energy consumption. The EVV architecture is outlined in the next section.

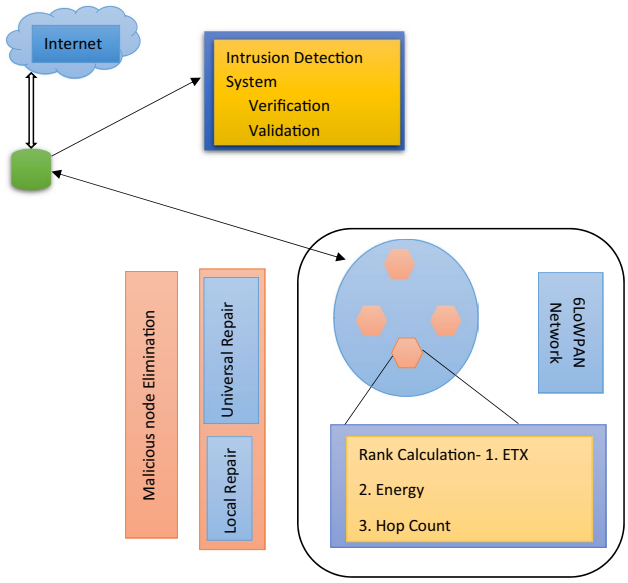

Fig. 1 The proposed EVV technique

As a means of preventing network loops, the RPL takes into account each node's rank metric. When a DODAG is built, each node keeps track of its position in relation to each other node. In RPL, the rank value always falls as you go up and rises as you go down. In the RPL structure, children choose their parents based on their rank value. In this case, not one but two ranks—the sr (self-rank) and the pr (parent-rank)—must be determined. The rank of a child node is calculated by the SR. Based on the definition of its preferred parent's rank in Eq. (1), a child node can determine its own rank. In order to determine rankings, the suggested work makes use of the energy metric. DIO messages are sent from the parent node to all of the child nodes. Parent node identifier, rank, and energy are all included in the DIO message. The children determine their own rank based on the rank of their chosen parent. After determining their rank, children nodes communicate with their selected parents by DAO message. The DAO message includes the node rank and current energy of each child node. A preferred parent node can calculate the starting energy of a child node. Each node starts with some amount of energy, which is depleted as data packets are sent and received. The Minimum Hop Rank Gain from one node to another is calculated using the DODAG root.

$$\mathrm{sr} = \mathrm{pr} + \mathrm{Rank}_{\mathrm{increase}} \tag{1}$$

$$\mathrm{Rank}_{\mathrm{increase}} = \mathrm{Credit} + \text{min hop rank gain} \tag{2}$$

$$\mathrm{Credit} = \frac{\text{Energy available}}{\text{Energy prior}} \tag{3}$$

$$\mathrm{cr} = \mathrm{pr} + \mathrm{Rank}_{\mathrm{decrease}} \tag{4}$$

$$\mathrm{Rank}_{\mathrm{decrease}} = \text{Energy intake} + \text{min hop rank gain} \tag{5}$$

where,

$$\begin{aligned}\text{Energy intake} = {} & (\text{packets sent} \times \text{needed energy}) \\ & + (\text{ideal time} \times \text{ideal need energy})\end{aligned} \tag{6}$$

To become a compromised node in an IoT network, an attacker node can aim for the rank of any given node or its neighbours. The compromised node then initiates a barrage of attacks and broadcasts false routing data. There is variance in the rankings because of how this crucial issue is perceived across the network. As a result, it is crucial to identify the rank inconsistency in RPL-based networks and make the necessary adjustments. Approach 1 describes the full approach for detecting rank inconsistencies.

Algorithm 1 Identify and alleviate the rank discrepancy

Input: N numeral of nodes

Process: Validate and authenticate the node's rank and energy

Output: Recognition and Eradication of Attacker node

1. Start
2. For whole node N {
3. DODAG create ( ){ For CN (i) ¬ 1 to N // CN-Child Node {Parent assortment ( )//Selecting the utmost parent grounded on rank {
4. Rank () //Computing the rank cost { Credit = Obtainable Energy / Preliminary Energy Rank gain = credit + Min Hop Rank gain SR = Parent Rank + Rank gain // Self Rank Deviousness
5. Energy Intake (EC) = (packets sent × needed energy) + (ideal time × need energy)
6. Rank gain = EC + min hop rank gain
7. Cr (Child Rank) = pr + Rank drop}}
8. End for}}
9. Whole Transaction computation ( ) // packets count sent by the child nodule {
10. For Ppi ¬ 1 to n // packets count acknowledged
11. Ppi + +}
12. Compute cr ( ) {

For CNi ¬ 1 to n //child nodules count {

13. Obtain DIO
14. For DIOj ¬ 1 to m // DIO obtained count {
15. End for j}
16. End for i}}
17. E(Ppi) = Ppi * energy required per packet
18. If CURi = = RCRi //fake rank documents
19. Attacker node documents ( )
20. For NR = 1 to n
21. If PRi > NRi + Ai
22. A + +
23. For NRi, i = 1 to n
24. If NA > NAth
25. If A is close to root //ring nurture to the networks
26. Universal Repair ( ) Rebuild the DODAG;
27. Else Native Repair ( ) Darning specific malicious node;
28. End if
29. End if

This helps to ensure that nodes are consistently ranked across the network. When the BR detects an inconsistency in the node's rank, it can trigger a false alarm. In addition to detecting rank inconsistencies, the EVV method can tell the difference between the energy levels of legitimate and invalid nodes, hence preventing errors in the RPL DODAG. To identify the malicious node spreading misinformation, the EVV algorithm makes advantage of the energy credit system.

The EVV algorithm relies on two things: (i) verifying the accuracy of each node's reported rank and energy, and (ii) comparing those values to those of their neighbours. When these two factors combine to provide inconsistent RPL results, the 6BR steps in to rectify the situation. The 6BR notifies its neighbours of the incorrect data, including the erroneous node's identifier, parent list, and version number. The neighbour nodes include the 6BR data into their routing table updates. The 6BR will unwhite list a node once it determines that it has rank inconsistency or energy inconsistency. At first, the 6BR notifies its neighbours about the attacker node and adds the defective node to the white list. Repeated malicious detection of the same node will result in removal from the white list by 6BR.

Result and Discussion

The suggested method The EVV is tested in the IoT-oriented Contiki Cooja simulator. The EVV's main purpose is to identify and prevent routing attacks, such as the RInA. The proposed method, called EVV, is implemented at the stem cell level. The modules can confirm the node's actions through the substantiation phase.

Intrusion Detection Systems (IDS) are commonly employed to monitor wireless sensor network node activity. Each node in a network logs all of its control messages and uses the EVV mechanism implemented in RPL for routing. In addition, the EVV includes protections against blackhole attacks, selective forwarding attacks, and sinkhole attacks on the RPL network. After presenting the experimental setting, this section presents the assessment of EVV. The settings for the simulator are shown in Table 1.

Convergence Latency

The EVV calculates the time required for the network to converge using a number of nodes. The convergence latency is the time it takes for the RPL network to converge

Table 1 Network simulation considerations

Constraints	Explanation
Nodes count	50, 100, 150, 200, 250
Replication area	1000 × 1000 m
Node organization	Random, grid
Operating System	Contiki
Radio mediocre	UDGM-distance damage
Emulator	Cooja
Routing protocol	RPL
Network protocol	IP based
Sensor node types	Sink, sender, border router
Packet analyser	Wireshark

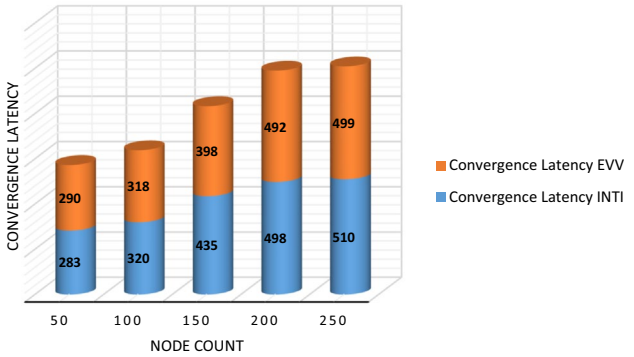

Fig. 2 Convergence latencies comparison of the proposed algorithm with the existing approach

Table 2 Convergence latencies comparison of the proposed algorithm with the existing approach

S. no	Node count	Convergence latency	
		INTI	EVV
1	50	283	290
2	100	320	318
3	150	435	398
4	200	498	492
5	250	510	499

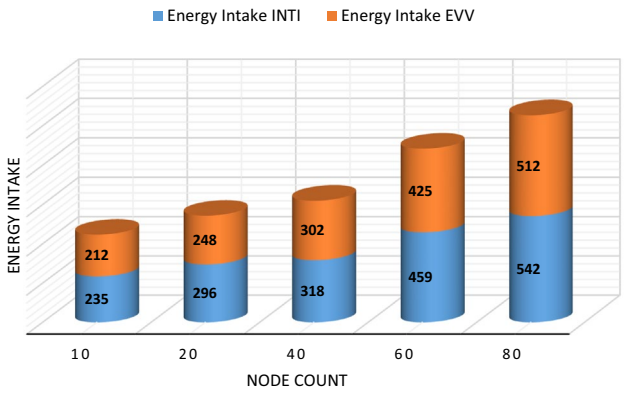

Fig. 3 Energy intake comparison of the proposed algorithm with the existing approach

again after the attacker node has been corrected. Convergence latencies are compared to the current INTI method in Fig. 2. In this case, 50, 100, 150, 200, and 250 nodes are used in the measurement (Table 2).

This finding demonstrates that the packet transmission latency can be minimized with the help of the proposed method. The suggested method measures how long it takes to identify an attacker for various network sizes. Ten, twenty, forty, and sixty nodes were used in the simulation. The proposed technique takes 290 ms to identify an attacker at a ratio of 0.2% over 60 nodes.

Table 3 Energy intake comparison of the proposed algorithm with the existing approach

S. no	Node count	Energy intake	
		INTI	EVV
1	10	235	212
2	20	296	248
3	40	318	302
4	60	459	425
5	80	542	512

Energy Intake

The EVV method is used to identify the offending node after an attack has been initiated. Energy usage versus attacker ratio (%) is displayed in Fig. 3. A simulation was run with 100 nodes, 40 percent of which were malicious (Table 3).

Attacker Documentation Delay

The EVV is put through its paces using a delay parameter for identifying the attacker and is compared to the LRPL and the INTI. Attacker documentation delays are depicted in Fig. 4. The effectiveness is measured using 10, 30, 50, and 100 nodes (Table 4).

The proposed method yields a decrease in the time it takes for a packet to transmit. The packet delay, measured in seconds, is used in the simulation. In comparison to the RIAIDRPL's 40 s of packet delay at the same attacker ratio of 0.2%, the current method, LRPL, takes 50 s of packet

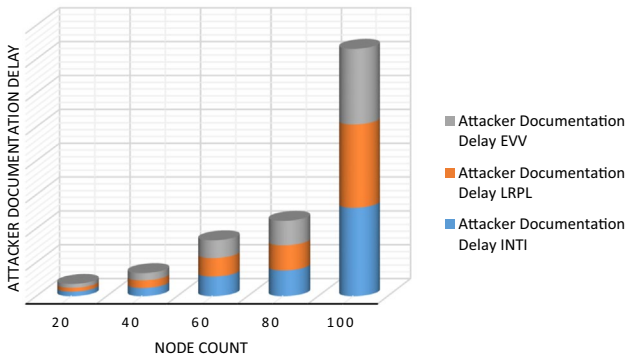

Fig. 4 Attacker documentation delay comparison of the proposed algorithm with the existing approach

Table 4 Attacker documentation delay comparison of the proposed algorithm with the existing approach

S. no	Node count	Attacker documentation delay		
		INTI	LRPL	EVV
1	20	135	115	112
2	40	243	226	215
3	60	582	549	520
4	80	763	742	723
5	100	2614	2459	2234

delay. The packet delay for the LRPL is 186 s for a 0.4% attacker ratio, while the proposed technique is just 112.

The simulation outcome of the suggested method By spotting and decreasing false rank attacker nodes, EVV improves overall network security. The network size and convergence delay time (in milliseconds) are two measures of performance. Different densities of nodes conduct the experiment.

The proposed strategy has a lower false positive rate across the board for different sized networks. This demonstrates that the suggested method successfully identified malicious nodes with an accuracy of about 96%.

Conclusion and Future Scope

The need for connected devices is increasing rapidly in response to the expanding demand for internet connectivity and related services. New technologies and methods have been developed in response to the increasing need for IoT applications. RPL is a protocol suite used in IoT networks that facilitates communication and movement between nodes. Despite the many benefits of deploying IoT, a small number of security concerns hampers commercial deployment. To identify the erroneous rank node in an RPL topology, the authors recommend using the EVV method. Each node's position relative to the root node is represented by a rank value. To reroute data packets or modify the data, an attacker node falsely claims a lower rank value. To identify the malicious hub, the proposed EVV method is implemented at the root hub. Attackers in RPL use the energy meter to their advantage and launch a variety of attacks by moving up the RPL directed attack graph. This work has presented an energy-based intrusion detection module to identify rogue nodes and protect networks from future attacks. Against a rank attack, also known as a rank inconsistency attack (RInA), this EVV module can hold its own select network parameters are used to evaluate the proposed EVV method against the current systems. Thus, compared to prior methods, the EVV significantly decreased the time required for both attacker identification and network convergence.

Only attacks based on rank inconsistency are the focus of the proposed study. However, a wide range of other problems, such as topological assaults, poor performance, and heavy network traffic, affects RPL. In the future, we can provide methods to address such problems. In order to improve the IoT, it is necessary to create and identify various forms of routing attacks due to the large number of interconnected devices. In order to keep tabs on the system, it would be helpful to group the devices together according to their technology. In order to prevent Wormhole attacks, the security of the RPL network must be strengthened while clustering.

Data Availability Data sharing is not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of Interest First author and second author declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with animals performed by any of the authors.

References

1. Al Sawafi Y, Touzene A, Day K, Alzeidi N. Toward hybrid RPL based IoT sensing for smart city. In: International conference on Information Networking (ICOIN), IEEE. 2018. p. 599–604.
2. Christian C, Poplade D, Nogueira M, Santos A (2015) Detection of sinkhole attacks for supporting secure routing on 6lowpan for internet of things. In: Integrated Network Management (IM), IFIP/IEEE international symposium on IEEE.
3. Jorge G, Monteiro E, Silva JS. Security for the internet of things: a survey of existing protocols and open research issues. IEEE Commun Surv Tutor. 2015;17(3):1294–312.
4. De La Cruz J (2017) Analysis of different routing Attacks against WSN's using RPL with Contiki OS and Cooja Simulator. Dissertation.
5. Pallavi S, Sarangi SR. Internet of things: architectures, protocols, and applications. J Electr Comput Eng. 2017.
6. Kumar P, Baliyan A, Prasad KR, Sreekanth N, Jawarkar P, Roy V, Amoatey ET. Machine learning enabled techniques for protecting wireless sensor networks by estimating attack prevalence and device deployment strategy for 5G networks. Wirel Commun Mobile Comput. 2022;2022:5713092. https://doi.org/10.1155/2022/5713092.
7. Hussain F, Hussain R, Hassan SA, Hossain E. Machine learning in IoT security: current solutions and future challenges. IEEE Commun Surv Tutor. 2020;22:1686–721.
8. Al-Garadi MA, Mohamed A, Al-Ali AK, Du X, Ali I, Guizani M. A survey of machine and deep learning methods for IoT security. IEEE Commun Surv Tutor. 2018;22:1646–85.
9. Yahya F, Zaki AFA, Moung EG, Sallehudin H, Bakar NAA, Utomo RG. An IoT-based coastal recreational suitability system using effective messaging protocol. Int J Adv Comput Sci Appl. 2021;12:8.

10. Routray SK, Gopal D, Javali A, Sahoo A. Narrowband IoT (NBIoT) assisted smart grids. In: Proceedings of the 2021 international conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India. 2021. p. 1454–8.
11. Sangra P, Rana B, Singh Y. Energy efficiency in IoT-based smart healthcare. In: Proceedings of third international conference on Computing, Communications, and Cyber-Security. Singapore: Springer: 2023. p. 503–15.
12. Mazhar T, Malik MA, Haq I, Rozeela I, Ullah I, Khan MA, Adhikari D, Ben Othman MT, Hamam H. The role of ML, AI and 5G technology in smart energy and smart building management. Electronics. 2022;11:3960.
13. Shukla PK, Sukla PK, Roy V. Network physical address based encryption technique using digital logic. Int J Sci Technol Res. 2020;9(4):3119–22.
14. Janarthanan T, Zargari S. Feature selection in UNSW-NB15 and KDDCUP'99 datasets. In: Proceedings of the 2017 IEEE 26th international symposium on Industrial Electronics (ISIE), Edinburgh, UK. 2017. p. 1881–6.
15. Karn RR, Kudva P, Elfadel IM. Learning without forgetting: a new framework for network cyber security threat detection. IEEE Access. 2021;9:137042–62.
16. Ahmed M, Byreddy S, Nutakki A, Sikos LF, Haskell-Dowland P. ECU-IoHT: a dataset for analyzing cyberattacks in internet of health things. Ad Hoc Netw. 2021;122: 102621.
17. Moustafa N. A new distributed architecture for evaluating AI-based security systems at the edge: network TON_IoT datasets. Sustain Cities Soc. 2021;72: 102994.
18. Kilincer IF, Ertam F, Sengur A. Machine learning methods for cyber security intrusion detection: datasets and comparative study. Comput Netw. 2021;188:107840.
19. Roy V. An improved image encryption consuming fusion transmutation and edge operator. J Cybersecur Inf Manag. 2021;8(1):42–52.
20. Neshenko N, Bou-Harb E, Crichigno J, Kaddoum G, Ghani N. Demystifying IoT security: an exhaustive survey on IoT vulnerabilities and a first empirical look on Internet-scale IoT exploitations. IEEE Commun Surv Tutor. 2019;21:2702–33.
21. Narayanan U, Paul V, Joseph S. Decentralized blockchain based authentication for secure data sharing in Cloud-IoT, DeBlock-Sec. J Ambient Intell Humaniz Comput. 2021;13:769–87.
22. Kalyani G, Chaudhari S. Cross layer security MAC aware routing protocol for IoT networks. Wirel Pers Commun. 2022;123:935–57.
23. Ali F, Mathew S. "An efficient multilevel security architecture for blockchain-based IoT networks using principles of cellular automata. Peer J Comput Science. 2022;8: e989.
24. Kaňuch P, Macko D. E-HIP: an energy-efficient open HIP-based security in IoT networks. Sensors. 2019;19:4921.