E2V: Techniques for Detecting and Mitigating Rank Inconsistency Attack (RInA) in RPL based Internet of Things

R Stephen¹, L Arockiam²

¹ Research Scholar, ² Associate Professor
¹, ² Department of Computer Science, St. Joseph’s College, Trichy, India.

¹ stephenr1989@gmail.com, ² larockiam@yahoo.co.in

Abstract: The Internet of Things (IoT) is a network of physical devices in which smart objects are interconnected that enable to collect and exchange the information via the internet. These devices are resource constrained and connected in the Low power and Lossy Networks (LLNs). Due to this nature, the RPL (IPv6 Routing Protocol for Low power and Lossy networks) is primarily designed for the resource constrained devices. But, this RPL undergoes various types of routing attacks. This paper considered the Rank Inconsistency Attack (RInA), which is illegitimately change the rank value and makes the network vulnerable. The proposed architecture E2V has three phases such as rank calculation, substantiation and malicious node elimination. The ultimate aim of E2V method is to detect and mitigate the RInA attack which includes sinkhole, selective forwarding and blackhole attacks. This system also identifies rank inconsistency based on the energy of each node. Hence, this approach enhances the secure routing in RPL based Internet of Things.

Keywords: Internet of Things, RPL, Rank Inconsistency attack, Energy

1. Introduction

In Internet of Things (IoT), the constrained devices are connected with IPv6 (Internet Protocol version 6) address over the Low Power and Lossy Networks (LLN) using standard routing protocols. Nowadays, IoT has become an emerging technology in a wide range of applications [1]. These include home automation, city management, healthcare, environmental monitoring etc. But, the major problem of IoT is that it is connected with unreliable networks due to the constrained resources such as the limited storage, memory and process. Routing in 6LoWPAN (IPv6 over Low Power Wireless Personal Area Networks) is classified into static and dynamic [2]. The distance vector protocols are used in dynamic environment and RPL is used in static environment.

RPL is an IP-based routing protocol for low power and lossy networks. It is primarily designed for the Internet of Things. RPL is a rank based DODAG (Destination Oriented Directed Acyclic Graph) tree topology protocol. Here, the source nodes select the set of parent nodes in which each node selects the preferred parent node based on better rank value [4]. In this case, an attacker acts as a parent node to broadcast fake rank value to its neighbor nodes. The nodes which select the attacker node become a compromised node to route the traffic through it. The attacker creates more harmful effects in the LLN when united with selective forwarding and black hole attacks. The RPL is susceptible by various internal and external attacks due to constrained resources. Particularly, internal attacks create more vulnerabilities and disrupt the network performances [5]. RPL construction is based on Rank metric. It is a challenging task to detect and mitigate the rank inconsistency attack. So, RPL needs a standard security mechanism to detect and mitigate these attacks.

In such scenario, the malicious node uses rank metric as a counterfeit to compromise to its neighbor nodes. The nodes which are compromised with the malicious node will lose the data packets. Generally, small
types of heterogeneous devices are connected in IoT network with constrained resources such as memory size, bandwidth, energy and computational ability. Studying the performance of the IoT network is essential to consider their limited resources. Among the constrained resources, energy is a vital resource to operate the devices or nodes, because most of the devices in the IoT network are battery powered. A node without sufficient resources is isolated and allowed for period to regain its exhausted resources, especially battery power, before it can be reconsidered for re-integration into the network. Since, most IoT nodes are energy constrained, using an energy conserving scheme will be beneficial to the operations of the IoT nodes. Also, a node with a high lossy link, very low bandwidth or a limited processing speed could extend delays in packet delivery [8].

Basically, RPL supports data confidentiality and integrity, also it has three basic security modes namely unsecured, pre-installed and authenticated. In unsecured mode, RPL control messages are sent without security mechanisms, they could be used as security primitives to meet the application requirements. In pre-installed mode, the nodes joining in a RPL instance, which have pre-installed keys that enable to process and protect the RPL messages. In authenticated mode, the nodes which have pre-installed keys as in pre-installed mode. These keys may only be used to join a RPL instance as a leaf node [3].

RPL can undergo attack from internal wireless sensor networks and external internet environment. The proposed Energy based Verification and Validation (E2V) techniques focused on internal attack. The traditional security solutions could be optimized to lightweight design due to the limited resource constrained devices [9]. This research focused to detect and mitigate the Rank Inconsistency Attack (RInA) which makes un-optimized path and formation of loop. Consequently, RInA affects the network performances in the way to decrease the packet delivery ratio, packet delay and reduces throughput. Hence, this paper proposes E2V techniques to detect and mitigate Rank Inconsistency Attack (RInA) in RPL based Internet of Things.

2. Objective of this paper

The primary objective of this paper is to provide a secure routing framework that incorporates two major processes namely validation and verification. These techniques are used here to detect the malicious nodes which have counterfeit rank value. The following are the approaches to achieve the objective.

- To develop an architectural framework for providing secured routing in RPL based IoT environment.
- To give a brief explanation for the methodology which is used in the proposed architecture.
- To explain the significance of the proposed work.

3. Rank Inconsistency Attacker (RInA) Model

The RPL is an IP- based routing protocol which is primarily designed for the Internet of Things. It is a network which is connected with constrained devices with limited power, memory and process. Due to constrained resources, the intruder launches different types of attacks to disrupt the routing path in the network. This chapter works on Rank Inconsistency Attack (RInA), due to its effectiveness in the RPL based Internet of Things. The rank is an integer value which determines the position of each node with respect to the root node. In RInA attack, a malicious node attracts the neighbor nodes with its counterfeit rank value which advertises that it’s closer to the root node. The compromised nodes route the traffic through this malicious node. Hence, a non-optimized path is created when a node selects the malicious node in the context of fake rank value. The RInA attack becomes more vulnerable when it is combined with sinkhole attack, selective forwarding attack and blackhole attack. In RPL DODAG, each node selects the preferred parent node from the set of nodes based on their best rank value. In which, the attacker node misuses the rank value as a counterfeit to create an inconsistency in the network. Here, the intruder launches an attack by propagating its rank as a better rank value and compromises other nodes by making them to select the malicious node as a preferred parent node. Hence, the malicious node either drops or selectively forwards certain data packets to the destination.
In figure 1, the node J is a malicious node and it attracts the nearby nodes H, I, K and L with its false rank value (R=2). But, its actual rank value is (R=5). Now, the nodes H, I, K, and L select the malicious node J as a preferred parent node and route their data packets through it. After the DODAG construction, node K is the parent of node L, node I is the parent of K, node F is the parent of nodes H and I. In a short period of time, the attacker node J advertises its false rank value as (R=2), which depicts its position to be nearer to the root node R and due to this H, I, K and L disconnect their existing paths and compromises with attacker node J, whereas according to RPL rules node J’s rank value is (R=5), which when compared to existing parent nodes is very low. These types of problems are defined as a Rank Attack (RA) or Rank Inconsistency Attack (RInA).

4. Impact of attacker node with false energy value

Energy is one of the major objectives which is used to calculate the node’s rank value in the RPL DODAG. In RPL network, each node has the energy (Joules) for receiving and transmitting the data packets to the destination. But, the energy level is not equal for all the nodes. The nodes consume certain level of energy for data transmission. Its energy consumption is based on idle time, transfer time and listen time of RPL operations. Periodically, the root node requests all the nodes to send their DIO (DODAG Information Object) messages which contain node ID, rank, current energy, version number etc. Here, each node calculates rank value with its current energy. But, an attacker node calculates rank value with its counterfeit energy level. Hence, an attacker node sends false rank value to the root node. The RPL network is a dynamic, so the nodes change their position and the new rank value is calculated. Due to this constraint, the proposed method E2V validates and verifies each node’s rank value based on their energy level (Joules).
Figure 2 shows the impact of attacker node with false energy level. Here, node E is an attacker node which has the false energy level (250J). In RPL network, the maximum energy level of entire DODAG is based on the root node R (255J). This node periodically receives energy, and thus these nodes are responsible to send their current energy to the root node ‘R’. In figure 2, the nodes A, B and C send their initial energy as 220J, 215J and 100J respectively. For a period of time, the nodes A, B and C consume certain level of energy for transmission of data packets such as 10J, 5J and 5J respectively. Now, these nodes A, B and C have current energy 210J, 210J and 95J respectively. The attacker node E’s initial energy is 217J. It consumes 10J for data transfer, then it has 207J as the current energy. But, it sends DIO message without updating current level of energy as 217J. Also, it calculates a new rank value with its false energy. Consequently, the root node identifies the attacker node E which has false energy and rank. As an effect, the nodes F, G and H become compromised nodes to create an inconsistency in the RPL DODAG.

Table 1. Root Node Detection Status based on Node’s Energy

| Node ID | Initial Energy (in Joules) | Energy Consumption (in Joules) | Current Energy (in Joules) | Root node (E2V) Detection Status |
|---------|-----------------------------|-------------------------------|-----------------------------|---------------------------------|
| A       | 220J                        | 10J                           | 210J                        | Valid                           |
| B       | 215J                        | 5J                            | 210J                        | Valid                           |
| C       | 100J                        | 5J                            | 95J                         | Valid                           |
| D       | 210J                        | 5J                            | 205J                        | Valid                           |
| E       | 217J                        | 10J                           | 217J                        | Invalid                         |
| F       | 210J                        | 10J                           | 100J                        | Valid                           |
| G       | 240J                        | 10J                           | 230J                        | Valid                           |
| H       | 250J                        | 10J                           | 240J                        | Valid                           |

Table 1 shows the E2V detection status which is deployed in the root node R. Here, the root node R periodically receives DIO message from all the nodes. Then, it validates and verifies the corresponding information of each node. However, the node E has an invalid energy level. The root node R detects the node E is an attacker node. The following section 5 describes the working procedure of E2V methodology against the rank inconsistency attack in RPL based Internet of Things.

5. Structure of E2V Architecture
In this E2V (Energy based Validation and Verification) architecture, three mechanisms are integrated for analyzing the performance of Intrusion Detection System (IDS) and to provide a secure communication in IoT environment. The three mechanisms are rank calculation, substantiation and malicious node elimination. All these three mechanisms are deployed by different procedures. The aim of the architecture is to provide secure routing in RPL based IoT environment.
The structure of the E2V architecture and its process is depicted in figure 3. Each proposed mechanism has different procedures for specific purposes. The ultimate aim of this architecture is to detect the malicious nodes which create rank inconsistencies in RPL protocol and provide a secure communication by using validation and verification techniques. The proposed method is used to identify the rank inconsistency attack in Internet of Things environment. The node’s energy is a key factor to detect the malicious node in a low power and lossy network which is connected through IPv6. The following section describes the E2V architecture mechanisms.

5.1 Rank Calculation based on Energy Metric
The rank of a node is calculated by Expected Transmission Count (ETX), Hop-Count and Energy. The RPL uses the rank metric for each node to avoid loop in the network. The nodes record its relative position to other nodes with regard to DODAG root. In RPL, the rank value monotonically decrease in upward direction and increase in downward direction. The nodes in the RPL network select their preferred parent using their rank value. Here, two rank calculations are involved namely Self-Rank (SR) and Parent Rank (PR). The SR determines the child node’s rank value. A child node computes its own rank from preferred parent’s rank as defined in equation (1). The proposed work uses the energy metric for rank calculation. The parent node sends DODAG Information Object (DIO) message to all its child nodes. The DIO message contains parent node ID, rank and energy. The child nodes compute their own rank from its preferred parent node’s rank. After computing rank value, the child node sends DODAG Acknowledgement Object (DAO) message to their preferred parent node. The DAO message contains child node’s rank and current energy. The initial energy of a child node is known by its preferred parent node. Initially, every node has an energy, after transmission of data packets its energy will be reduced. The DODAG root determines the MinHopRankIncrease between nodes.

\[ SR = PR + \text{Rank}_{\text{increase}} \]  

Where,

\[ \text{Rank}_{\text{increase}} = \text{credit} + \text{MinHopRankIncrease} \]

\[ \text{Credit} = \frac{\text{Available energy}}{\text{Initial energy}} \]
\[ CR = PR + \text{Rank}_{\text{decrease}} \]  

Where,  
\[ \text{Rank}_{\text{decrease}} = \text{Energy Consumption} + \text{MinHopRankIncrease} \]

Where,  
\[ \text{Energy Consumption} = (\text{packets sent} \times \text{required energy}) + (\text{ideal time} \times \text{ideal require energy}) \]

5.2 Substantiation

The attacker node is identified in the substantiation phase. IPv6 Border Router (6BR) is a root node which is responsible for maintaining the network operations. The 6BR periodically collects information about the topology from all the nodes. The child and parent nodes send the control message to 6BR with required information. The rank information received from neighboring nodes are validated and verified. The validation of child node’s rank is accomplished by cross verification of the received rank information. The child rank (CR) is calculated by the preferred parent as explained in 5.1. The 6BR verifies the rank value of CR and PR nodes where each node sends its own rank as well as its neighboring nodes rank value. Hence, the 6BR validates the node’s rank value which is received from all the nodes. If the 6BR finds any contradiction in the node’s rank value, it detects it as a malicious node and it will get stored in the black-list. This also sends an alert message to the remaining nodes by which the malicious node do not travel to the next transmission.

The E2V mechanism is deployed in the 6BR node that is employed in a centralized approach to monitor the behavior of nodes in the network through the validation and verification mechanisms.

5.2.1 Validation

This strategy is used to compute the rank value for each node based on energy. It is defined in section 5.1. The 6BR has the routing table to maintain the routing information of each node. This information includes node Id, rank and energy. The ultimate aim of this method is to validate the rank and energy of each node.

5.2.2 Verification

This strategy is used to verify the rank value of each node which is received from parent nodes. This verification process is based on energy metric. Periodically, the 6BR receives the current energy of a node and also 6BR knows the initial energy of a node. The following formula (3) is used to verify the energy value (\( V_E \)) by using energy metric. In RPL, each node has an energy for transmission of data packets. But, it is not same for all the nodes. Now, the nodes spend some joules for data packets transmission, then every node will send the current energy when the 6BR is required. Perhaps, the malicious nodes send the energy as a counterfeit value. The 6BR verifies the current energy with the existing data of a node. If it is contradicting, then it will be considered as a malicious node.

\[ V_E = \text{Initial energy} - \text{Current energy} \]  

5.3. Malicious Node Elimination

The malicious node elimination is the key feature in intrusion detection system. The dynamic change is executed in the network by removing the attacker node is termed as network repair or routing repair. In RPL, there are two types of repairs carried out to eliminate the attacker node. The E2V mechanism has the ability to repair the RPL topology by using these two types of RPL repair mechanisms. These are (i) local repair and (ii) global repair. These two mechanisms are used for a routing protocol to dynamically update routing decisions and adapt the network topology when a node or link failure happens. In local repair, the network change impacts on a specific part of the network (Sub DAG). In global repair, the
removal of an attacker node impacts the whole network (DODAG). The global repair is triggered by the root node which involves in additional control overhead.

5.3.1. Local repair

The E2V mechanism detects network inconsistency and local loops between nodes. In such case, the E2V uses local repair mechanisms to provide an alternate path. The E2V mechanism is deployed in the IPv6 Border (6BR) router or root node. In local repair, the 6BR node detects the particular malicious node and sends the best optimum path to the remaining nodes.

5.3.2 Global repair

The local repair mechanism is not sufficient for several network inconsistencies. Hence, the E2V requires the global repair mechanism in the context of multiple inconsistencies. In global repair mechanism, the entire topology will be reconstructed and increments the new DODAG version number. Consequently, the nodes compute their rank value and choose a new position in the DODAG. In global repair, the 6BR node repairs the entire network in the context of multiple network inconsistencies.

6. E2V: Proposed Algorithm

In the IoT network, the attacker node targets any individual node or its neighbor node’s rank to make it as a compromised node. Then, the attacker node uses it to launch multiple attacks and sends wrong routing information. The view of this critical issue in the network leads to rank inconsistency. Therefore, it is important to detect and correct the rank inconsistency in RPL based networks. The complete rank inconsistency detection algorithm is described in algorithm.

Algorithm Construct_DODAG

Input: n nodes
Output: DODAG Tree

1 Begin
2 Initialization:
3 \( U_n \rightarrow \) set of nodes not connected in DODAG
4 \( C_n \rightarrow \) set of nodes connected to DODAG
5 Set the Hop_Count of all nodes \( i \) in \( U_n \) as 0
6 // join the Root node into the DODAG and remove it from \( U_n \)
7 \( C_n \leftarrow C_n \cup \) Root and \( U_n \leftarrow U_n \setminus \) Root
8 repeat
9 for each node in \( n \) do
10 { Send DIO message to all node along with Rank \( R_k \)
11 Receive DAO message from all child nodes
12 node(i).Hop_Count=node(i).Hop_Count+1
13 Send DIS message from the newly joined nodes
14 for all node \( j \) in \( n \) do
15 if(node(i).msg==DIO) then
16 \( i \leftarrow \) node in \( U_n \) which is nearest to Root
17 \( C_n \leftarrow C_n \cup \{ i \} \)
18 \( P \leftarrow \) parentSelection \((i, C_n)\)
19 if \((P \neq \phi \text{ or } R \in P)\) then
20 }
21 }
22 }
In order to detect the rank inconsistency and to make sure that the node’s rank is consistent across the network, the 6BR checks each node’s rank and energy and compares the same information with its neighbor node’s information and detect the inconsistencies. The 6BR has the ability to raise a false alarm when the detection of node’s rank is inconsistent. The E2V algorithm is not only to find the rank inconsistency, but also it has the ability to distinguish between the valid and invalid node’s energy level which will avoid inconsistencies in the RPL DODAG. Hence, the E2V algorithm uses the credit of energy as a parameter to find the faulty node which has faulty information.

The E2V algorithm rely on two conditions: (i) checks the reported value of individual node’s rank and energy, (ii) compares the same reported value with its neighbor nodes. The 6BR corrects the faulty information when the above two conditions lead to contradicting values which cause RPL inconsistency
problem. The 6BR reports the faulty information to its neighbors with faulty node’s ID, parent list and version number. The neighbor nodes update their routing table regarding the 6BR information. Once the 6BR detects a node with rank inconsistency or energy inconsistency, it will remove the faulty node from its white list. At first, the 6BR updates the faulty node in the white list and sends the report about the attacker node to its neighbors. If 6BR detects the same node again as a malicious node then it will remove it from the white list.

7. Simulation Results

The proposed mechanism E2V is evaluated in the Contiki OS and it is a well-known operating system for the Internet of Things. The Cooja simulator is used to test the RPL implementation. The E2V is primarily designed to detect routing attacks, particularly to detect and mitigate the Rank Inconsistency Attack (RInA). The E2V mechanism uses the RPL implementation in the Contiki operating system to develop the detection modules in the RPL network. The nodes in the RPL network are resource constrained except that of the border router which is not a constrained node. The proposed method E2V is deployed in the root node. The modules have the ability to validate and verify the node’s behavior by using substantiation phase.

Generally, the Intrusion Detection System (IDS) is used to track the behavior of the nodes in the wireless sensor networks. The RPL implementation with E2V mechanism is employed in network routing where each node keeps track of all its control messages. The E2V also implements against the sinkhole attack, blackhole attack and selective forwarding attack in the RPL network. This section presents the evaluation of E2V after describing the experimental setup. This evaluation is based on the network parameters such as network convergence delay, energy consumption and attacker identification delay. The experiments with Contiki network simulator Cooja has proved to produce realistic results. The Cooja simulator runs deployable Contiki code. The E2V experiments used Tmote Sky nodes for three kinds of nodes, they are sink node, sender node and border router node which is shown in the table 2.

| Parameters                  | Description                                      |
|-----------------------------|--------------------------------------------------|
| No. of nodes                | 200, 100, 100                                    |
| Simulation Area             | 1000 x 1000 m                                   |
| Node Arrangement            | Random, Grid                                    |
| Radio Medium                | UDGM-Distance Loss                              |
| Operating System            | Contiki                                          |
| Simulator                   | Cooja                                            |
| Routing Protocol            | RPL                                             |
| Network Protocol            | IP based                                        |
| Types of sensor node        | Tmote Sky(Sink), Tmote Sky(Sender), Tmote Sky(Border router), |
| Packet Analyzer             | Wireshark                                       |

7.1 Convergence Delay

The E2V measures the convergence delay with various number of nodes. The convergence delay refers to the RPL network convergence after rectifying the attacker node. Figure 4 shows the comparison of convergence delay with the existing INTI (Intrusion detection of sinkhole attacks on 6LoWPAN for Internet of Things) method. Here, the measurement used 50, 100, 150 and 200 nodes. The E2V reduced the convergence delay almost 419ms while the INTI had 499ms of delay.
7.2 Energy Consumption

After the attacker is triggered, the E2V mechanism detects and mitigates the attacker node. Figure 5 shows the energy consumption with various attacker ratio (%). It is simulated with 100 nodes with 40% of attacker ratio. The E2V reduced energy consumption to 381J while the existing INTI took 394J.

7.3 Attacker Identification Delay

The E2V is tested with attacker identification delay parameter and is also compared with two existing systems LRPL (Loop free routing protocol for low power and lossy networks) and INTI. Figure 6 shows the result of attacker identification delay. It is evaluated with 10, 30, 50 and 100 number of nodes. The E2V reduced the attacker identification delay approximately by 2215ms while the existing systems LRPL and INTI achieved it in 2430ms and 2714ms respectively.
8. Conclusion

The proposed E2V mechanism is used to detect the false rank node in RPL topology. The rank value is the position of each node with respect from the root node. In RPL, each node calculates its own rank and selects its preferred parent node. An attacker node claims that its rank value is lower than the true rank node in order to choose it as a parent to collect packets and later tampers it. The proposed E2V mechanism is deployed in the root node which detects the attacker node. In this work, E2V architecture for the Internet of Things is proposed with respect to the energy metric. In RPL, an attacker can exploit the energy metric and launch different attacks by getting a better position in the RPL DODAG. To detect these attacks and also to find the malicious nodes, this work has developed energy based intrusion detection system. This E2V system can resist against rank attacks called rank inconsistency attack (RInA). In RPL, various types of parameters are used to select the best optimum routes to the destination. The proposed E2V method is compared with the existing systems based on selected network parameters. Thus, the E2V reduced the energy consumption, attacker identification delay and network convergence delay when compared with existing methods and the results are tabulated.

References

[1] Kopetz and Hermann, "Internet of things", Real-time systems, Springer US, 2011, pp.307-323.
[2] Mbarushimana C, and Shahradi A, “Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks”, In Advanced Information Networking and Applications Workshops, Vol. 2, 2007, pp. 679-684.
[3] Gaddour Olfa and Anis Koubâa, "RPL in a nutshell: A survey", Computer Networks, Vol. 56, Issue 14, 2012, pp. 3163-3178.
[4] Hummen René, Hanno Wirtz, Jan Henrik Ziegeldorf, Jens Hiller, and Klaus Wehrle, "Tailoring end-to-end IP security protocols to the Internet of Things", 21st IEEE International Conference on Network Protocols (ICNP), IEEE, 2013, pp. 1-10.
[5] Guo Jianlin, Chuan Han, Philip Orlik, Jinyun Zhang, and K. Ishibashi, "Loop-free routing in low-power and lossy networks", In SENSORCOMM 2012, The Sixth International Conference on Sensor Technologies and Applications, 2012, pp. 59-66.
[6] Landsmann Martin, Matthias Wahlsisch, and Thomas C. Schmidt, "Topology authentication in rpl", In Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on IEEE, 2013, pp. 73-74.
[7] Roman Rodrigo, Jianying Zhou and Javier Lopez, "On the features and challenges of security and
privacy in distributed internet of things", *Journal on Computer Networks*, Vol. 57, No. 10, 2013, pp. 2266-2279.

[8] Airehrou David, Jairo Gutierrez, and Sayan Kumar Ray, "A testbed implementation of a trust-aware RPL routing protocol", 27th International Telecommunication Networks and Applications Conference (ITNAC) on IEEE, pp. 1-6, 2017.

[9] Pongle Pavan, and Gurunath Chavan, "A survey: Attacks on RPL and 6LoWPAN in IoT", *IEEE International Conference on Pervasive Computing (ICPC)*, 2015, pp. 1-6.