Proceedings of APCC2015 copyright © 2015 IEICE 14 SB 0087

Secure Parent Node Selection Scheme in Route Construction to Exclude Attacking Nodes from RPL Network

Kenji Iuchi, Takumi Matsunaga, Kentaroh Toyoda and Iwao Sasase
Dept. of Information and Computer Science, Keio University
3-14-1 Hiyoshi, Kohoku, Yokohama, Kanagawa, 223-8522 Japan,
Email: matsunaga@sasase.ics.keio.ac.jp

***Abstract*—The IPv6 Routing Protocol for Low-power and Lossy networks (RPL) is a standard routing protocol to realize the Internet of Things (IoT). Since RPL is a tree-based topology network, an attacking node may falsely claim its rank towards neighbor nodes in order to be chosen as a parent of them and to collect more packets to tamper. In this paper, we propose a secure parent selection scheme so that each child node can select a legitimate node as its parent. In the proposed scheme, each node chooses a parent after excluding the best candidate if multiple parent candidates exist. Our scheme utilizes the fact that an attacking node claims falsely a lower rank than that of a legitimate nodes. We show that attacking nodes have no merits to claim lower ranks than true ones in a secure parent node selection scheme. By the computer simulation, we show that the proposed scheme reduces the total number of child nodes attached to attacking nodes in comparison with the conventional RPL scheme.**

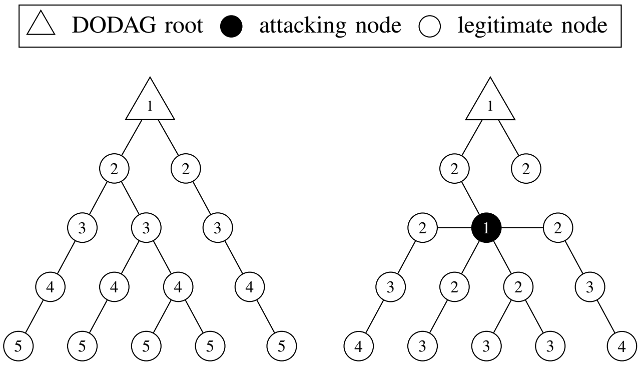
I. Introduction

In recent years, the Internet of Things (IoT) is getting attractive due to increasing devices connected to the Internet [1]. In the IoT, resource-constrained sensing devices are connected to the Internet via IPv6 networks so as to monitor and control everything, e.g., energy consumption of appliances or everlasting structure monitoring. The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), which constructs a tree-based topology network, is selected as a standard routing protocol to realize the IoT [2]. In RPL, nodes create a loop-free tree-based topology network which is called as Destination-Oriented Directed Acyclic Graph (DODAG) to communicate each other. In order to construct DODAG, each node has a rank which is a cumulative value, e.g., the number of hops from the DODAG root and strictly increases from the DODAG root to leaf nodes. The DODAG root and nodes periodically broadcast a DODAG Information Object (DIO) message to inform their ranks. Each node selects the least rank neighbor node as its parent node.

In RPL, security is important since RPL is adopted for smart grids, industrial automation, smart cities, building automation, and structural health monitoring [3]. The communication in RPL is protected on an end-to-end communication to use IPsec protocol [4]. Authentication Header (AH) protocol in IPsec ensures the authentication and the integrity of application data and IPv6 headers. Encapsulating Security Payload (ESP) protocol in IPsec ensures of the confidentiality application data in addition to AH protocol. Therefore, attacks by external attacking node from a third party can be protected. External attacking nodes do not participate in the routing and make interception or falsification. On the other hand, internal attacking nodes can claim fake rank that is lower rank than true rank to collect more packets from child nodes [5][6]. Internal attacking nodes participate in the routing and generate fake messages or drop packets. In order to detect internal attacking nodes, Dvir et al. propose Version number and Rank Authentication in RPL (VeRA) to protect from attacks claiming lower rank by internal attacking nodes [7]. VeRA ensures ranks to increase strictly from the DODAG root to leaf nodes by utilizing one-way hash chain. Each node authenticates neighbors' rank by calculating hash chain repeatedly. However, computation complexity is increased for resource limited node. Raza et al. propose real-time intrusion detection scheme in RPL called SVELTE and Perrey et al. propose topology authentication in RPL called TRAIL to detect internal attacking nodes by finding rank inconstancy [8][9]. In SVELTE, the DODAG root judges the node as an attacking node if its rank is lower than its parent node by collecting information such as neighbor and parent rank from each node. In TRAIL, the parent node judges its child node if its child rank is lower than its own rank. However, the conventional schemes have a problem that a child node selects an attacking node as its parent since a child node does not judge whether its parent node is attacking node or not.

In this paper, we propose a secure parent node selection scheme so that each child node can select a legitimate node as its parent. In the proposed scheme, each node chooses a parent after excluding the best candidate if multiple parent candidates exist by utilizing the fact that an attacking node intends to claim falsely a lower rank than that of a legitimate node. Each node can judge whether rank values that its neighbor nodes broadcast are too low since it can obtain a maximum and average rank of its neighbor nodes. After that, each node selects its parent node except for nodes whose rank is judged to be too low. Therefore, each node avoids selecting an attacking node as its parent node and sending packets to the attacking node.

We evaluate the total number of child nodes attached to attacking nodes. As a result, we show that the proposed scheme reduces the total number of child nodes attached to attacking nodes in comparison with the conventional RPL scheme and that the attacking nodes have no merits to claim lower ranks than true ones so as to collect more packets.



(a) Non attacking node in topology. (b) An attacking node in toppology.

Fig. 1. An example of a network topology that an attacking node exists. Each number indicates rank value.

The main contributions of this paper are three folds: (1) we point out that a child node selects an attacking node as its parent in the related work. (2) we implement and evaluate the proposed secure route construction scheme. (3) we show that the proposed scheme reduces the total number of child nodes attached to attacking nodes in comparison with the conventional RPL scheme.

The organization of the paper in the following: the operation of RPL and attacking nodes are provided in Section II. In Section III, we summarize merit and demerit of related works. In Section IV, the proposed secure route construction scheme is presented. In Section V, the total number of child nodes attached to attacking nodes is evaluated. In Section VI, we conclude the paper.

II. Preliminaries

A. RPL

In RPL, the nodes create a loop-free tree-based topology which is called DODAG to communicate each other. In order to construct DODAG, each node has a rank that is cumulative value. Rank value is increased strictly from the DODAG root to leaf nodes. The simplest rank is calculated from the number of hops from the DODAG root and this calculation methodology is called Objective Function 0 (OF0) [10]. The routes from the DODAG root to leaf nodes are constructed by DIO messages initiated by the DODAG root. DIO messages include a rank of node that broadcasts a DIO message. Each node selects a parent based on the DIO messages received from its neighbors so that its rank is lowest. The routes from leaf nodes to DODAG root are constructed by Destination Advertisement Object (DAO) messages. Each node sends a DAO message to the parent. The DODAG root broadcasts a DIO message periodically to form the optimum route.

B. Attacker Model

In this paper, we assume that internal attacking nodes can claim fake rank that is the lower rank than the true rank to collect more packets from child nodes. Attacking nodes may drop, collapse, or tamper collected packets, and the attacker model is based on [11]. We show the example of network topologies (a) when no attacking node exists (in Fig. 1(a)) and (b) when an attacking node exists in attacking nodes that collects more packets (in Fig. 1(b)). As we can see from Fig. 1(a) and Fig. 1(b), when an attacking node intends to claim falsely lower rank than its neighbor node, he/she is more likely to be chosen as a parent. Since each node selects the lowest rank node as its parent in RPL, the attacking node can collect more packets by claiming lower rank to its neighbors.

III. Related Work

In this section, we summarize novel attacking node detection schemes in RPL merits and demerits of them.

A. VeRA

VeRA avoids attacking nodes from claiming lower rank than true rank by utilizing one-way hash chain. One-way hash chain is successive application of hash function, e.g. $h(h(h(h(x))))$, denoted $h^4(x)$ and thus each node calculates $h^4(x)$ from $h(x)$, however, it cannot calculate $h(x)$ from $h^4(x)$. Therefore, one-way hash chain can be used to ensure rank to be increased strictly from the DODAG root to leaf nodes. Each node authenticates neighbors' rank by calculating hash chain repeatedly. The DODAG root generates a random number r, and calculates hash chain $h_{check} = h^{R_{lim}}(r)$, where R_{lim} is the maximum rank value in the DODAG. VeRA assumes that each node knows the hash chain value $h^{R_{lim}}(r)$ and R_{lim} since the DODAG root sends this value in advance. When a node sends a DIO message, a node sends $h_{sender} = h^{R_{self}}(r)$ including its DIO message, where R_{self} is a node's rank itself. Receiving a DIO message, each node checks if $h^{R_{lim}-R_{sender}}(h_{sender}) = h_{check}$, where R_{sender} is the sender's rank. If $h^{R_{lim}-R_{sender}}(h_{sender}) \neq h_{check}$, each node considers the sender as an attacking node.

B. SVELTE

SVELTE detects attacking nodes by finding rank inconsistency in the DODAG root. The DODAG root judges the node as an attacking node if its rank is lower than its parent node since the rank of parent node must be lower than that of its child nodes. Therefore, as the first module, the DODAG root requests every node to report its own rank and neighbor nodes' ranks. Receiving a request, each node responds with its neighbor and parent ranks. As the second module, the DODAG root analyzes the collected data and detects attacking nodes. The DODAG root checks each node's rank inconsistency by comparing the rank that it insists with the rank that its neighbors report. The DODAG root judges that a node is an attacking node if the difference between the ranks is larger than the pre-defined threshold.

C. TRAIL

TRAIL detects attacking nodes by finding rank inconsistency in each parent node instead of the DODAG root. A parent node judges its child node as an attacking node if its child rank is lower than itself since the rank of parent node must be lower than that of its child nodes. A child node sends its rank to its parent node so as to verify that its rank is honest. Each parent node verifies whether the two conditions are satisfied. The first one is whether the rank in the message is higher than its own. The second one is whether the rank that sender broadcasts lies in between the rank in the message and its own. If any

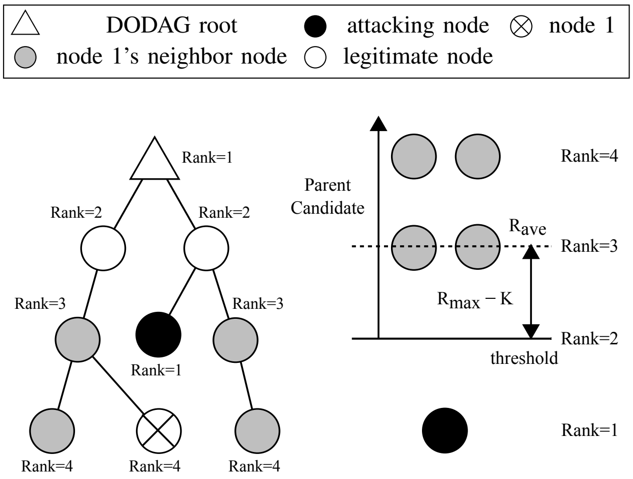


Fig. 2. An example of the route construction phase.

condition is not fulfilled, the parent node considers its child node as an attacking node.

D. Merits and Demerits of Related Work

VeRA can strictly detect attacking nodes by utilizing one-way hash chain. However, VeRA has two problems. The one is computation complexity is increased for resource limited node. The other one is vulnerability of VeRA. VeRA is vulnerable to hash chain forgery attack and replay attack. In order to solve this problem, Landsmann et al. proposed to add a nested encryption to VeRA [12]. However, the scheme is computationally complex. Therefore, VeRA cannot be applied to resource limited nodes, e.g., sensor nodes. SVELTE has a merit that each node is lower computation complexity than VeRA. However, SVELTE has two problems. The one is false detection is high. In order to mitigate this problem, Matsunaga et al. propose a low false alarm attacking nodes detection scheme [13][14]. The other one is that the DODAG root has to report the attacking node's information to each node but it is questionable that such information is correctly distributed under the existence of attacking nodes. TRAIL has merits that the computation complexity is lower than VeRA and that each node does not need to send neighbor nodes' information to the DODAG root. However, TRAIL has a problem that a child node might select an attacking node as its parent.

IV. Proposed Scheme

Here, we propose a secure parent node selection scheme so that each child node can select a legitimate node as its parent. At the route construction phase, each node chooses a parent after excluding the best candidate if multiple parent candidates exist by utilizing the threshold. Each node can judge whether rank value that its neighbor nodes broadcast is too low since it can obtain a maximum and average rank of its neighbor nodes. This notion comes from the fact that attacking nodes intend to claim falsely lower rank than legitimate nodes. After that, each node selects its parent node except for nodes whose rank is judged to be too low. Therefore, each node avoids selecting an attacking node as its parent node and sending packets to the attacking node.

A. Algorithm

In the proposed scheme, each node i calculates its own threshold $threshold_i$ with the maximum and average rank of its neighbor nodes. $threshold_i$ is calculated as follows

$$threshold_i = R_{ave} - R_{max} \times K \tag{1}$$

where, R_{ave} is an average of its neighbor node's rank, R_{max} is a maximum rank of neighbor node's rank, and K $(0 < K < 1)$ is a constant parameter, respectively. If its neighbor node rank is lower than $threshold_i$, it judges the neighbor node as an attacking node and excludes from parent candidates. Then, it selects a node that is the lowest rank in its neighbor nodes except for the attacking node.

We show an example of the parent selection route phase in Fig. 2. Node 1 calculates $threshold_i$ with its neighbor ranks. In this case, $R_{ave} = (1+3+3+4+4)/5 = 3$ and $R_{max} = 4$. If $K = 0.25$, $threshold_i = 3-4\times 0.25 = 2$. Node 1 excludes a node whose rank is lower than $threshold_i$ from its parent candidates since 1 (= attacking node's falsely claimed rank) $<$ 2 (= $threshold_i$). Therefore, in the proposed scheme, node 1 avoids selecting an attacking node as its parent.

B. Threshold

If an attacking node is far from the DODAG root, attacking node's rank is much smaller than legitimate node's rank. Each node can avoid selecting an attacking node as its parent even if $threshold_i$ is lower than legitimate node's rank since its rank is high. If an attacking node is near from the DODAG root, attacking node's rank is slightly smaller than legitimate node's rank. Each node can avoid selecting an attacking node as its parent only if $threshold_i$ is slightly lower than legitimate node's rank since its rank is close to attacking node's rank. If R_{max} is large, a node is far from the DODAG root. If R_{max} is low, a node is near from the DODAG root. Therefore, each node calculates $threshold_i$ with $R_{ave} - R_{max} \times K$.

C. Parameter K

If the parameter K is too small, the value of $threshold_i$ gets high and thus each node avoids selecting not only attacking nodes but also legitimate nodes as its parent. As a consequence, detour may occur and the number of hops from each node to the DODAG root may increase. If the parameter K is too large, the value of $threshold_i$ gets low and thus each node may select an attacking node as its parent. Thus, the probability of avoiding attacking nodes from a parent is decreased. Therefore, we need to determine the parameter K to take the number of hops and the probability of avoiding an attacking node into consideration. $threshold_i$ is expanded as follows

$$\begin{aligned} threshold_i &= \frac{R_{max} + m}{n} - R_{max} \times K \\ &= R_{max} \times \left(\frac{1}{n} - K\right) + \frac{m}{n} \end{aligned} \tag{2}$$

where, m is the sum of neighbor nodes' rank expected for R_{max}, and n is the number of neighbor nodes. If $K < 1/n$, $threshold_i > m/n$, i.e., a legitimate node whose rank is lower than m/n is not selected as a parent. If $K \geq 1/n$, $threshold_i \leq m/n$, i.e., a legitimate node whose rank is lower

TABLE I. SIMULATION MODEL

Name	Value
Simulator	Cooja
Simulation area	200m × 200m
Number of all nodes	32
Number of DODAG root	1
Number of attackers	1-3
Transmission range	50 m
MAC protocol	IEEE 802.15.4
Objective function	OF0

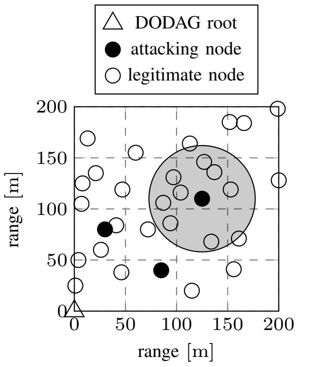


Fig. 3. Simulation topology. A circle filled with transparent gray indicates the transmission range.

than m/n is likely to be selected as a parent. The proposed scheme is effective if $n \geq 1/K$, i.e., there are many neighbor nodes. Although our scheme is not effective when each node has less parent candidates, it may not be a problem since attacking nodes cannot collect a lot of packets as well.

D. Merits and Demerits

The proposed scheme has a merit that attacking nodes cannot collect more packets if they falsely broadcast lower rank. The attacking nodes we assume have no merits to claim lower rank than true ones so as to collect more packets. As a result, attacking nodes may consider to increase $threshold_i$ by broadcasting higher rank than legitimate nodes' ranks since legitimate nodes' ranks are lower than $threshold_i$ and thus attacking nodes may collect more packets. If an attacking node broadcasts higher rank, $threshold_i \leq m/n$ in the case of $K \geq 1/n$ in Eq. (2). Therefore, an attacking node cannot collect more packets by broadcasting higher rank. The proposed scheme has a demerit that the number of hops from each node to the DODAG root may be increased since each node may avoid selecting a legitimate node as its parent.

V. EVALUATION

A. Simulation Model

We evaluate the number of child nodes that select an attacking node as their parents by Contiki's network simulator Cooja [15][16]. We compare the proposed scheme with the conventional RPL scheme since, as described in Section III in [8][9], each node selects its parent node by the same method as the conventional RPL scheme.

We simulate three scenarios. Scenario 1 assumes that attacking nodes pretend to be the DODAG root, i.e., they broadcast the same ranks as that of DODAG root. Scenario 2 assumes sophisticated attacking nodes that pretend to be the legitimate node, i.e., they broadcast ranks as (true rank $-\Delta R$) not to be detected as attacking nodes easily, where ΔR is a minimum hop-by-hop increasing rank value in [2]. Scenario 3 assumes that attacking nodes behave the same as the legitimate node, i.e., without faking ranks. We show the simulation topology for the evaluation in Fig. 3, and the simulation model in TABLE I. The DODAG root is fixed at the origin (0, 0), legitimate and attacking nodes are randomly deployed, so that we can simulate different topologies.

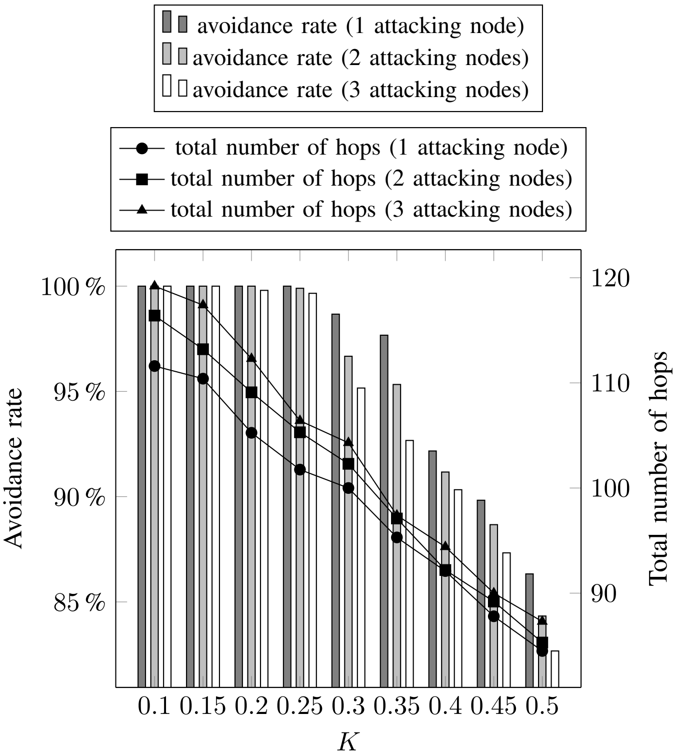


Fig. 4. Avoidance rate and total number of hops in scenario 1 in parameter tuning.

B. Parameter Tuning

Before evaluation, we tune a parameter K by measuring the avoidance rate and the total number of hops. Here, the avoidance rate denotes the ratio of the number of nodes that avoid selecting an attacking node as its parent node to the number of legitimate nodes. In order to minimize the attacking nodes' influence, both high avoidance rate and low total number of hops are required. Fig. 4 shows the avoidance rate and the total number of hop versus K in the scenario 1. From Fig. 4, both high avoidance rate and low total number of hops are achieved with $K = 0.25$. Although $K < 0.25$ brings about high avoidance rate, its total number of hops is high since each node excludes not only attacking nodes but also legitimate nodes that have lower rank for their parent candidates. Therefore, We use $K = 0.25$ in the following simulation. Since we use $K = 0.25$, from Eq. (2), $n \geq 1/0.25 = 4$. This indicates the proposed scheme is effective if the number of neighbor nodes is not less than four.

C. Simulation Results and Discussion

Fig. 5 shows the total number of child nodes attached to attacking nodes, which is denoted as $N_{attacked}$, versus the number of attacking nodes. Fig. 6 shows $N_{attacked}$ versus the number of hops from the DODAG root to an attacking node. From Fig. 5 and 6, we can see that as $N_{attacked}$ gets larger, attacking nodes attach to more child nodes. As shown in Fig. 5 and Fig. 6, the proposed scheme achieves the lower $N_{attacked}$ than RPL in both scenario 1 and scenario 2. Fig. 5 and 6 indicate that, in RPL, $N_{attacked}$ is larger than those in our scheme since each node selects a node that has the lowest rank as its parent. On the other hand, in our scheme, $N_{attacked}$

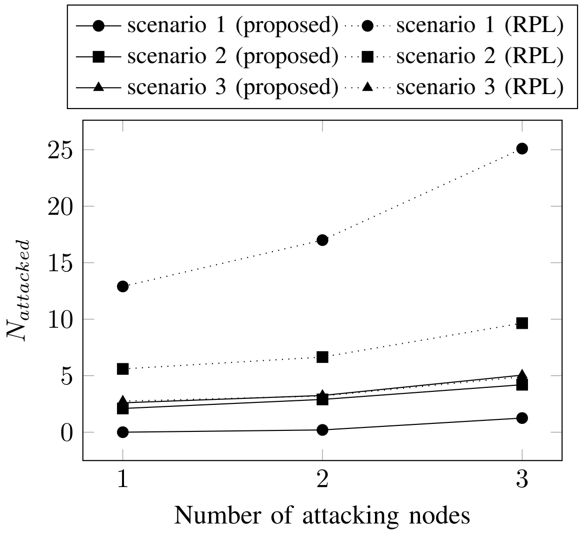


Fig. 5. $N_{attacked}$ versus the number of attacking nodes.

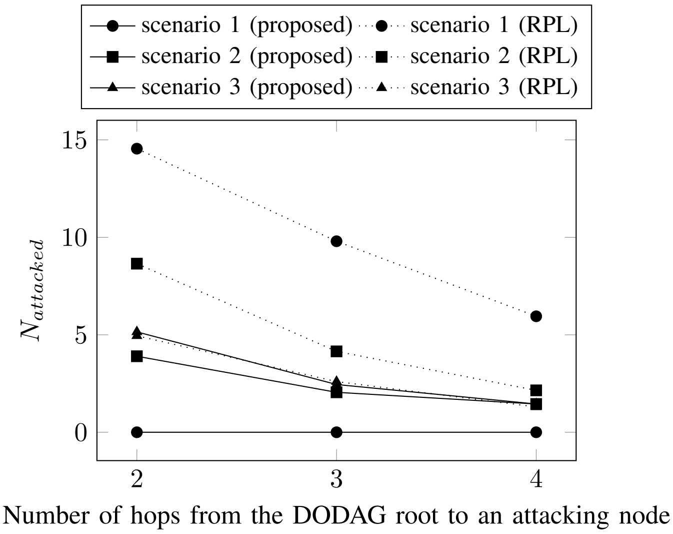


Fig. 6. $N_{attacked}$ versus the number of hops from DODAG root to an attacking node.

gets less since each node excludes attacking nodes from its parent candidates. We then compare the effect of our scheme against various scenarios. As we can see from Fig. 5 and Fig. 6, $N_{attacked}$ gets less in scenario 3 than in scenario 1 and scenario 2. In scenario 1 and scenario 2, ranks of attacking node are lower than those of legitimate ones. On the other hand, in scenario 3, a rank of attacking nodes is same as that of legitimate ones. As we can see from Fig. 5 and Fig. 6, the scenario 3, i.e., when an attacking node claims its true rank, is the best strategy for attacking nodes to be chosen as parent nodes by legitimate ones regardless of the number of attacking nodes and the number of hops from the DODAG root to an attacking node. From these results, the attacking nodes we assume have no merits to claim falsely lower ranks than true ones when our parent selection scheme is applied.

VI. Conclusion

In this paper, we have proposed that secure parent selection scheme so that each child node can select a legitimate node as its parent. In the proposed scheme, each node judges whether its neighbor node is an attacking node or not by utilizing the threshold, and it selects its parent node except for nodes whose rank is judged to be too low. We show that the proposed scheme reduces the total number of child nodes attached to attacking nodes in comparison with the conventional RPL scheme and the attacking node we assume have no merits to claim falsely lower ranks than true ones by computer simulation.

Acknowledgment

This work is partly supported by the Grant in Aid for Scientific Research (No.26420369) from Ministry of Education, Sport, Science and Technology, Japan.

References

[1] K. Ashton, “That ‘ Internet of Things ’ Thing,” RFID Journal, 2009. [Online]. Available: http://www.rfidjournal.com/articles/view?4986

[2] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” RFC 6550, 2012. [Online]. Available: http://tools.ietf.org/html/rfc6550

[3] J. Vasseur and A. Dunkels, *Interconnecting Smart Objects with IP: The Next Internet*. Morgan Kaufmann, 2010.

[4] K. Seo and S. Kent, “Security Architecture for the Internet Protocol,” RFC 4301, 2005. [Online]. Available: http://tools.ietf.org/html/rfc4301

[5] A. Le, J. Loo, A. Lasebae, M. Aiash, and Y. Luo, “6LoWPAN: a study on QoS security threats and countermeasures using intrusion detection system approach,” *International Journal of Communication Systems*, vol. 25, no. 9, pp. 1189–1212, 2012.

[6] A. Le, J. Loo, A. Lasebae, A. Vinel, Y. Chen, and M. Chai, “The Impact of Rank Attack on Network Topology of Routing Protocol for Low-Power and Lossy Networks,” *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3685–3692, 2013.

[7] A. Dvir, T. Holczer, and L. Buttyan, “VeRA-Version Number and Rank Authentication in RPL,” in *Proc. IEEE Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2011, pp. 709–714.

[8] S. Raza, L. Wallgren, and T. Voigt, “SVELTE: Real-time intrusion detection in the Internet of Things,” *Ad hoc networks*, vol. 11, no. 8, pp. 2661–2674, 2013.

[9] H. Perrey, M. Landsmann, O. Ugus, T. C. Schmidt, and M. Wählisch, “TRAIL: Topology Authentication in RPL,” arXiv:1312.0984, 2013. [Online]. Available: http://arxiv.org/abs/1312.0984

[10] P. Thubert, “Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL),” RFC 6552, 2012. [Online]. Available: http://tools.ietf.org/html/rfc6552

[11] C. Karlof and D. Wagner, “Secure routing in wireless sensor networks: attacks and countermeasures,” *Ad hoc networks*, vol. 1, no. 2, pp. 293–315, 2003.

[12] M. Landsmann, M. Wahlisch, and T. Schmidt, “Topology Authentication in RPL,” in *Proc. IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, April 2013, pp. 73–74.

[13] T. Matsunaga, K. Toyoda, and I. Sasase, “Low False Alarm Rate RPL Network Monitoring System by Considering Timing Inconstancy between the Rank Measurements,” in *Proc. International Symposium on Wireless Communications Systems (ISWCS)*, 2014.

[14] ——, “Low false alarm attackers detection in rpl by considering timing inconstancy between the rank measurements,” *IEICE Communications Express*, vol. 4, no. 2, pp. 44–49, 2015.

[15] A. Dunkels, B. Gronvall, and T. Voigt, “Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors,” in *Proc. IEEE Conference on Local Computer Networks (LCN)*, 2004, pp. 455–462.

[16] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, “Cross-Level Sensor Network Simulation with COOJA,” in *Proc. IEEE Conference on Local Computer Networks (LCN)*, 2006, pp. 641–648.