Baghdad Science Journal

Volume 23 | Issue 3 Article 26

3-25-2026

Implementing a Secure RPL Routing Protocol for IoT Networks using Lightweight Authentication Technique

Hussein A. Nassrullah
Department of Computer, College of Engineering, University of Baghdad, Baghdad, Iraq,
h.nassrullah@coeng.uobaghdad.edu.iq

Zainab T. Alisa
Department of Computer, College of Engineering, University of Baghdad, Baghdad, Iraq,
zainab.alisa@coeng.uobaghdad.edu.iq

How to Cite this Article

Nassrullah, Hussein A. and Alisa, Zainab T. (2026) "Implementing a Secure RPL Routing Protocol for IoT Networks using Lightweight Authentication Technique," *Baghdad Science Journal*: Vol. 23: Iss. 3, Article 26.
DOI: https://doi.org/10.21123/2411-7986.5251

BAGHDAD SCIENCE JOURNAL 2026;23(3):1040–1056

Scan the QR to view the full-text article on the journal website

RESEARCH ARTICLE

Implementing a Secure RPL Routing Protocol for IoT Networks using Lightweight Authentication Technique

Hussein A. Nassrullah *, **Zainab T. Alisa**

Department of Computer, College of Engineering, University of Baghdad, Baghdad, Iraq

ABSTRACT

The use of Internet of Things (IoT) networks is increasing daily both in terms of the number of active IoT devices and the number of sectors and applications using IoT technologies. This widespread deployment increases the importance of protecting these networks and secure their data from theft or manipulation. The IoT routing protocol (RPL) is the primary protocol used in resource constrained IoT networks. However, this protocol contains a number of security vulnerabilities that practically restrict its use in many sensitive applications. This research proposes a lightweight secure operating mode for the RPL protocol called Lightweight Security Mode RPL (LSM-RPL) that use Hash-Based Message Authentication Code (HMAC) for source authentication and data integrity. The proposed mode employs two types of pre-configured secret keys: The first is secret but shared by all network members to protect broadcast messages and prevent external intrusion. The second type of key is private key that shared only between the node and the root. It used to protect unicast messages and prevent internal intrusion. Simulation results demonstrated the proposed security layer strong ability to overcome the negative impact of several common RPL attacks. These results were compared with several state-of-the-art studies in this field. For example, under the Decrease Rank Attack scenario with two malicious nodes, LSM-RPL demonstrated significant improvements over RPL, achieving up to 10.1% higher Packet Delivery Rate (PDR), and reducing energy consumption and total packets received by 45.9% and 26.4%, respectively.

Keywords: Attack mitigation, Decrease rank attack, HMAC, Information security, IoT, Keys distribution, Protocol evaluation, RPL

Introduction

Over the past few years, the use of the Internet of Things (IoT) has spread and begun to use in various sectors and applications.[1] The International Data Corporation (IDC) predicts that the number of active IoT devices will reach 55 billion devices. These devices will generate a massive amount of data, estimated at 80 billion zeta-bytes.[2] Most IoT networks transmit and store their data via the cloud. With the growing artificial intelligence (AI) revolution, IoT data can be analyzed and processed using AI and enabling these networks to make intelligent decisions independently without human intervention.[3] All of this has led to the integration of IoT networks into large and diverse sectors such as agriculture, industry, healthcare, and military applications. Due to the increased use of IoT networks and their integration into sensitive applications, securing these networks has become crucial. For example, in healthcare[4] applications breaches of these networks can violate patient privacy and may endanger their lives. Similarly in industrial applications or autonomous driving technologies,[5] information security breaches could disrupt production or even endanger lives. All of this has made information security a fundamental aspect of research related to IoT networks.[6] The Routing Protocol for Low power and lossy network (RPL)

Received 6 May 2024; revised 16 November 2024; accepted 18 November 2024.
Available online 25 March 2026

* Corresponding author.
E-mail addresses: h.nassrullah@coeng.uobaghdad.edu.iq (H. A. Nassrullah), zainab.alisa@coeng.uobaghdad.edu.iq (Z. T. Alisa).

https://doi.org/10.21123/2411-7986.5251

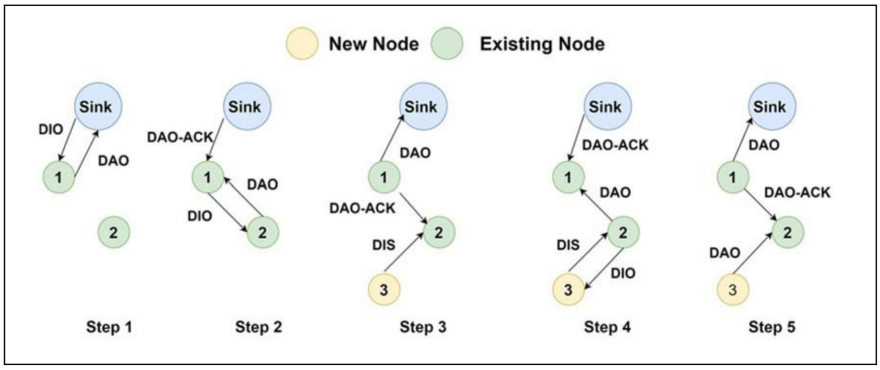

Fig. 1. Topology construction in RPL.

has been the officially adopted by Internet Engineering Task Force (IETF) standardization group[7,8] as a standard routing protocol for Low power and Lossy Networks (LLNs) since 2012. This protocol possesses several characteristics that make it suitable for the Internet of Things (IoT), such as its lightweight nature, as it is designed for resource-constrained networks, and its scalability and ability to operate across diverse devices and environments.[9] However, the RPL protocol suffers from numerous security vulnerabilities at multiple levels, which threatens its practical use in many IoT applications. Any proposed solutions to secure IoT networks must take into account the constrained nature of IoT devices.[10] This research focuses on protecting RPL protocol by adding a lightweight layer of security while maintaining its operational capability on resource constrained devices. This study focus on enhancing RPL's security level while maintaining its efficiency to operate on constrained IoT devices. The paper is organized as follows: First, the background and security challenges in IoT networks are reviewed. Following that, related works that address security vulnerabilities in the RPL protocol will be discussed. Then the presentation of the proposed LSM-RPL scheme. The simulation setup and results are then provided, demonstrating the effectiveness of the approach. Finally, the paper concludes with a discussion of future work and potential directions for enhancing IoT network security.

Review background

RPL is a standard routing protocol designed for resource constrained networks, unreliable links, and frequent connectivity changes (the typical characteristics of many IoT networks). The RPL protocol has a number of key features, including the ability to adapt to changes in the network, scalability, and low power consumption. RPL is a Distance Vector routing protocol tailored for Low-Power and Lossy Networks (LLNs). Within this protocol, the organization of network path information takes the form of a collection of Directed Acyclic Graphs (DAGs); RPL employs a Directed Acyclic Graph (DAG) as its core routing structure, which is further categorized into Destination-Oriented Directed Acyclic Graphs (DODAGs). A typical DODAG comprises sensor nodes and a sink node (root) responsible for aggregating data from these sensor nodes. The RPL protocol uses four ipv6 ICMP control messages to construct the upward (from nodes to root) and downward (from root to nodes) routing and to construct the network topology,[9] as in Fig. 1.[11] The four messages are:

- DODAG Information Solicitation (DIS): This is sent by a node when it wants to join the RPL network and request information about DODAGs from its neighbors.
- DODAG Information Object (DIO): A DODAG root uses this object to broadcast information about the DODAG to potential network members. It contains details such as the DODAG ID, version, objective function, configuration parameters, and other routing information.
- Destination Advertisement Object (DAO): A DAO is sent by a node to the root to indicate its presence in the network and reach certain destinations. It is used to build downward routes.
- Destination Advertisement Acknowledgment (DAO-ACK): is sent by the root to acknowledge the DAO message.

Security challenges in IoT

IoT networks usually consist of a group of constrained devices with limited hardware and software capabilities.[12] Conventional cryptographic systems

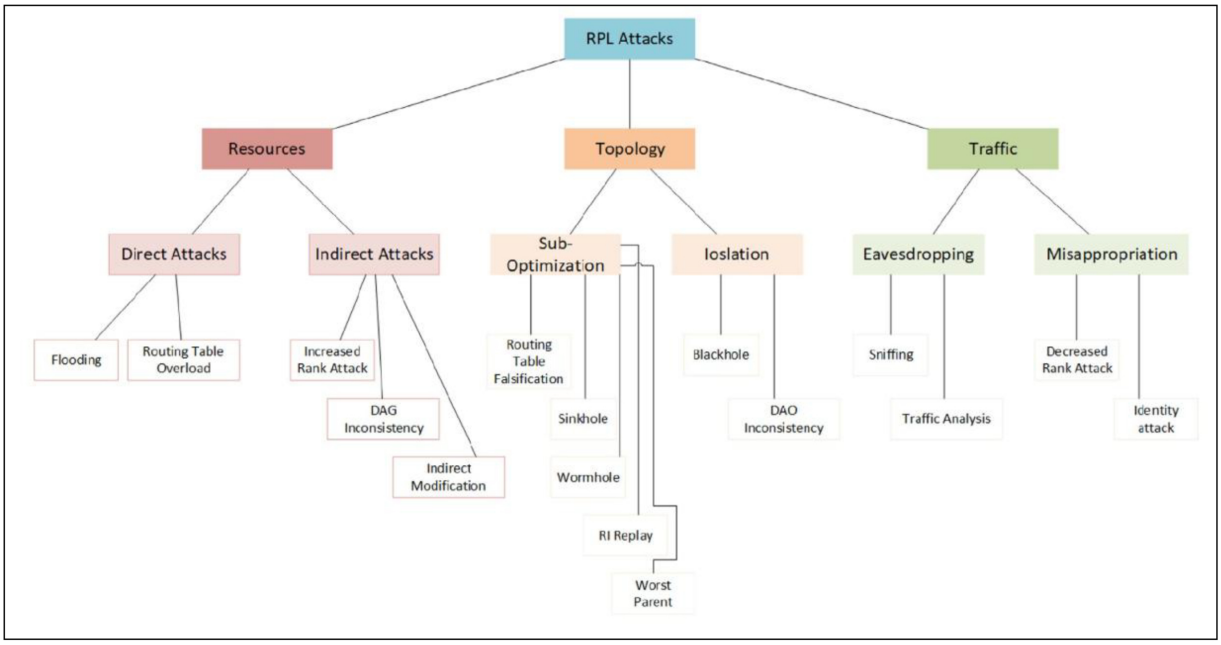

Fig. 2. Taxonomy of RPL attack types.

are often complex and demand high computational power, making them unsuitable for IoT devices.[13] The rapid expansion of IoT networks has also increased the attack surface, exposing these networks to a wide range of security threats, including DDoS attacks, unauthorized access, data breaches, and routing attacks such as version number and rank attacks. These attacks can severely disrupt network operations and compromise data integrity.[14]

IETF developed the RPL protocol to work with constrained devices, suggesting several repair mechanisms for RPL networks, such as the detection of link failure, loop occurrence, and other network inconsistencies. Although the IETF provides mitigation mechanisms for such issues, for example, by launching global repairs through increasing the version number in the DIO control message sent by the root, RPL remains subject to a large number of security issues. Some of these issues exploit RPL repair procedures, such as the version number attack, as will be discussed later.[9] Given the limited resources of IoT devices, there is a pressing need for a lightweight security solution that can effectively mitigate these threats without imposing significant computational or energy overheads.[14]

RPL protocol faces a variety of security threats that can compromise the stability and efficiency of IoT networks. These threats can be broadly categorized into three dimensions: Resources, Topology, and Traffic. Resource-based attacks, such as flooding, aim to deplete the physical or computational resources of RPL nodes. Topology-based attacks manipulate the network's structure, causing issues like routing table overload. Traffic-based attacks, including eavesdropping, exploit vulnerabilities in the data transmitted over the network.

Fig. 2[15] presents a detailed taxonomy of these RPL attack types, offering a comprehensive framework for understanding and mitigating these security threats.[8,15]

In this work, two of the most common security attacks encountered in the RPL protocol will be addressed:

Version number attack (VNA)

RPL protocol, the root is responsible for creating DODAG by periodically sending a DIO message. Each DODAG is uniquely identified by DODAGID and version number (V.N.). Incrementing version number initiates global repair and means an advertisement from the root to other network nodes about the new DODAG version. Only the root node was authorized to change the version number, but intruders may exploit this feature to tamper with the version number filed in the DIO message, as shown in Fig. 3, and then forward it to its neighbors.[16] Version Number Attack has been classified as an internal RPL-based attack. It affects IoT network performance by adding waste global repair overhead, which causes increased energy consumption and decreases network lifetime,

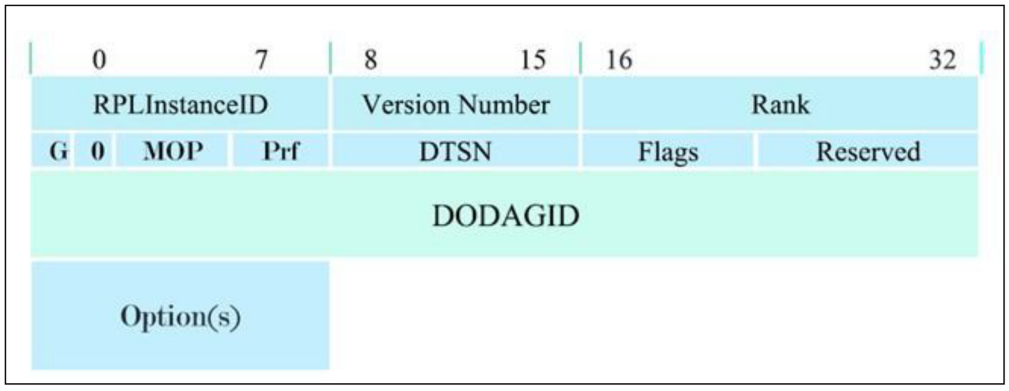

Fig. 3. DIO base object structure.

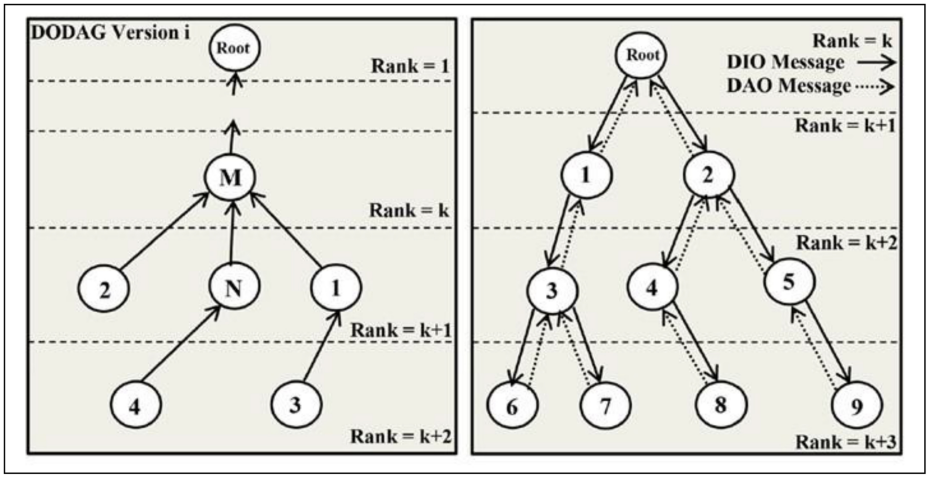

Fig. 4. Rank distribution in RPL network.

furthermore decreasing the packet delivery rate and raising the end-to-end delay.[8]

Increased and decreased rank attacks

The rank value is a number that represents the nearest node to the root. It is initially sent by the root in a DIO message as shown in Fig. 3.[8]

Every node that receives a DIO message selects its preferred parent according to the rank value in the DOI message received from that parent. When the node wants to broadcast a DIO message to its neighbors, it should increase its rank value by adding the distance between the node and its parent according to an objective function as shown in Fig. 4.[17]

A malicious node broadcasts an invalid rank value to its neighbors in a rank attack. In a decreased rank attack, the malicious node broadcasts less than actual rank values, which causes other nodes to select the malicious node as their preferred parent.[8,18] Decrease rank attacks (DRA) represent a series of dangers on network topology, especially if they are mixed with other attacks like blackholes and selective forward attacks. In an increased rank attack, the malicious node broadcasts more than its actual rank value, so its neighbor may choose a nonoptimal parent as the preferred parent. This attack can disrupt the routing topology and decrease network performance.

Related works

Alani et al.[19] introduce a two-layer intrusion detection system for IoT network, utilizing machine learning techniques to address security vulnerabilities. It achieves high accuracy in detecting intrusions with minimal time overhead however, the paper does not address specific RPL problems or attacks and does not utilize datasets related to RPL, which limits its applicability to RPL-based networks. Additionally, the complexity of machine learning models may pose computational challenges for resource-constrained IoT devices, further limiting scalability.

IRaoof et al.[20] conducted a comprehensive study on various types of RPL attacks and their effect on network performance, analyzing the effectiveness of intrusion detection systems in detecting and mitigating these attacks. Agiollo in[21] proposed an Intrusion Detection System (IDS) named DETONAR that uses a packet sniffing approach to detect and mitigate attacks in RPL-based IoT networks. DETONAR combines signature and anomaly-based rules to identify malicious behavior in RPL traffic However; to apply this research, the author assumes to use external sniffing devices that can capture each traffic in the network but such an assumption is difficult to apply practically. Other works focus on Elliptic-Curve cryptography (ECC). Ismail et al.[22] have focused on enhancing the efficiency of mathematical operations employed in ECC. On the other hand, Adarbah et al.[23] suggested enhancements to RPL security to mitigate selected forward attacks utilizing Elliptic-Curve Diffie-Hellman (ECDH) for key exchange and authentication. They suggested a hybrid encryption, which uses public and private keys to generate and exchange session keys and then uses this session key in symmetric encryption. M. Nikravan et al.[17] proposed a lightweight Identity Based Offline–Online Signature (IBOOS) scheme to counter version and rank attacks. The IBOOS algorithm assumes the root acts as a private key generator, sending each node's private key through a secure channel; however, this assumption is impractical for most IoT networks. Prathapchandran et al.[24] introduced a trust-based lightweight security mechanism called RFTrust to address the sinkhole attack in RPL-based IoT environments. They used Random Forest (R.F.) and Subjective Logic to improve network security. Conti et al.[25] introduced SPLIT, a secure and scalable RPL routing protocol for IoT networks. SPLIT employs a lightweight self-attestation technique to ensure the software integrity of IoT nodes. The SPLIT algorithm assumes that each IoT node consists of a trusted component (Verifier) and an untrusted component (Prover), and the Verifier calculates the hash of the untrusted component. G. Glissa et al. in[26] proposed a secure mechanism for RPL protocol called Secure-RPL (SRPL) to address the vulnerabilities and attacks related to control message manipulation of RPL protocol. SRPL introduces the concept of rank threshold and hash chain authentication to prevent misbehaving nodes from maliciously changing control message values, such as the rank of a node, which can disrupt the network. Essop et al.[27] focused on the need for accurate and efficient anomaly-based intrusion detection systems (AIDSs) for IoT networks, generating labeled RPL attack datasets to provide researchers with benchmark datasets that can be used to develop accurate and efficient AIDSs for RPL networks. P. Newton et al.[28] proposed a technique for detecting and overcoming rank attacks in the RPL routing protocol. The proposed technique uses a lightweight Hashed Message Authentication Code (HMAC-LOCHA) to verify the integrity and authenticity of control messages; however, this paper needs a detailed discussion of implementation. Specifically, it does not cover how to handle secret key management and how HMAC-LOCHA prevents internal attacks where a malicious node with the same shared secret key might tamper with control message information and claim it originates from the root. In,[18] S. Karmakar et al. suggested a low overhead strategy for detecting rank attacks in RPL networks that employs the HMAC-LOCHA algorithm for verifying the integrity and legitimacy of control messages exchanged between nodes and the sink. The strategy involves modifying the header of RPL DAO control messages to include parent rank and hash code information to improve detection accuracy. While simulations show improved detection accuracy, a critical limitation exists. The Authors assume a shared secret key among all nodes; consequently, a malicious parent node could tamper DAO pass-through packets from its child, generate a new HMAC code using the same secret key, and send it to the root so that the attacker can compromise the network security in such practical scenarios. Other studies including Ambarkar et al.,[2] which presented a mutual authentication mechanism utilizing flag bits in RPL control packet headers to block unauthenticated nodes. However, their approach relies on the RPL instance ID as a security key, which is a significant limitation. Since all nodes in the network share this ID, a malicious node can generate a fake authentication package and claim it originates from the root. M. Momand et al. in[29] proposed a Machine Learning technique to detect multiple types of RPL attacks called (MLRP) by creating a dataset with normal and attack behaviors of IoT nodes in the RPL network. MLRP uses the dataset to learn and test machine learning detection systems to detect mainly three types of attacks (version, rank, and DoS attack). Algahtani et al.[30] Provided a reference implementation of five types of RPL attacks. They published the full Source Code on GitHub, offering an efficient way to implement and test multiple RPL attacks simultaneously. Jamil's work[31] on a triple DES-based blockchain algorithm enhances IoT network security by ensuring data integrity and scalability, offering a more efficient alternative to RSA, particularly for IoT devices with limited computational capacity. Kareem et al.[32] use machine learning techniques, specifically XGBoost, Naive Bayes (NB), and Linear Discriminant

Analysis (LDA), to detect RPL attacks. The study focuses on identifying four specific attacks: Flooding Attack, Blackhole Attack, Decreased Rank Attack, and DODAG Version Number Attack, using a multi-class classification approach. The main limitation in such study is the use of a dataset based on record of all packets transferred across the entire network. Such dataset can easily build in simulation program but impractical in real world IoT networks. Other authors[33] have investigated the DAO induction attack. In DAO induction attack, the compromised node triggers other nodes to respond with DAO control messages to add overhead on the network. They proposed a solution that identifies and blocks the malicious node.

In summary, the previous studies have significant contribution in RPL security but many on them do not address the specific challenges of resource constrained IoT environments. Some solutions rely on complex security algorithms that required high computations or use machine learning and require special environments for dataset generation making them difficult to implement in practical scenarios. Others assume there are a secure channel between the root and all nodes, which is often impractical in IoT networks. This work aims to propose a novel security mechanism to address RPL specific vulnerabilities while maintaining efficiency to work with IoT constraint devices.

The proposed scheme

This work proposes a secure mode of operation in RPL protocol. The proposed technique is called Lightweight Security Mode RPL (LSM-RPL). Its implementation published as open source as a forked from the Contiki-NG repository and can be found in the GitHub repository *. LSM-RPL protect RPL control messages by employing Hashed Message Authentication Code (HMAC) technique to ensure data integrity and source authentication. It prevents any node from joining the network unless it is predefined in the root and have the keys. This will address external attackers by prevent unauthorized node from joining the network. The proposed scheme also protects the network against the internal attacks if one of the distributed nodes is compromised by an attacker and behaves as a malicious node. The proposed method is supposed to be able to identify and isolate the malicious node from the network. Each sensor node that wants to join the RPL network should have two types of secret keys: a private key (K_{pr}) and a shared key (K_{sh}). K_{pr} is the secret key between the node and the root, while K_{sh} is a secret key shared between all the nodes in the network. The root is assumed to be secured and should store a list of private keys, one for each node on the network. It should also store the Shared secret key of the network. These keys are used to authenticate RPL control messages using HMAC, as will be explained later.

In this work, the term “signed” didn’t mean a digital signature using asymmetric key cryptography but HMAC calculation. Therefore, when saying the message M is signed with secret key K, it means that the message M is appended with the HMAC digest of M calculated using K. To protect the network against RPL attacks that manipulate the RPL control message (like version and rank attacks), it is important to ensure the integrity and authenticity of these control messages. Using HMAC with K_{sh} is suggested to protect the message sent with broadcast addresses like (DIO and DIS) and using HMAC with K_{pr} when sending a control message between the specific node and the root with unicast address (like DAO and DAO-ack in RPL non-storing mode). To avoid heavy overhead on the network, sending the hash code (digest) of the HMAC function in the same ICMP control message header is suggested. LSM-RPL proposes to sign a DIO control message using a shared secret key (K_{sh}) to prevent any external attacker from fabricating a fake DIO control message. Fig. 5 shows the proposed structure of the secure DIO control message. The essential modifications were made to the structure of the DIO message by Inserting the HMAC digest, with a length of 4 bytes, into the already reserved fields that are not used in the options part of the ICMP IPv6 header. Thus, there is no increase in the size of the DIO message nor any additional overhead transmission data. In this work, the proposal is to use MACs Based on Hash Functions (HMAC) instead of MACs Based on Block Ciphers (CMAC) because HMAC is matched faster than CMAC and is suitable with IoT constraint devices.[34] The proposed HMAC uses a 128-bit key length and 32-bit digest. This critical size is sufficient for IoT applications using the HMAC method, as the attackers cannot conduct offline attacks by attempting to calculate the digest for multiple messages because they lack the secret key of the HMAC. Instead, attackers would need to monitor the passing packets and their digests, which are of exceptionally colossal size (more than 264 Blocks). Accomplishing this task would require recording data at 1 Gbps for approximately 250,000 years to break a 128-bit key.[34]

* source code: "Lightweight Security Mode RPL (LSM-RPL)" available at: https://github.com/HUNSR/contiki-ng-LSM-RPL

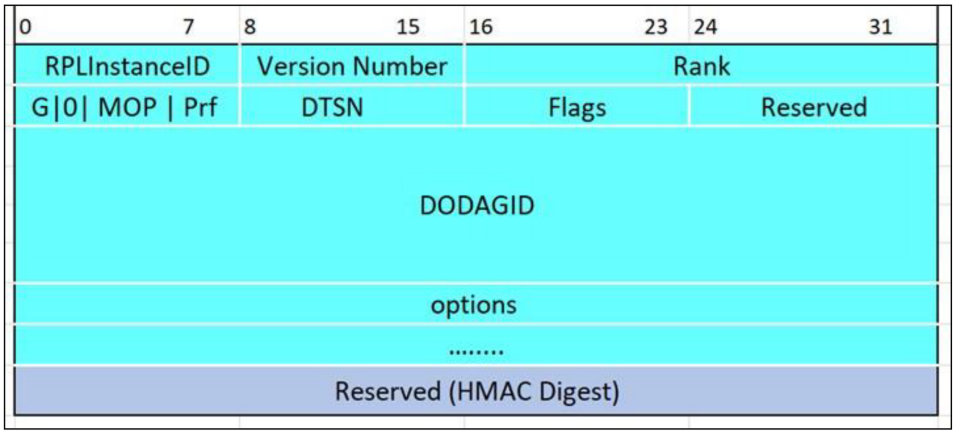

Fig. 5. Proposed DIO base object structure.

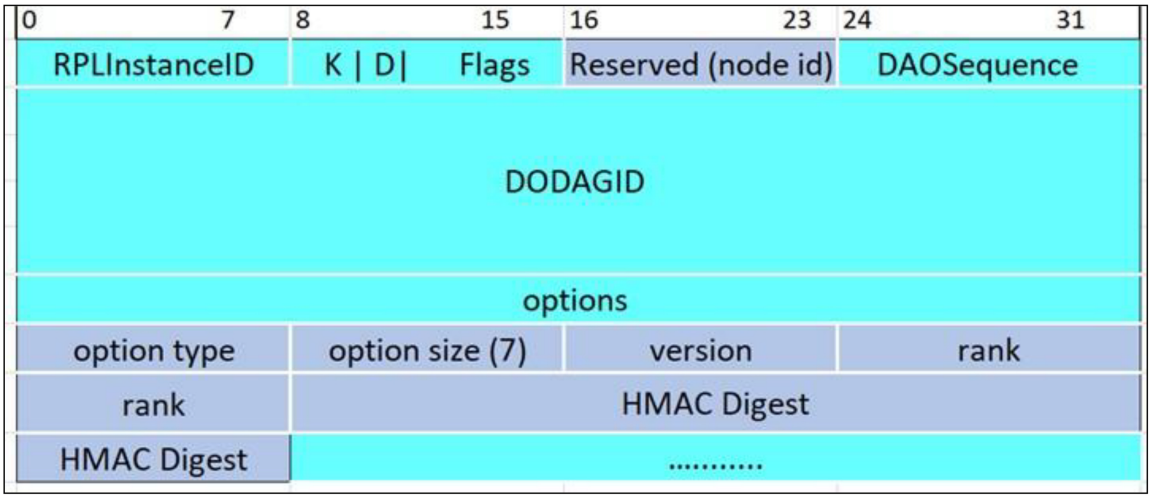

Fig. 6. Proposed DAO structure.

However, protecting all deployed IoT motes may be practically impossible, and the attacker may be able to compromise a few nodes in the network either through the physical attack of the IoT device or remotely compromise the node by exploiting the constraint hardware and software capability.[33] Therefore, a list of private keys is used to prevent the attacker from controlling the entire network even if he succeeded in compromising a number of nodes in the network and getting the secret keys (K_{sh} and K_{pr}) of these nodes. The best option in this case is to remove the compromised nodes from the network. So, the first step is to identify the compromised nodes and then inform the rest of the nodes in the network not to deal with these compromised nodes.

In the proposed solution, if the attacker gains control of a legitimate node, they can tamper with the DIO message by altering values such as the version or rank. This is possible because the DIO signed with a shared secret key, which the attacker already possesses. Consequently, the attacker can send his neighbors an undetectable fake DIO message. But in RPL protocol, if one of the neighbor nodes decides to choose the malicious node as its preferred parent based on the received DIO message, it should send a DAO message to the root, pass through candidate parents (malicious node), and receive DAO-ACK from the root to accomplish the joining of RPL network.

To protect the RPL network against version and rank attacks, LSM-RPL proposes an essential modification to the DAO message as illustrated in Fig. 6. This modification involves appending the current version and rank values along with the digest of entire DAO message signed with K_{pr}. In this case, if the malicious node (preferred parent node) discards the DAO message, this means denial of service and prompting the child node to choose another preferred parent according to RPL protocol. Suppose the malicious node manipulates the DAO message before forwarding it toward the root. In that case, the message will

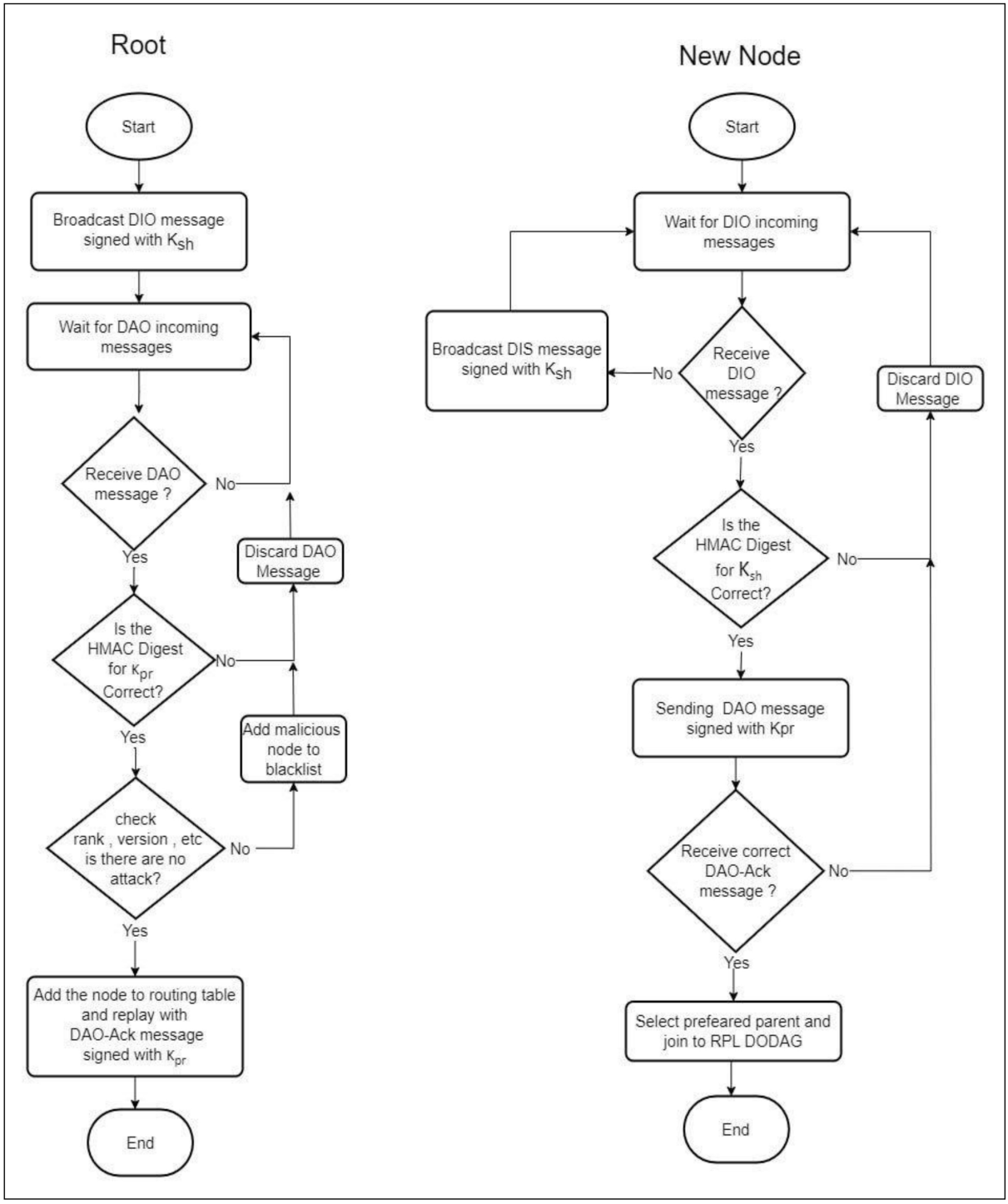

Fig. 7. Joining algorithm for LSM-RPL networks.

lose its authentication, as the malicious node lacks the private key to communicate between the child node and the root. Alternatively, if the malicious node forwards the DAO message unchanged, in this case, the root can detect any manipulation in the original DIO message information, thus detecting attacks like version and rank attacks.

In response, if the DAO message remains intact and untampered, the root replay with the DAO-Ack message that will be sent to the unicast address of the DAO sender signed by K_{pr}, as shown in Fig. 7. In RPL, when the node receives a DIO message from its neighbors, including its preferred parent or any other neighbors, with a different version number than the

```
C rpl-dag.c 2 ×
os > net > routing > rpl-lite > C rpl-dag.c > process_dio_from_current_dag(uip_ipaddr_t *, rpl_dio_t *)
431  process_dio_from_current_dag(uip_ipaddr_t *from, rpl_dio_t *dio)
467    if (rpl_lollipop_greater_than(dio->version, curr_instance.dag.version))
468    {
469      if (curr_instance.dag.rank == ROOT_RANK) ···
478      else
479      {
480        // hunsr code start
481  #if LSM == 1
482        nbr = rpl_neighbor_get_from_ipaddr(from);
483        if (nbr == NULL || nbr != curr_instance.dag.preferred_parent)
484        {
485          LOG_WARN("hunsr server detect version attack from ");
486          LOG_INFO_6ADDR(from);
487          LOG_WARN(" (current version: %u, received DIO version: %u), prevent apply global repair\n",
488          curr_instance.dag.version, dio->version);
489          return;
490        }
491  #endif /* LSM == 1 */
492        // hunsr code end
493        LOG_WARN("new DIO version (current: %u, received: %u), apply global repair\n",
494        curr_instance.dag.version, dio->version);
495        global_repair_non_root(dio);
496      }
497    }
```

Fig. 8. Code snippet for avoiding Inconsistent DIO version by unauthenticated neighbors.

current one being used by the node, it is required to initiate global repair to resolve the inconsistency in DIO versions. However, if a malicious node joins the RPL network and conducts an internal version number attack by broadcasting DIO messages with varying version numbers, it can significantly degrade network performance. This will negatively affect the network performance. To mitigate such scenarios, LSM-RPL suggests restricting updating the version number to only be done by the preferred parent. Fig. 8 shows for the code snippet implementation in Visual Studio Code. This limitation ensures that the preferred parent, which has undergone prior authentication, is the sole entity authorized to broadcast the version number update.

In the implementation of this work, LSM macro is utilized as a configuration option to enable or disable Lightweight Security Mode (LSM) of RPL protocol. Setting the LSM macro value to 1 activates LSM functionality across all parts of the codebase and including security-related features such as version attack detection and prevention. This approach of implementation minimize the program image size and centralize the activation of LSM in a single configuration parameter. For instance, in the snippet provided, the #if LSM == 1 is a preprocessor directive to ensure that the version attack detection mechanism is only active when LSM is enabled. By employing the LSM macro consistently across the codebase, flexibility in customizing the protocol's behavior is maintained, while promoting code modularity and ease of maintenance.

The proposed method for distributing secret keys and ensuring authentication for new nodes, as well as maintaining the integrity of control messages can serve as a foundational approach to safeguarding the RPL protocol against a range of attacks that target RPL control messages, these include DAO induction attack, Neighbor attack, Sybil attack, Parent Switching Attack, and other such attacks, by implementing strategies similar to those outlined in this work.

Simulation and results

In this section, the performance of the proposed routing protocol, LSM-RPL, is evaluated and compared with the standard RPL protocol and state-of-the-art technique used in this field. The evaluation employed the Contiki-NG[35] and Cooja simulator. Contiki-NG is a new generation of Contiki O.S., which is an open-source operating system designed for low-power, memory-constrained devices used in the Internet of Things (IoT), while the Cooja simulator is a popular RPL network simulator with a graphical user interface built explicitly for Contiki and Contiki-NG.[35] This allows the emulation of sensor networks and test communication protocols in a virtual environment before deploying them on real IoT device hardware. Although Contiki-NG is primarily designed for Linux environments, in this research, Windows Subsystem for Linux (WSL) is used to run it on Windows machines.

WSL acts as a compatibility layer, enabling us to run Linux binaries, including Contiki-NG, directly on Windows without requiring any modifications. The Contiki-NG system provides implementations for IoT protocols, including the RPL protocol. These

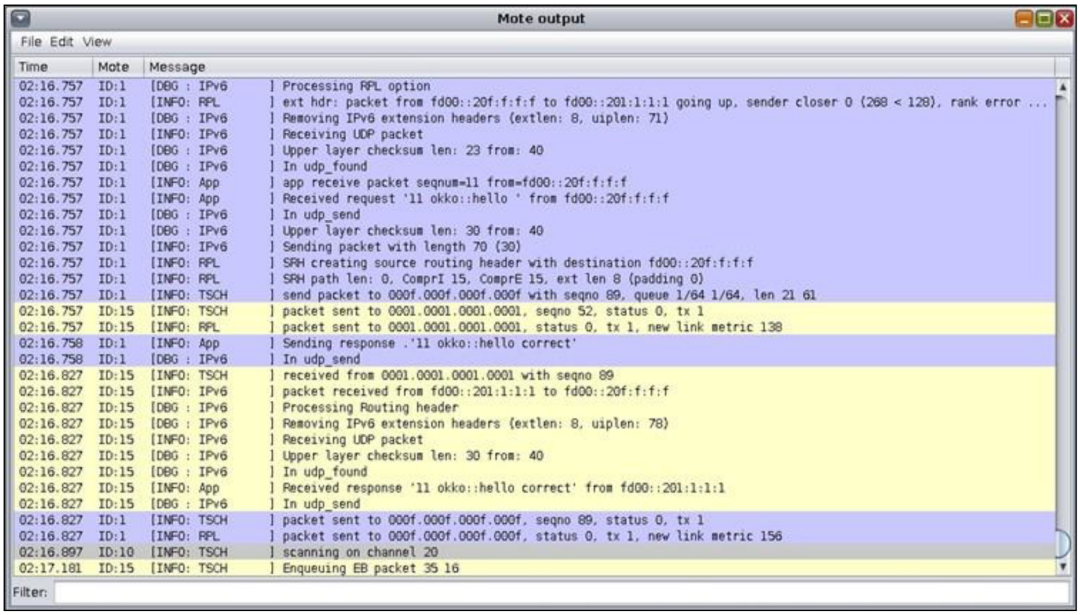

Fig. 9. Cooja simulator log file.

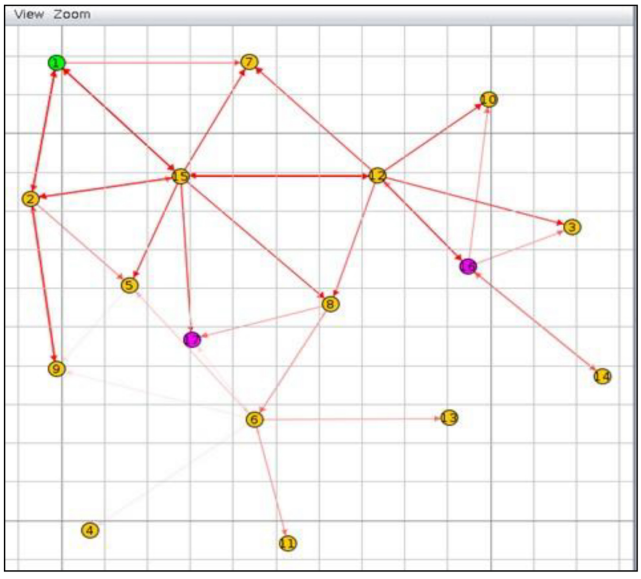

Fig. 10. Network layout, node 1 as root, Nodes 2–15 legitimate, Nodes 16 & 17 malicious.

protocols are typically written in the C programming language. Therefore, when modifying the RPL C code, the Visual Studio Code editor is utilized. The Wireshark program is used to analyze data packets sent by IoT nodes, especially headers of ICMP control messages. Wireshark is a popular open-source network protocol analyzer for troubleshooting, analysis, and protocol development. The output results obtained from the Cooja simulator are a log file containing thousands of lines, as illustrated in Fig. 9. Python code is employed to read and analyze such results effectively, which parses the log file line by line to extract network performance parameters.

Simulation environment

To evaluate the proposed routing protocol, LSM-RPL, a small-scale network simulation was conducted using Contiki-NG OS and the Cooja simulator. In Fig. 10 illustrates the network layout, which consists of 14 nodes in addition to the root node and one or two compromised (malicious) nodes, arranged within a 150 × 150 meter area.

Unit Disk Graph Medium (UDGM) was utilizes as the radio model in simulation. It represents a simple model based on distance between the two nodes. If the other node within communication range, the

Table 1. Simulation parameters.

Parameter	Value
Simulation time	3600 second
Network size in meter	150 * 150
Radio Model	Unit Disk Graph Medium (UDGM)
Objective function	Minimum Rank with Hysteresis
Medium Access Control (MAC)	Time-Slotted Channel Hopping (TSCH)
Number of normal nodes	15
Number of sink nodes (root)	1
Number of malicious nodes	1, 2
CPU clock frequency in megahertz.	48
Current consumption during radio transmission at 5 dBm.	9.100 mA
Current consumption during radio reception.	5.900 mA
Current consumption of CPU when active during radio operations.	0.061 * 48 = 2.928 mA
Current consumption of CPU during sleep mode.	1.335 mA
Current consumption of CPU during deep sleep mode.	0.010 mA
Operating voltage	3.3 volte

node can directly communicate with it. The objective function used in this simulation is Minimum Rank with Hysteresis. Selection of the best routes in this objective function based on node ranks and to prevent frequent route changes it applies hysteresis concept.

The performance of the proposed LSM-RPL protocol was evaluated compared with the standard RPL protocol under three attack scenarios: No Attack (NA), Version Number Attack (VNA), and Decrease Rank Attack (DRA). All details about the simulation parameters is provided in Table 1.

Parameters used for energy estimation are based on the datasheet of the CC2650 system-on-chip (SoC) from Texas Instruments, These energy parameters are used by energy model embedded with Contiki-NG to estimate the nodes energy consumption.

Evaluation metrics

This section lists the basic evaluation metrics used in this work to evaluate the different scenarios. Each metric reflects an aspect of protocol behavior and network performance.

Packet Delivery Ratio (PDR): PDR represents the ratio of successfully delivered packets to the total packets transmitted by the nodes. It reflect the ability of the routing protocol in arriving the data packets to their destinations. However, PDR alone may not provide a complete picture about network performance. This is because sometimes the nodes may fail to join the network and didn't send any packets on the network. To address this issue, it should monitor the total number of packets sent by the nodes or the total number of packets received by the root in addition to PDR.

Number of packets received by the root: It represents the total number of packets sent by the nodes and successfully arrived to the root.[36] Such parameters represent accumulative number that increases as the network runtime increase.

Number of Parent Node Switches: This metric represent the frequency of parent node changes within the network. The parent switch accrue normally when a node find neighbor node with lower rank than its current parent. It reflects the stability of the routing protocol and the ability of protocol to maintain parent-child relationships. It is clear that frequent changes in parent nodes add overhead to the network and decrease its performance.

Number of Packet Overheads: The Number of Packet Overheads metric evaluates the additional control messages or overhead generated by the routing protocol to establish and maintain network connectivity. As mentioned previously, in the RPL protocol, there are four control messages that represent overhead packets (DIO, DIS, DAO and DAO-ACK).

Energy Consumption: Many IoT devices are often operate in resource-constrained environments with a limited battery, and therefore reducing energy consumption is a critical parameter in IoT networks, especially communication, processing, and Sensing Energy.[37]

Latency: Latency in IoT networks refers to the time it takes for a data packet to travel from its source (the sending node) to its destination (typically a sink node or a central server). Latency is significant because it directly impacts the responsiveness and reliability of the network.[38]

Jitter: Jitter in IoT networks refers to the variation in packet arrival times. It reflect the quality of real-time communications, such as voice or video streaming. It is important in real time IoT application such as sensor networks and automated systems.[38]

Table 2. Comparison of RPL and LSM-RPL performance under various attack scenarios.

Attack	PDR	Energy Consumption (mJ)	Total packet sent	Total packet received	Number of Overhead packets	Parent Switches	Latency (ms)	Jitter (ms)
No Attack (RPL)	99.92	105116.187	4921	4917	1782	22	462.27	484.38
No Attack (LSM-RPL)	99.98	105305.673	4928	4927	2004	21	543.55	582.30
VNA (1 M) (RPL)	98.43	107665.47	4578	4506	5604	1419	490.23	494.60
VNA (1 M) (LSM-RPL)	99.92	105140.541	4923	4919	2734	18	537.19	551.64
DRA (1 M) (RPL)	97.16	111966.129	4645	4513	6962	351	1156.87	1118.73
DRA (1 M) (LSM-RPL)	99.59	105626.796	4914	4894	2170	114	608.93	608.08
VNA (2 M) (RPL)	89.86	179400.87	3294	2960	8300	589	367.25	338.92
VNA (2 M) (LSM-RPL)	99.96	112003.221	4885	4883	3726	18	455.36	454.61
DRA (2 M) (RPL)	94.67	118359.45	4054	3838	13324	962	2363.51	2092.98
DRA (2 M) (LSM-RPL)	99.3	112927.551	4885	4851	2480	93	718.62	745.16

Results and discussion

The performance of the standard RPL protocol and the proposed Lightweight Secure mode RPL (LSM-RPL) protocol was evaluated across different attack scenarios with one or two malicious nodes. Each malicious nodes could execute two types of attacks VNA and DRA. Table 2 summarized the results of all scenarios. All the values in the rows of table represent the aggregate performance of all nodes in the network excluding the root node and the malicious nodes. “(1 M)” and “(2 M)” denote one and two malicious nodes respectively.

Under normal operating conditions without attacks (first two rows) both RPL and LSM-RPL achieved high PDR. LSM-RPL’s energy consumption was slightly higher than RPL’s, with an increase of approximately 0.28% because of the computation overhead. This small overhead is acceptable compared with the security benefits provided by LSM-RPL. Other evaluation metrics under normal condition are very close.

To evaluate the proposed LSM-RPL protocol, Two types of adversaries was considered: Internal and external attackers. An internal attacker refers to a compromised node that is already part of the network and have valid cryptographic credentials. In contrast, an external attacker is an unauthorized node attempting to interfere with the network despite not being a member of it. The proposed LSM-RPL protocol’s results demonstrated no variance between the external attack and no attack scenarios. This because LSM-RPL’s protocol successfully prevent nodes lacking the shared secret key (K_{sh}) from joining the network.

In internal attack scenarios, the attacker compromises the victim node and get access to both shared secret key (K_{sh}) and private secret key (K_{pr}). LSM-RPL addresses internal attacks by utilizing HMAC for source authentication and message integrity to ensure that any manipulation in control messages that include routing information can be detected and mitigated even if the attacker have the secret keys. LSM-RPL proposed protocol assumes that the secret key (K_{pr}) is remain secret and known only by the root so if the sender signs the control message by this key, any intermediate node cannot manipulate the message even if it is compromised and work as malicious node. This prevent malicious nodes from falsely report their rank.

Under the VNA scenario (row3 and 4) simulation results shows that LSM-RPL achieve significant improvment over standard RPL. In one malicious node, LSM-RPL shows a slight improvement in PDR by approximately 1.49% compared to RPL. While with two malicious nodes improvement in PDR rise to 10% over RPL. LSM-RPL displayed superior energy efficiency, reducing energy consumption by approximately 2.24% and 37.65% with one and two malicious nodes, respectively. Moreover, LSM-RPL outperforms RPL in terms of total packets received by approximately 9.1% and 6% in one and two malicious nodes, respectively. Under the Decrease Rank Attack (DRA) scenario, LSM-RPL also demonstrated significant enhancements over RPL with up to 10.1%, 45.9%, and 26.4% in terms of Packet Delivery Rate (PDR), minimizing energy consumption and total packets received, respectively.

Last two columns in Table 2 show that the Decrease Rank Attack (DRA) significantly increases both latency and jitter in RPL, as it disrupts optimal path selection, forcing nodes to choose suboptimal routes to the root. In contrast, the Version Number Attack (VNA) has a lesser impact on these metrics since it doesn’t alter path selection, though it may affect other aspects like energy consumption and packet overhead. LSM-RPL exhibits slightly higher latency and jitter compared to RPL under the “No Attack” scenario due to the additional security mechanisms, which introduce extra processing overhead. However, LSM-RPL significantly outperforms standard RPL, particularly under DRA conditions in scenarios

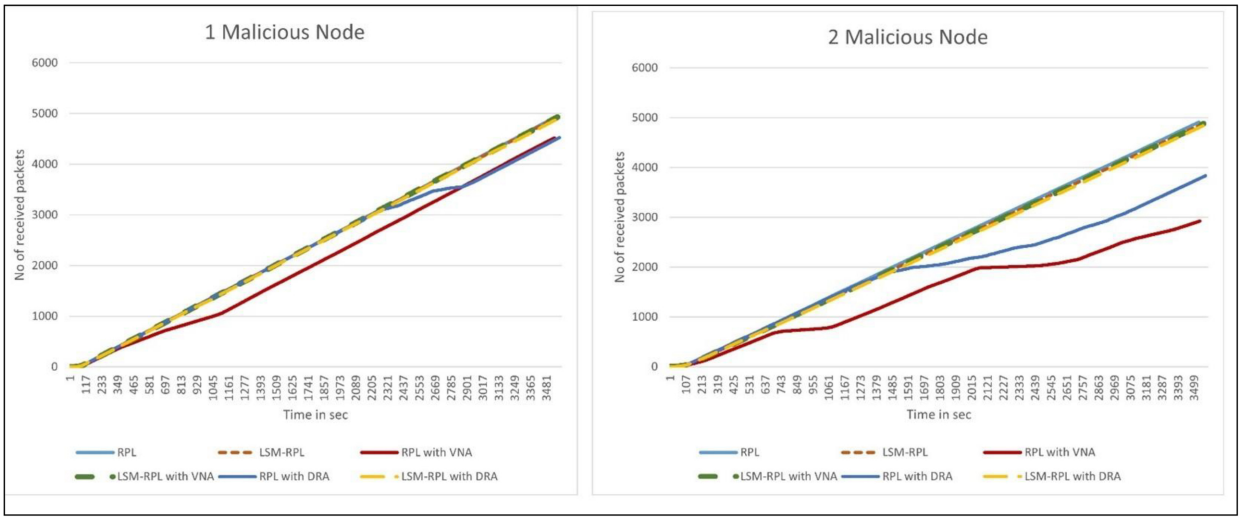

Fig. 11. Packet received over time.

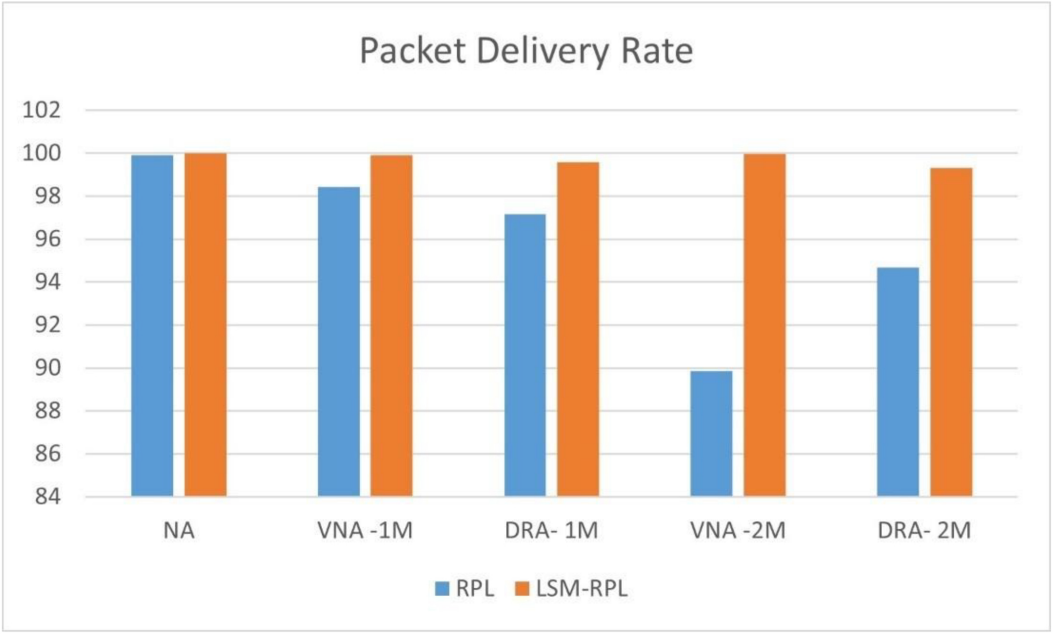

Fig. 12. PDR under different attack scenarios.

with one and two malicious nodes, maintaining more stable latency and jitter.

The results of Table 2 showed that increasing the number of malicious nodes significantly degrades network performance. However, the proposed LSM-RPL mitigates this damage noticeably and clearly. The impact of different security attacks on the packet reception rate over time shows in Fig. 11. The experiment lasted 1 hour (3600 seconds). It is observed that, the attack significantly impacted the number of packets received on the standard RPL protocol, especially when 2 malicious nodes were present. However, LSM-RPL succeeded wildly in avoiding the negative impact of the attack.

Fig. 12 shows the impact of security attacks on PDR over time. The network maintains a high and stable PDR near 100% without attacks. However, attacks significantly affect PDR: VNA exhibits the most disruptive effect, causing a substantial and worsening PDR drop. DRA shows a less severe decrease in PDR with some fluctuations. These results suggest the effectiveness of LSM-RPL in mitigating attacks and maintaining PDR. The comparison in Table 3 underscores the advantages of LSW-RPL compared to state-of-the-art research, particularly regarding its ability to mitigate security threats in WSNs. Table 3 provides a comparison of LSW-RPL with state-of-the-art research in the field. Where

Table 3. Comparison of LSW-RPL with state-of-the-art research in the field.

Features	18	2	17	28	RPL	LSM-RPL
Secure channel independence	✓	✓	X	✓	✓	✓
Use lightweight techniques	✓	X	X	✓	✓	✓
Protection against adversarial code manipulation	X	X	✓	X	X	✓
Secure against external attack	✓	X	✓	✓	X	✓
Secure against Internal attack	X	X	✓	X	X	✓
Provide source code	X	X	X	X	✓	✓

secure channel independence emphasizes that the protocol or system operates autonomously, regardless of the presence or absence of a secure communication channel, it represents an advantageous feature, particularly in IoT applications where establishing a secure channel between nodes can be challenging. The protection against adversarial code manipulation refers to the capability of a protocol to defend itself against deliberate attempts by adversaries to modify or tamper with its code in order to compromise the protocol's security or functionality.

However, while LSM-RPL has demonstrated robust performance in the scenarios tested, particularly in handling internal attacks, its performance in larger, more complex networks may be affected. Since LSM-RPL builds upon the RPL protocol, it inherits certain scalability limitations inherent in RPL. As network size increases, the additional processing overhead introduced by LSM-RPL's security mechanisms, combined with RPL's challenges in maintaining optimal routing in large-scale deployments, might lead to increased latency and energy consumption, potentially impacting overall network efficiency. This scalability aspect warrants further investigation, and future research should focus on optimizing LSM-RPL for larger IoT deployments, considering the balance between security and performance.

Future work

This study has demonstrated the effectiveness of the LSM-RPL protocol in mitigating version and rank attacks. However, several research areas need to further exploration to enhance the protocol's robustness and applicability in larger IoT networks.

- Scalability Analysis: Future research should focus on analyzing the impact of network scalability on LSM-RPL performance. As the network size increases, the processing overhead of hashing and authentication in proposed LSM-RPL protocol may also increase. Understanding how the network size influences the protocol's performance in large-scale networks is essential for optimizing LSM-RPL for large-scale networks.
- Exploration of Other RPL Attacks: This study specifically addressed version and rank attacks, but RPL protocol is susceptible to various other threats, such as black hole attacks, wormhole attacks and selective forwarding attacks. Future work should expand the scope of RPL potential attacks by modifying the control message headers to address other attacks using the same methods of detection and evaluation methods employed in this work.
- Intrusion Detection System (IDS) Integration: To enhance the proposed protocol and provide the ability to detect novel attacks, an IDS can be developed that operates in conjunction with LSM-RPL. Such a system could function at the network root, leveraging data sent from nodes. The IDS reports transmission between the root and other nodes could be signed with a private key to ensure integrity. This IDS could provide an additional layer of security to LSM-RPL.

Conclusion

The RPL routing protocol is the standard routing protocol from 2012 and is widely used in IoT networks. However, it suffers from numerous security vulnerabilities that effectively impact on its performance. These attacks make the enhancement of its security is very important to ensure resilience against these vulnerabilities. This paper introduces a secure operational mode for the RPL protocol that enhance the countermeasures against various security breaches. The performance of the proposed approach was evaluated using different evaluation metrics and simulation scenarios. Two widespread attack types were tested: version attacks and rank decrease attacks. The simulation results showed encouraging results in preventing common attacks and improving network performance.

Acknowledgment

The authors would like to thank the College of Engineering, University of Baghdad, for providing the

necessary facilities and support to conduct this research. They also express their gratitude to colleagues who offered valuable insights and feedback throughout the study.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm:
- All the Figures and Tables in the manuscript are ours. Any Figures and images, that are not ours, have been included with the necessary permission for re-publication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at University of Baghdad.

Authors' contribution statement

Z.T.A. and H. A. N. designed the study and developed the methodology. H. A. N. conducted the coding and simulation experiments. Both authors analyzed the data. H.A. N. drafted the manuscript. Z.T. A. revised the manuscript structure and performed the final review and error correction.

References

1. Rashid ZM, Karim BA, Al-Hindawi AMJ. Medical implantable antennas for IoT based health monitoring applications: A Review. J Eng. 2025;31(5):148–171. https://doi.org/10.31026/j.eng.2025.05.09.
2. Ambarkar SS, Shekokar N. An efficient authentication technique to protect iot networks from impact of rpl attacks. Int J Eng Trends Technol. 2021;69(10):137–145. https://doi.org/10.14445/22315381/IJETT-V69I10P217.
3. Rangarajan S, Al-Quraishi T. Navigating the future of the internet of things: Emerging Trends and Transformative Applications. BJIoT. 2023;2023:8–12. https://doi.org/10.58496/bjiot/2023/002.
4. Mohammed HJ. IoT-based low-cost smart health monitoring system using Raspberry Pi Pico W and Blynk Application. J Eng. 2024;30(07):90–108. https://doi.org/10.31026/j.eng.2024.07.06.
5. Abood ZA, Taher HB, Ghani RF. Detection of road traffic congestion using V2V communication based on IoT. Iraqi J Sci. 2021;62(1):335–345. https://doi.org/10.24996/ijs.2021.62.1.32.
6. Tariq U, Ahmed I, Khan MA, Bashir AK. Fortifying IoT against crimpling cyber-attacks: a systematic review. Karbala Int J Mod Sci. 2023;9(4):665–686. https://doi.org/10.33640/2405-609X.3329.
7. Brandt A, Hui J, Kelsey R, Levis P, Pister K, Struik R, *et al.* RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550. IETF. 2012:1–157. https://doi.org/10.17487/rfc6550.
8. Darabkh KA, Al-Akhras M, Zomot JN, Atiquzzaman M. RPL routing protocol over IoT: A comprehensive survey, recent advances, insights, bibliometric analysis, recommendations, and future directions. J Netw Comput Appl. 2022;207:103476. https://doi.org/10.1016/j.jnca.2022.103476.
9. Al-Amiedy TA, Anbar M, Belaton B, Bahashwan AA, Hasbullah IH, Aladaileh MA, *et al.* A systematic literature review on attacks defense mechanisms in RPL-based 6LoWPAN of Internet of Things. IoT. 2023;11:100741. https://doi.org/10.1016/j.iot.2023.100741.
10. Fadhil MS, Farhan AK, Fadhil MN. A lightweight aes algorithm implementation for secure iot environment. Iraqi J Sci. 2021;62(8):2759–2770. https://doi.org/10.24996/ijs.2021.62.8.29.
11. Bang A, Rao UP. Impact Analysis of Rank Attack on RPL-Based 6LoWPAN Networks in Internet of Things and Aftermaths. Arab J Sci Eng. 2023;48(2):2489–2505. https://doi.org/10.1007/s13369-022-07342-y.
12. Salman AA, Alisa ZT. Improving the Network Lifetime in Wireless Sensor Network for Internet of Thing Applications. Al-Khwarizmi Eng J. 2019;15(4):79–90. https://doi.org/10.22153/kej.2019.09.007.
13. Chiadighikaobi IR, Katuk N. A scoping study on lightweight cryptography reviews in IoT. Baghdad Sci J. 2021;18:989–1000. https://doi.org/10.21123/bsj.2021.18.2(Suppl.).0989.
14. H J FB, S S A. Survey on IoT Security: Attacks, Challenges and Countermeasures. Webology. 2022;19(1):3741–3763. https://doi.org/10.14704/WEB/V19I1/WEB19246.
15. Almusaylim ZA, Alhumam A, Jhanjhi NZ. Proposing a Secure RPL based Internet of Things Routing Protocol: A Review. Ad Hoc Networks. 2020;101:1–12. https://doi.org/10.1016/j.adhoc.2020.102096.
16. Medjek F, Tandjaoui D, Djedjig N, Romdhani I. Multicast DIS attack mitigation in RPL-based IoT-LLNs. J Inf Sec Appl. 2021;61:102939. https://doi.org/10.1016/j.jisa.2021.102939.
17. Nikravan M, Movaghar A, Hosseinzadeh M. A Lightweight Defense Approach to Mitigate Version Number and Rank Attacks in Low-Power and Lossy Networks. Wirel Pers Commun. 2018;99(2):1035–1059. https://doi.org/10.1007/s11277-017-5165-4.
18. Karmakar S, Sengupta J, Bit SD. LEADER: Low Overhead Rank Attack Detection for Securing RPL based IoT. In: Proc 2021 Int Conf Communication Systems & Networks (COMSNETS); Bangalore, India. 2021;429-437. https://doi.org/10.1109/COMSNETS51098.2021.9352937.
19. Alani MM, Awad AI. An Intelligent Two-Layer Intrusion Detection System for the Internet of Things. IEEE IEEE Trans Industr Inform. 2023;19(1):683–692. https://doi.org/10.1109/TII.2022.3192035.
20. Raoof A, Matrawy A, Lung CH. Enhancing Routing Security in IoT: Performance Evaluation of RPL's Secure Mode under Attacks. IEEE Internet Things J. 2020;7(12):11536–11546. https://doi.org/10.1109/JIOT.2020.3022276.
21. Agiollo A, Conti M, Kaliyar P, Lin TN, Pajola L. DETONAR: Detection of Routing Attacks in RPL-Based IoT. IEEE Trans Netw Serv Manag. 2021;18(2):1178–1190. https://doi.org/10.1109/TNSM.2021.3075496.
22. Ismail KS, Al-Juboori FA, Nasrullah MA. Efficient method to find the multiplicative inverse in GF (2m) using FPGA by exponentiation to (2k). In: 2012 Int Conf on Future Communication Networks. IEEE. 2012:159–163. https://doi.org/10.1109/ICFCN.2012.6206861.

23. Adarbah HY, Moghadam MF, Maata RLR, Mohajerzadeh A, Al-Badi AH. Security Challenges of Selective Forwarding Attack and Design a Secure ECDH-Based Authentication Protocol to Improve RPL Security. IEEE Access. 2023;11:11268–11280. https://doi.org/10.1109/ACCESS.2022.3221434.
24. Prathapchandran K, Janani T. A trust aware security mechanism to detect sinkhole attack in RPL-based IoT environment using random forest – RFTRUST. Comput Netw. 2021;198:108413. https://doi.org/10.1016/j.comnet.2021.108413.
25. Conti M, Kaliyar P, Rabbani MM, Ranise S. SPLIT: A Secure and Scalable RPL routing protocol for Internet of Things. 2018 14th Int Conf on Wireless and Mobile Computing, Networking and Communications (WiMob). IEEE. 2018:1–8. https://doi.org/10.1109/WiMOB.2018.8589115.
26. Glissa G, Rachedi A, Meddeb A. A Secure Routing Protocol Based on RPL for Internet of Things. 2016 IEEE Global Communications Conference (GLOBECOM). IEEE. 2016:1–7. https://doi.org/10.1109/GLOCOM.2016.7841543.
27. Essop I, Ribeiro JC, Papaioannou M, Zachos G, Mantas G, Rodriguez J. Generating Datasets for Anomaly-Based Intrusion Detection Systems in IoT and Industrial IoT Networks. Sensors. 2021;21(4):1528–1541. https://doi.org/10.3390/s21041528.
28. Newton PC, Samuel FJ. Secured Technique to Detect and Avoid Malicious Nodes in Internet of Things. Indian J Sci Technol. 2022;15(44):2363–2374. https://doi.org/10.17485/IJST/v15i44.1387.
29. Momand MD, Khan Mohsin M, Ihsanulhaq. Machine Learning-based multiple attack detection in RPL over IoT. In: Proc. of the 2021 International Conference on Computer Communication and Informatics. (ICCCI). 2021;Coimbatore, India:1–8. https://doi.org/10.1109/ICCCI50826.2021.9402388.
30. Algahtani F, Tryfonas T, Oikonomou G. A reference implementation for RPL attacks using Contiki-NG and COOJA. 2021 17th International conference on distributed computing in sensor systems (DCOSS). IEEE. 2021:280–286. https://doi.org/10.1109/DCOSS52077.2021.00053.
31. Jamil LS. Developing Blockchain Algorithms in the IoT network to secure data integrity and system scalability. Iraqi J Sci. 2024;65(6):3403–3418. https://doi.org/10.24996/ijs.2024.65.6.35.
32. Kareem AK, Shaban AM, Nafea AA, Aljanabi M, Aliesawi SAS, Mal-Ani M. Detecting routing protocol low power and lossy network attacks using Machine Learning techniques. 2024 21st International. Multi-conference. on systems, signals & Devices (SSD), Erbil, Iraq. 2024:57–62. https://doi.org/10.1109/SSD61670.2024.10549143.
33. Baghani AS, Rahimpour S, Khabbazian M. The DAO induction attack: Analysis and countermeasure. IEEE IOT J. Institute of Electrical and Electronics Engineers Inc. 2022;9(7):4875–4887. https://doi.org/10.1109/JIOT.2021.3108154.
34. Stallings W. Cryptography and network security principles and practice. Eighth edition. Pearson; 2022.
35. Oikonomou G, Duquennoy S, Elsts A, Eriksson J, Tanaka Y, Tsiftes N. The Contiki-NG open source operating system for next generation IoT devices. SoftwareX. 2022;18:101089. https://doi.org/10.1016/j.softx.2022.101089.
36. Kumar N, Jamwal P. Analysis of modern communication protocols for IoT applications. Karbala Int J Mod Sci. 2021;7:392–404. https://doi.org/10.33640/2405-609X.3165.
37. Jasim D, Omar K, Hamad A. Improving IoT applications using a proposed routing protocol. J Eng. 2014;20(11):50–62. https://doi.org/10.31026/j.eng.2014.11.04.
38. Althoubi A, Alshahrani R, Peyravi H. Delay analysis in IoT sensor networks. Sensors. 2021;21(11):1–17. https://doi.org/10.3390/s21113876.

اقتراح وتنفيذ بروتوكول توجيه آمن لشبكات إنترنت الأشياء باستخدام تقنية المصادقة الخفيفة

حسين عبد الصاحب نصرالله[1]، زينب توفيق آل عيسى[2]

[1] هندسة الحاسبات، كلية الهندسة، جامعة بغداد، بغداد، العراق.
[2] هندسة الكهرباء، كلية الهندسة، جامعة بغداد، بغداد، العراق.

الخلاصة

الانتشار الواسع لشبكات إنترنت الاشياء (IoT) واستخدامها الكبير في مختلف المجالات الحياة الصناعية والزراعية والعسكرية والصحية وغيرها، يرفع من أهمية تأمين بيانات هذه الشبكات ضد الانتهاكات والتهديدات الامنية المحتملة. على الرغم من أن بروتوكول RPL هو بروتوكول التوجيه القياسي المعتمد لشبكات انترنت الاشياء محدودة الموارد، إلا أنه لازال يعاني من عدد غير قليل من الثغرات الأمنية مما يجعل استخدامه خطيراً في العديد من التطبيقات وخاصة التطبيقات المهمة والحساسة. تقترح هذه الدراسة نمطاً تشغيلياً آمناً وخفيفًا لبروتوكول RPL تم تسميته الوضع الأمين الخفيف (LSM-RPL) لتوفير قاعدة عمل آمنة يمكنها التصدي لعدد غير قليل من الهجمات والتهديدات الامنية. يستخدم الوضع المقترح تقنية كود المصادقة المستند إلى التجزئة (HMAC) بالإضافة إلى نوعين من المفاتيح السرية لضمان المصادقة على العقد المتصلة بالشبكة وحجب العقد الخبيثة وكذلك ضمان سلامة حزم البيانات المرسلة من التلاعب اثناء انتقالها بين العقد. النوع الاول من المفاتيح السرية يكون مفتاح سري مشترك بين جميع عقد الشبكة يستخدم لحماية الحزم المرسلة إلى جميع الجيران، في حين أن النوع الآخر يكون فريد وخاص لكل عقدة يستخدم لحماية الحزم المرسلة إلى الجذر. يعتبر هذا النهج ذو المفاتيح المزدوجة إسهامًا جديدًا غير موجود في الأعمال السابقة، حيث يعزز كفاءة الاتصال والأمان. تناولت هذه الدراسة نوعين من الهجمات الشائعة، وهما هجوم الإصدار وهجوم خفض الترتيب. أظهرت نتائج المحاكاة لعدد من سيناريوهات الاختراق أداء جيدًا للطريقة المقترحة LSM-RPL مقارنة ببروتوكول RPL القياسي وكذلك مقارنة بأحدث تقنيات وبحوث تحسين أمان RPL المعروفة وقد تم التقييم من خلال عدد من معايير التقييم الشائعة. على سبيل المثال، في سيناريو هجوم خفض الترتيب بوجود عقدتين مخترقة، أظهر النمط الآمن المقترح LSM-RPL اداء جيد قياسا ببروتكول التوجيه القياسي RPL، حيث بلغت نسبة التحسين في نسبة تسلم الحزم المرسلة وتقليل استهلاك الطاقة وإجمالي الحزم المستلمة بحدود 10.1%، 45.9% و26.4% على التوالي.

الكلمات المفتاحية: التخفيف من حدة الهجوم، هجوم خفض الرتبة، رمز مصادقة الرسائل المستند إلى التجزئة، أمن المعلومات، إنترنت الأشياء (IoT)، توزيع المفاتيح، تقييم البروتوكول، بروتوكول التوجيه للشبكات منخفضة الطاقة والبيانات القابلة للضياع، تحسينات الأمان.