A Gini Index-Based Countermeasure Against Sybil Attack in the Internet of Things

Bryan Groves Cong Pu

***Abstract*—As an essential ingredient of Internet of Things (IoT), IPv6-based Low Power and Lossy Networks (LLNs) largely consisting of various resource-constrained devices are expeditiously proliferating and playing an important role in the realization of ubiquitous computing and communication infrastructure. In order to provide efficient and reliable communication between resource-constrained devices and connect them to the Internet, a novel routing protocol for LLNs, a.k.a. RPL, has been proposed. However, due to wide distribution, openness, and instinctive resource constraints of IoT devices, IoT and its applications become an ideal target for cyber attacks. Thus, investigating potential attacks against IoT-related routing protocol is a top priority to improve the security of the future IoT systems. In this paper, we propose a Gini index-based countermeasure to effectively detect and mitigate sybil attack in RPL-based LLNs, where the malicious node multicasts an excessive number of DODAG Information Solicitation (DIS) messages with different fictitious identities to cause the legitimate nodes to restart the Trickle algorithm frequently and broadcast a large number of DODAG Information Object (DIO) messages to quickly drain the limited energy resource of legitimate nodes. We conduct extensive simulation experiments for performance evaluation and comparison using OMNeT++, and the simulation results show that the proposed countermeasure can accurately detect and effectively mitigate sybil attack, indicating a viable approach against cyber attack in the Internet of Things.**

***Index Terms*—Sybil Attack, Gini Index-Based Detection, RPL, Low Power and Lossy Networks, Internet of Things**

I. Introduction

The vision of Internet of Things (IoT) foresees a future communication paradigm in which information systems will be seamlessly integrated with heterogeneous smart sensors, objects and devices that are capable of communicating with each other without human intervention [1]. IoT applications are expected to penetrate a variety of civilian and military application areas, such as smart grid, smart transportation, and smart cities. For example, in a smart city environment, many radio frequency (RF) transmitters and receivers are deployed, and the goal of the military is to explore the potential to exploit civilian IoT software defined radio infrastructure as radio frequency sensors to attain information dominance [2]. In this context, IPv6-based Low Power and Lossy Networks (LLNs) comprised of a myriad of multi-sized and various resource-constrained devices endowed with the capabilities of sensing, computing, and wireless communicating represent a key enabler for IoT deployments. However, wide distribution, openness, and limited resource make IoT devices and LLNs look especially attractive to attackers and become an ideal target for cyber attacks. Therefore, security and specifically the ability to detect compromised devices, together with securing the routing functionalities and collecting and preserving evidence of an attack or malicious activities emerge as a priority in successful deployment of IoT applications.

As the demand of providing Internet connectivity to resource-constrained devices and efficiently constructing reliable routes over lossy wireless links increase, a novel routing protocol for LLNs, also referred to as *RPL* [3], has been proposed by Internet Engineering Task Force Working Group. By leveraging IEEE 802.15.4 at the PHY and MAC layer to form LLNs, Cisco has built Field Area Networks (FANs) for smart grids based on the IPv6 architecture and employed RPL to provide end-to-end two-way communication to each smart metering endpoint [4]. However, RPL was not originally designed with the consideration of the security requirements for cyber attacks, and security mechanisms are also optional to implement because they greatly affect the performance of resource-constrained devices [5]. Thus, RPL-based LLNs are vulnerable to various Denial-of-Service (DoS) attacks that primarily target service availability [6].

In this paper, we present and investigate a potential DoS attack, which is *sybil attack*, in RPL-based LLNs. In sybil attack, the malicious node multicasts an excessive number of DODAG Information Solicitation (DIS) messages with different fictitious identities to cause the legitimate nodes to restart the Trickle algorithm frequently and broadcast a large number of DODAG Information Object (DIO) messages. In RPL, DIS and DIO are control messages necessary to build the routing topology. As a result, immoderate receiving and broadcasting control messages drain the limited energy resource of legitimate nodes, and finally cause the legitimate nodes to be unable to communicate and suffer from denial of service. The sybil attack primarily targets the vulnerability of DIO transmission mechanism in RPL by violating an implicit assumption, i.e., all legitimate nodes unhesitatingly and faithfully broadcast a DIO message when they receive a DIS message without a Solicited Information option, or with a Solicited Information option and all matched predicates in the Solicited Information option. In light of these, we propose a Gini index-based countermeasure to effectively detect and mitigate sybil attack. Our contribution is summarized below:

- We analyze the vulnerabilities of RPL routing protocol and present a potential and severe DoS attack, which is

Bryan Groves (Email: bryangroves95@gmail.com) is with the Joint Force Headquarters - Department of Defense Information Network. Cong Pu (Email: puc@marshall.edu) is with the Weisberg Division of Computer Science, Marshall University, Huntington, WV 25755.

978-1-7281-4280-7/19/$31.00 ©2019 IEEE

sybil attack. This is the first in-depth work to investigate the performance impact of sybil attack in RPL-based LLNs.

- We propose a Gini index-based countermeasure and its corresponding techniques, also referred to as *GINI*, against sybil attack. The basic idea is that each node measures the dispersity of the identities in the received DIS messages to detect whether there is a sybil attack. If so, the *GINI* will trigger the attack mitigation process to eliminate sybil attack.

We develop a discrete event-driven simulation framework by using OMNeT++ [7] and evaluate its performance through extensive simulation experiments in terms of four performance metrics. The simulation results indicate that the proposed countermeasure can not only accurately detect sybil attack, but also significantly improve the performance in terms of detection rate, energy consumption, and detection latency.

The rest of the paper is organized as follows. An overview of existing and relevant literature is provided in Section II. The basic RPL operations and its potential vulnerabilities are presented and analyzed in Section III, respectively. Section IV focuses on the adversarial model and the proposed countermeasure. Extensive simulation experiments are presented and analyzed in Section V. Finally, concluding remarks with future research direction are provided in Section VI.

II. Related Work

In this section, we analyze a variety of existing attacks and countermeasures in LLNs and similar environments.

A significant amount of research work has mainly focused on developing countermeasures to defend against different attacks in various environments, such as camouflage-based detection against selective forwarding attack in Energy Harvesting Motivated Networks (EHNets) [8], [9], acknowledgment-based approach against stealthy collision attack in EHNets [10], single checkpoint-assisted approach against selective forwarding attack in Wireless Sensor Networks [11], DSR-based bait detection scheme against routing misbehaviors in Mobile Ad Hoc Networks [12], jamming-resilient multipath routing protocol against jamming attack in Flying Ad Hoc Networks [13], and countermeasure against interest flooding attack in Named Data Networking [14], [15].

As the emergence of IoT and rapidly proliferating IoT applications, investigating potential attacks in RPL-based LLNs has been a top priority over the past few years. The [16] investigates the flooding attack in RPL networks, and observes that this attack can significantly decrease the packet delivery rate while increase the end-to-end packet delay. A light-weight anti-jamming technique, named Dodge-Jam, is proposed to address the stealthy jamming attacks with small overhead in LLN environments [17]. The Dodge-Jam is composed of ACK channel hopping, Multi-ACK channel hopping, and Multi-block data shift techniques to evade stealthy jamming attackers and recover packets from jammed transmissions. The [18] studies a suppression attack that targets on the vulnerabilities of Trickle algorithm and does not require to steal cryptographic keys from some legitimate nodes to attack the network. The proposed suppression attack induces victim nodes to suppress the transmission of DIO messages, which are the RPL messages used to construct the routing topology. This causes a general degradation of the routes' quality that can result in network partitions eventually. In [19], a sink-based intrusion detection system for the detection of rank attack in RPL is presented, where all detection processes take place at the sink node. The [20] proposes a lightweight identity based offline–online signature based scheme to defend against version number attack and rank spoofing attack. A novel intrusion detection system (SVELTE) is proposed to secure IPv6 over Low-power Wireless Personal Area Network (6LoWPAN) running with RPL from network layer and routing attacks. In the SVELTE, 6LoWPAN Mapper gathers information about the RPL network and reconstructs the network in the border router, and the intrusion detection module integrated with mini-firewall analyzes the traffic and detects intrusion [21]. The [22] presents a DAO insider attack in RPL's Internet of Things networks, where a malicious node sends fake DAO control messages to its parent nodes periodically to trigger parent nodes to forward the fake messages upward to the root node. Extensive simulation results show that this attack can have a detrimental side effect on the performance of RPL's Internet of Things networks, such as increasing power consumption and latency, and reducing reliability.

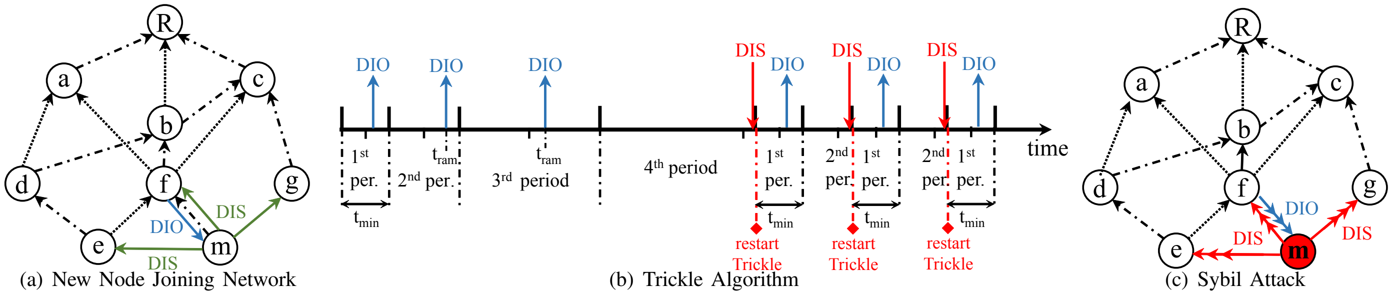
In summary, various attacks and their countermeasures have been well studied in various networks and similar environments. However, little attention has been paid to sybil attack and corresponding countermeasure in RPL-based LLNs.

III. RPL Routing Protocol and Sybil Attack

A. Overview of RPL Routing Protocol

In order to provide a specific routing solution for Low Power and Lossy Networks, where a set of resource-constrained devices (later nodes) communicate directly or indirectly through lossy links with high packet error rate and link outages, a novel distance vector and source routing protocol, named *RPL*, is proposed in [3]. RPL organizes nodes in one or more Destination-Oriented Directed Acyclic Graphs (DODAGs) to maintain the network state information. Each DODAG consists of a number of normal nodes and one DODAG root, and is distinguished by RPL Instance ID, DODAG ID, and DODAG Version Number. In DODAG, normal nodes collaboratively collect and forward information to the DODAG root, while the DODAG root is responsible for connecting to the Internet.

After the deployment of LLNs, the DODAG root will first issue a DAG Information Object (DIO) control message to construct a DODAG and build upward routes directed from other nodes to the DODAG root. The DIO control message includes the DODAG root's ID, the rank of the DODAG root, and an Objective Function which describes the routing metrics and constraints. When a node receives the DIO message and is willing to join the DODAG, it adds the sender of DIO message to its parent list, computes its own rank according to the piggybacked Objective Function, and passes on the DIO



(a) New Node Joining Network (b) Trickle Algorithm (c) Sybil Attack

Fig. 1. (a) A new node n_m multicasts the DIS message to probe for the DIO message from adjacent nodes to join the network; (b) Example of Trickle algorithm, where upward arrows represent emitted DIO messages and downward arrows represent received DIS messages; (c) A malicious node n_m multicasts multiple DIS messages to perform sybil attack, where overlapped red and blue arrows represent multiple DIS messages and DIO messages, respectively.

message with the updated rank information. In RPL, the rank is used to imply the node's position relative to other nodes with respect to a DODAG root. When the DIO message reaches the border node, the upward route is built and each node can send/forward the information to the DODAG root through its parent list. In order to build end-to-end communication (downward routes) from the DODAG root to other nodes, the border node needs to issue a Destination Advertisement Object (DAO) control message to propagate reverse route information and record the nodes visited along the upward routes. If a new node wants to join the existing DODAG, it can request topology information from the adjacent nodes in the existing DODAGs by multicasting a DAG Information Solicitation (DIS) control message without a Solicited Information option, or with a Solicited Information option and all predicates matched in the Solicited Information option.

In order to share the topology and routing information among nodes in DODAG, each node periodically broadcasts the DIO message according to the Trickle algorithm [23]. The Trickle algorithm is a density-aware local communication primitive with an underlying consistency model to dynamically guide and adjust the emission of DIO messages. In more details, if a node receives the DIO message piggybacked with the routing information that is consistent with its currently stored information, it reduces the emission rate of DIO message. When the DIO message piggybacked with inconsistent routing information is received, the node increases its emission rate of DIO message. *Additionally, a DIS message which is used by a new node who wants to join the existing DODAG is also considered as inconsistent routing information, since the network topology will be changed after the new node successfully joins the DODAG.*

For example, as shown in Subfig. 1(a), suppose that a new node n_m wants to join the existing DODAG, and then broadcasts a DIS message. When an adjacent node, e.g., n_f, receives the DIS message from n_m, it terminates the scheduled emission of DIO message, and restarts the Trickle algorithm from a period of a minimum length t_{min}. Here, the Trickle algorithm is shown in Subfig. 1(b), where the time is divided into periods of variable length. Usually, the node schedules the emission of DIO message at a random time t_{ram} in the second half of each period, and then listens to wireless channel for inconsistent routing information (e.g., DIS message from new node). If the node does not receive a DIS message or inconsistent routing information until t_{ram}, it broadcasts the scheduled DIO message, and then doubles the length of the next period. This will continue until the length of the time period reaches a previously defined maximum length. Otherwise, the transmission of the scheduled DIO message is terminated and the period starts over from a minimum length t_{min}. As shown in Subfig. 1(b), n_f receives the DIS message within the 4^{th} period, the Trickle algorithm starts again from t_{min}.

B. Sybil Attack

In RPL, a new node utilizes the Solicited Information option and DIS messages to request DIO messages from neighboring nodes to join the existing DODAG. In addition, a new node can specify a number of predicate criteria in the Solicited Information option to be matched by a receiving node to achieve the goal of limiting the number of DIO replies. However, the DIS transmission mechanism can be exploited by an adversary to attack the network as well. If the malicious node generates and multicasts a large number of DIS messages piggybacked with different fictitious identities, all receiving nodes will believe that new nodes want to join the network, and then restart the Trickle algorithm from the beginning repeatedly and broadcast an excessive number of DIO messages. For example, in Subfig. 1(c), suppose that n_m is a malicious node and multicasts multiple DIS messages piggybacked with randomly generated different fictitious identities to all its neighbor nodes n_e, n_f, and n_g. When the neighbor node, e.g., n_f, receives multiple DIS messages with different identities, it believes that multiple new nodes wish to join the network and request for current network information. According to RPL, n_f restarts the Trickle algorithm from t_{min} repeatedly, and then broadcasts multiple DIO messages piggybacked with current network information at a random time in the second half of t_{min}. As a result, n_f receives an excessive number of DIS messages and broadcasts a large number of DIO messages, which significantly exhausts energy resource and communication bandwidth, and finally causes n_f to run out of its energy and suffer from denial of service.

IV. The Proposed Gini Index-Based Approach

In this section, we first present the adversarial model and then propose a Gini index-based countermeasure, also referred to as *GINI*, to detect and mitigate sybil attack.

A. Adversary Model

We consider an LLN running with RPL, where a set of resource-constrained nodes and one DODAG root communicate directly or indirectly through lossy links. Each node is uniquely identified by a node ID, e.g., a media access control (MAC) address (48 bits). For the simplicity, we assume that the RPL only maintains one DODAG structure rooted at the DODAG root in this paper. An adversary is able to capture and compromise a legitimate node, gain access to all stored information including public and private keys, and reprogram it to behave maliciously. In addition, the malicious node may create the fictitious identities derived either from its own MAC address or a randomly generated fake MAC address. Due to the constant size of MAC address (e.g., 48 bits), it is not guaranteed that every randomly generated fictitious identity is different from all real MAC addresses used in the network. However, the probability of generating a fake MAC address which is same as the existing address in the network will be extremely low and close to zero, because the 24-bit address space contains 2^{24} possible MAC addresses[1][24]. Thus, we implicitly assume that the randomly generated fictitious identity does not exist in the network and will be considered as new identity by legitimate nodes.

B. Gini Index-Based Countermeasure

The basic idea of the proposed *GINI* countermeasure is to use the statistical properties of identities to detect and mitigate sybil attack. To be specific, the *GINI* measures the dispersity of the identities in the received DIS messages to detect whether there is a sybil attack based on the Gini index-based theory [25]. If so, the *GINI* triggers the attack mitigation process to eliminate sybil attack.

First, each node records a trace of the received DIS messages from newly joined nodes during each observation window period ω and maintains a new node trace table (*TT*) to monitor any potential DIS message forwarding misbehavior of its neighbor nodes. Due to the limited storage space, the traces recorded in the previous observation window period, where the traces timestamped less than t_{cur} - ω, will be evicted from the table. Here, t_{cur} is the current system time, and ω is a system parameter and its impact on the performance is observed in Section V. An entry of the TT consists of node identity in the received DIS message (n_{mac}) and timestamp (ts). For example, in Subfig. 1(a), a new node n_m wants to join the existing network, so it multicasts a DIS message to probe for the DIO messages from adjacent nodes. When an adjacent node, e.g., n_f, receives the DIS message, it first records the received DIS message and adds an entry into the TT, TT_f = TT_f ∪ [n_m, t_{cur}]. And then, it terminates the scheduled transmission of DIO message, restarts the Trickle algorithm from a period of a minimum length t_{min}, and then broadcasts the DIO message piggybacked with current routing information.

Second, when an observation window ω ends, each node measures the dispersity of new nodes' identities based on Gini index theory [25]. Gini index is an impurity-based criterion that measures the divergence between the probability distributions of the target attribute's value. Suppose that a set D contains examples from N classes, and p_i is the relative frequency of examples in class i in D. The Gini impurity $Gini(D)$ is then defined as

$$Gini(D) = 1 - \sum_{i=1}^{N} p_i^2. \tag{1}$$

The Gini impurity reflects the impurity level of a set of information. Specifically, when the examples are equally distributed among all N classes, $Gini(D)$ reaches the maximum value (1 $-$ $\frac{1}{N}$). However, $Gini(D)$ reaches the minimum value zero when all examples belong to one class. In this paper, we use the Gini impurity to measure the dispersity of new nodes' identities in the received DIS messages and detect the potential sybil attack. If there is no sybil attack, the Gini impurity of the identities of newly joined nodes varies in a normal range, since the number of newly joined nodes is small, and the identities have a relative stable distribution. When sybil attack exists in the network and the attacker multicasts an excessive number of DIS messages with different fictitious identities, the Gini impurity of the identities in the received DIS messages will be influenced and exceeds the normal range.

In light of these, the entire identity set or MAC address space (2^{24}) is equally divided into N classes, each node obtains the identities of the received DIS messages from the TT and calculates the probability distribution of identities in the identity class i within the ω period. We define D_i as the set of new nodes' identities in the i^{th} observation window period ω^i. Then, each node computes the Gini impurity of the identities in the received DIS messages within ω^i according to Eq. 1, and compares $Gini(D_i)$ with the previous observation window period's Gini impurity $Gini(D_{i-1})$ according to

$$Atk(D_i) = \begin{cases} 1, & \frac{Gini(D_i) - Gini(D_{i-1})}{Gini(D_{i-1})} > Th_{Gini,i} \\ 0, & \frac{Gini(D_i) - Gini(D_{i-1})}{Gini(D_{i-1})} <= Th_{Gini,i} \end{cases} \tag{2}$$

Here, $Atk(D_i)$ = 1 indicates that there is a potential sybil attack existing in the network. $Th_{Gini,i}$ is a threshold value updated by the low pass filter with a filter gain constant α,

$$Th_{Gini,i} = \alpha \cdot Th_{Gini}^{avg} + (1 - \alpha) \cdot Th_{Gini,i-1}, \tag{3}$$

where Th_{Gini}^{avg} is the average threshold value of Gini impurity over all past observation window periods, and $Th_{Gini,i-1}$ is the threshold value of Gini impurity in the $i-1^{th}$ observation window period.

Third, once sybil attack is detected by the Gini impurity detection mechanism, the attack mitigation procedure will be triggered to mitigate sybil attack by limiting the DIO message replying rate. In order to take into account the actual state of

[1] Traditional MAC addresses are 48 bits. The leftmost 24 bits called a "prefix" is associated with the manufacturer, which is assigned by the IEEE. The rightmost 24 bits of a MAC address represent a manufacturer-assigned identification number for the specific device.

the network and react to varying attack patterns quickly, we propose to utilize an adaptive DIO message replying rate to determine how many DIO messages should be replied within each observation window. The node who detects sybil attack will construct an *Alert* packet and broadcast it to all adjacent nodes to announce the potential sybil attack nearby. When a node receives the *Alert* packet, it will reduce the DIO message replying rate in the next observation window according to the function λ^{dio}, which has the following form,

$$\lambda^{dio} = \delta + \varphi \cdot e^{1-det_{atk} \cdot \gamma}, \tag{4}$$

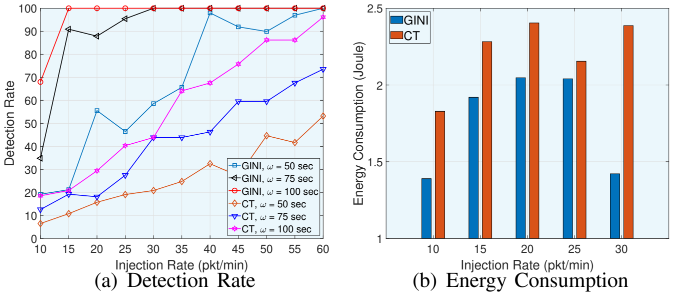
where δ and φ are system parameters and δ is an asymptote to ensure that the λ^{dio} never reach 0. γ has an impact on the change of λ^{dio}, and a larger value for γ leads to a smaller λ^{dio} being reached quicker generally. det_{atk} is the accumulated detection rate of sybil attack, and is represented as

$$det_{atk} = \frac{c_{atk}}{c_{win}}, \tag{5}$$

where c_{atk} is the total number of detected sybil attack according to Eq. 2 and c_{win} is the total number of observation windows. The rationale behind this design is that the DIO message replying rate λ^{dio} can change based on network conditions. If an attacker is aggressive, the DIO message replying rate λ^{dio} drops quickly and increases slowly once the attack stops. If there is no sybil attack, the DIO message replying rate λ^{dio} can be maintained at a high level. For example, in Subfig. 1(c), n_f can decide whether to reply the received DIS message based on λ^{dio} during each observation window. If n_m continues to perform sybil attack, the λ^{dio} will quickly drop, indicating many received DIS messages will be ignored. If the sybil attack disappears for a certain amount of time (e.g., two continuous observation windows), n_f will increase λ^{dio} slowly.

V. Performance Evaluation

We conduct extensive simulation experiments using OMNeT++ [7] to evaluate the performance of proposed scheme. A 100×100 m^2 square network area is considered, where 50 nodes and one DODAG root are uniformly distributed. The radio model simulates CC2420 with a normal data rate of 250 Kbps, and 802.15.4 MAC/PHY operates with a default configuration in the 2.4 GHz band [26]. The communication range of each node is 30 (m). To emulate the scenario that a node runs out of its battery or is damaged, and then leaves the network, 1 to 5 legitimate nodes are randomly generated and join the network. A set of malicious nodes are randomly located in the network, and multicasts malicious DIS messages with a different packet injection rate. And the total simulation time is set to 5000 seconds, and each simulation scenario is repeated 5 times to obtain steady state performance metrics. In this paper, we measure the performance in terms of detection rate, energy consumption, number of detected sybil attack, and detection latency by changing key simulation parameters, including malicious DIS message injection rate (Υ^{dis}) and size of observation window (ω). We compare the performance of the proposed *GINI* approach with the fixed threshold scheme, denoted as *CT*, for performance comparison and evaluation.

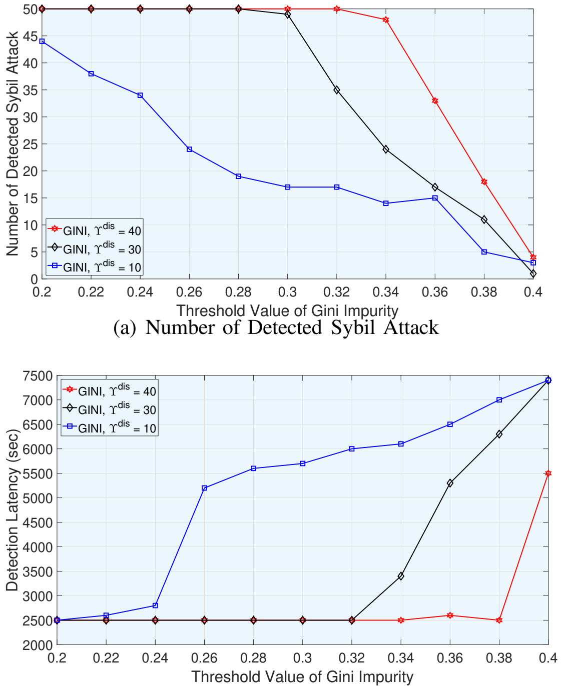


(a) Detection Rate (b) Energy Consumption

Fig. 2. The performance of detection rate and energy consumption against malicious DIS message injection rate.

First, we observe the performance of detection rate and energy consumption against malicious DIS message injection rate in Fig. 2. In Subfig. 2(a), the overall detection rate of GINI and CT increases as the malicious DIS message injection rate Υ^{dis} increases. With a larger Υ^{dis}, more malicious DIS messages will be broadcasted by malicious nodes within each observation window. Since each node can receive a larger number of malicious DIS messages during each observation window, more forwarding misbehaviors can be easily detected, and a larger detection rate is observed. The proposed GINI scheme outperforms the CT because the GINI is very sensitive to the change of the number of received malicious DIS messages. Even though there is a minor increment of the number of received malicious DIS messages in the next observation window, the GINI impurity of two consecutive observation window will have a significant difference. Thus, more forwarding misbehaviors can be detected, and a larger detection rate is observed by the GINI. As the observation windows ω increases, a larger detection rate is observed. This is because each node can receive more malicious DIS messages within a larger observation window, thus, more forwarding misbehaviors can be detected, resulting in a larger detection rate. In Subfig. 2(b), the energy consumption of the GINI is lower than that of the CT. Since the GINI can detect more forwarding misbehaviors, the number of detected forwarding misbehaviors can quickly reach a threshold value. After that, each node limits the DIO message replying rate. As a result, the number of received malicious DIS messages and replied DIO messages are significantly reduced, and a lower energy consumption is observed.

Second, we measure the performance of number of detected sybil attack and detection latency against threshold value of Gini impurity in Fig. 3. In Subfig. 3(a), as the threshold value of Gini impurity increases, the number of detected sybil attack decreases. With a larger threshold value, each node needs to receive more malicious DIS messages in the next observation window to detect the potential sybil attack. If the malicious node broadcasts a lesser number of malicious DIS messages, the differences of Gini impurity in two consecutive observation windows are not always larger than the threshold value of Gini impurity. As a result, the potential sybil attack cannot be detected, and the number of detected sybil attacks decreases.



(a) Number of Detected Sybil Attack

(b) Detection Latency

Fig. 3. The performance of the number of detected sybil attack and detection latency against threshold value of Gini impurity.

With a larger malicious DIS message injection rate, a larger number of detected sybil attacks are observed because the malicious node broadcasts more malicious DIS messages, and the sybil attack can be easily detected by the GINI scheme. As shown in Subfig. 3(b), when the threshold value of Gini impurity increases, the detection latency increases. This is because a larger threshold value of Gini impurity will cause a lesser number of sybil attacks to be detected, more time is required to detect enough number of sybil attacks to trigger the attack mitigation procedure. When the malicious DIS message injection rate increases, the detection latency increases as well. Since a lesser number of malicious DIS messages are broadcasted by malicious nodes with a smaller malicious DIS message injection rate, it takes more time to detect enough forwarding misbehaviors, which results in a larger detection latency.

VI. Conclusion and Future Work

In this paper, we focus on the study of RPL security in the realm of IoT. The basic operations and potential vulnerabilities of RPL are first summarized and analyzed. Then, a Gini index-based countermeasure is proposed to detect and mitigate sybil attack. Extensive simulation results show that the proposed countermeasure can not only accurately detect and efficiently mitigate sybil attack, but also significantly improve the performance in terms of detection rate, energy consumption, and detection latency, indicating a viable approach in RPL-based LLNs. As a future work, since radio propagation and its channel dynamics cannot easily be captured by simulation, we plan to develop a small-scale testbed and deploy a real network composed of TelosB motes in an indoor environment to see the full potential of the proposed countermeasure.

Acknowledgment

This research was supported by 2019 NASA Undergraduate Affiliate Fellowship Program Award.

References

[1] M. Conti *et al.*, "Internet of Things security and forensics: Challenges and opportunities," *Future Gener Comput Syst*, vol. 78, pp. 544–546, 2018.

[2] A. Cohen *et al.*, "Radio Frequency IoT Sensors in Military Operations in a Smart City," in *Proc. IEEE MILCOM*, 2018, pp. 763–767.

[3] T. Winter and P. Thubert, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," *RFC Standard 6550*, March 2012.

[4] Cisco, *Connected Grid Networks for Smart Grid - Field Area Network*, https://www.cisco.com/c/en/us/solutions/industries/energy/external-utilities-smart-grid/field-area-network.html.

[5] H. Kim, J. Ko, D. Culler, and J. Paek, "Challenging the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL): A Survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2502–2525, 2017.

[6] C. Pu and B. Groves, "Energy Depletion Attack in Low Power and Lossy Networks: Analysis and Defenses," in *Proc. ICDIS*, 2019, pp. 14–21.

[7] A. Varga, *OMNeT++*, 2014, http://www.omnetpp.org/.

[8] C. Pu and S. Lim, "Spy vs. Spy: Camouflage-based Active Detection in Energy Harvesting Motivated Networks," in *Proc. IEEE MILCOM*, 2015, pp. 903–908.

[9] C. Pu, S. Lim, B. Jung, and J. Chae, "EYES: Mitigating Forwarding Misbehavior in Energy Harvesting Motivated Networks," *Elsevier Computer Communications*, vol. 124, pp. 17–30, 2018.

[10] C. Pu, S. Lim, J. Byungkwan, and M. Manki, "Mitigating Stealthy Collision Attack in Energy Harvesting Motivated Networks," in *Proc. IEEE MILCOM*, 2017, pp. 575–580.

[11] C. Pu and S. Lim, "A Light-Weight Countermeasure to Forwarding Misbehavior in Wireless Sensor Networks: Design, Analysis, and Evaluation," *IEEE Systems Journal*, vol. 12, no. 1, pp. 834–842, 2018.

[12] C. Pu, S. Lim, C. Jinseok, and J. Byungkwan, "Active Detection in Mitigating Routing Misbehavior for MANETs," *Wireless Network*, vol. 25, no. 4, pp. 1669–1683, 2017.

[13] C. Pu, "Jamming-Resilient Multipath Routing Protocol for Flying Ad Hoc Networks," *IEEE Access*, vol. 6, pp. 68 472–68 486, 2018.

[14] T. Zhi *et al.*, "A Gini Impurity-Based Interest Flooding Attack Defence Mechanism in NDN," *IEEE Commun. Lett.*, vol. 22, no. 3, pp. 538–541, 2018.

[15] C. Pu, P. Nathaniel, and B. Jacqueline, "Self-Adjusting Share-Based Countermeasure to Interest Flooding Attack in Named Data Networking," in *Proc. IEEE CPSCom*, 2019, pp. 142–147.

[16] T. Nguyen *et al.*, "The Flooding Attack in Low Power and Lossy Networks: A Case Study," in *Proc. IEEE SaCoNeT*, 2018, pp. 183–187.

[17] J. Heo *et al.*, "Mitigating Stealthy Jamming Attacks in Low-power and Lossy Wireless Networks," *Journal of Communications and Networks*, vol. 20, no. 2, pp. 219–230, 2018.

[18] P. Perazzo, C. Vallati, G. Anastasi, and G. Dini, "DIO Suppression Attack Against Routing in the Internet of Things," *IEEE Commun. Lett.*, vol. 21, no. 11, pp. 2524–2527, 2017.

[19] U. Shafique *et al.*, "Detection of Rank Attack in Routing Protocol for Low Power and Lossy Networks," *Annals of Telecommunications*, vol. 73, no. 7–8, pp. 429–438, 2018.

[20] M. Nikravan *et al.*, "A Lightweight Defense Approach to Mitigate Version Number and Rank Attacks in Low-Power and Lossy Networks," *Wireless Personal Communications*, vol. 99, no. 2, pp. 1035–1059, 2018.

[21] 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.

[22] B. Ghaleb *et al.*, "Addressing the DAO Insider Attack in RPL's Internet of Things Networks," *IEEE Commun. Lett.*, vol. 23, no. 1, pp. 68–71, 2019.

[23] P. Levis and T. Clausen, "The Trickle Algorithm," *RFC Standard 6206*, March 2011.

[24] *Standard Group MAC Address*, https://standards.ieee.org/products-services/regauth/grpmac/index.html.

[25] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers - A survey," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 35, no. 4, pp. 476–487, 2005.

[26] A. Boulis, *Castalia*, 2014, http://castalia.forge.nicta.com.au.