This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2020.2971463, IEEE Internet of Things Journal

Sybil Attack in RPL-Based Internet of Things: Analysis and Defenses

Cong Pu, *Member, IEEE*

Abstract*—Over the past few years, Internet of Things (IoT) has emerged as a promising paradigm that connects various physical devices to the Internet, and contributes to the development of countless next-generation applications. As a major enabler for IoT, IPv6-based Low Power and Lossy Networks (LLNs) have been receiving considerable attention as a mature solution for scalable data collection in an ubiquitous computing and communication infrastructure. In order to provide efficient point-to-multipoint and multipoint-to-point communication, a novel routing protocol for LLNs, also well-known as RPL, has been proposed and standardized. Nonetheless, due to devices' constraints on processing power, memory, and energy, and the lack of specific security models of RPL routing protocol, LLNs become an ideal target for various security attacks. In this paper, we propose a Gini index-based countermeasure, also called *GINI*, 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 also present a simple analytical model and its numerical results in terms of detection rate. We evaluate the proposed *GINI* countermeasure through extensive simulation experiments using OMNeT++ and compare its performance with two existing schemes, SecRPL and two-step detection. The simulation results show that the proposed *GINI* countermeasure can not only improve detection rate and detection latency, but also reduce energy consumption, indicating a viable approach against sybil attack in the Internet of Things. For continuous improvement and future research, we further discuss the proposed *GINI* countermeasure in terms of design features, design constraint, and possible extensions.

***Index Terms*—Sybil Attack, Denial-of-Service, Gini Index, 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 and objects that are capable of communicating with each other without human intervention [1]. These smart and connected devices generate data that will be utilized by IoT applications to aggregate, analyze, and deliver insight, which helps drive more informed decisions and actions. IoT applications are expected to penetrate a variety of civilian and military application areas, such as smart grid, smart transportation, smart cities, smart home, etc. 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]. According to Cisco, the IoT market has grown for some time and is due to reach 31 billion connected devices by 2020 and 75 billion devices by 2025 [3]. The McKinsey Global Institute forecasts the economic impact of IoT reaching as high as $11 trillion by 2025, and a massive chunk of that is predicted to live in business and industrial applications [4]. By leveraging edge and fog computing, a variety of communication solutions, and various network architectures, we envision that wirelessly connected smart devices in the realm of IoT will enhance information accessibility and availability as well as improve our life further.

In the context of promptly emerging IoT applications, 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. For example, in order to support a number of advanced smart-grid applications and its benefit as a true multi-service platform, Cisco proposes an open-standards-based IPv6 architecture for smart-grid field area network deployments that make use of or are built on IPv6 technology and LLNs [5]. However, wide distribution, openness, and limited resource make smart meters, distribution automation devices, and smart-grid field area networks look especially attractive to attackers and become an ideal target for cyber attacks [6]. 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* [7], has been proposed by Internet Engineering Task Force (IETF) Working Group. One of the design features of RPL is that the protocol is flexible and dynamic so that it can operate in harsh environments with low-speed links potentially experiencing high error rates, while generating very low control plane traffic. In addition, RPL offers other attractive features such as Trickle algorithm limiting the chattiness of control plane, various routing metrics for dynamic link and node, and multi-topology routing. However, RPL was not originally designed with the

C. Pu is with the Weisberg Division of Computer Science, Marshall University, Huntington, WV 25755, USA (e-mail: puc@marshall.edu).

consideration of security requirements for cyber attacks, and security mechanisms are also optional to implement because they greatly affect the performance of resource-constrained devices [8]. Thus, RPL-based LLNs are vulnerable to various Denial-of-Service (DoS) attacks that primarily target service availability by disrupting network routing protocols [9].

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 present an overview of RPL routing protocol, analyze the vulnerabilities of DODAG Information Object (DIO) transmission mechanism and Trickle algorithm, and identify a potential and severe DoS attack, which is sybil attack. To show the serious performance impact of sybil attack in RPL-based LLNs, preliminary simulation results in terms of energy consumption and changes of DIO timeout period are also provided.
- We propose a novel countermeasure and its corresponding techniques based on the Gini index theory, also referred to as GINI, against sybil attack. The basic idea of the GINI is to measure the dispersity of the identities in the received DIS messages to detect sybil attack. We also analyze the Gini index theory with its numerical results to show how the Gini index theory can be applied to the detection of sybil attack.
- We propose a simple analytical model of the GINI and show its numerical results in terms of detection rate. We also revisit existing SecRPL scheme [10] and two-step detection approach [11], and modify them to work in the framework for performance comparison. In addition, we discuss the proposed GINI countermeasure in terms of design features, design constraint, and possible extensions for future research.

We develop a discrete event-driven simulation framework by using OMNeT++ [12] and evaluate its performance through extensive simulation experiments in terms of detection rate, detection latency, and energy consumption. The simulation results indicate that the proposed countermeasure can not only accurately detect and mitigate sybil attack, but also significantly improve the performance in terms of detection rate, detection latency, and energy consumption.

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. An analytical model and its numerical results are presented in Section V. Extensive simulation experiments are provided and analyzed in Section VI. We further discuss the proposed countermeasure in terms of design features, design constraint, possible improvement, and the immunity to other attacks in Section VII. Finally, concluding remarks with future research direction are provided in Section VIII.

II. Related Work

In this section, we analyze a variety of existing security 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) [13], [14], acknowledgment-based approach against stealthy collision attack in EHNets [15], single checkpoint-assisted approach against selective forwarding attack in Wireless Sensor Networks (WSNs) [16], DSR-based bait detection scheme against routing misbehaviors in Mobile Ad Hoc Networks [17], jamming-resilient multipath routing protocol against jamming attack in Flying Ad Hoc Networks [18], countermeasure against interest flooding attack in Named Data Networking [19], [20], RPL-based secure routing protocol against rank manipulation attack in Internet of Things [21], optimization model with optimal resources usage and security guarantee in WSNs [22], optimization framework with security and quality of services (QoS) guarantee in WSNs [23], and multi-channel operation based approach against beacon jamming attack in Vehicular Ad Hoc Networks [24].

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 [10] 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. The authors in [25] model sybil attack using artificial bee colony algorithm and then propose an intrusion detection algorithm against sybil attack in the Internet of Things. Since sybil attack is a population-based attack and the foraging behavior of fictitious identities is similar to the foraging behavior of honey bees, sybil attack is modeled into five phases, namely initialization phase, fitness factor computation, compromising or fabricating phase, contagious phase, and hive selection and launching phase. To detect sybil attack, three new variables, namely nonce ID, control message counter, and timestamps, are added in the DODAG Information Object (DIO) control message. The nonce ID is allocated to each node when it joins the network, and also broadcasted with unique DODAG ID to the neighbor nodes. If the nonce ID and DODAG ID in the received message do not match with the previous record, there is a potential possibility of sybil attack. Control message counter and timestamps is used to track the number of received control messages and the time of arrival of received control messages to detect the sign of sybil attack, respectively. In [26], a time-based trust-aware RPL routing protocol (SecTrust-RPL) is proposed and implemented to protect Internet of Things against sybil attack as well as rank attack. The basic idea of SecTrust-RPL is that each node computes the trustworthiness of its direct neighbors based on the direct trust value and the recommended trust value. The neighbor nodes with higher trust values are chosen to be involved in routing operations, while nodes with lower trust values are categorized either as malicious nodes who launch destructive and devastating attacks or selfish nodes who seek to preserve their resources.

In [27], a comprehensive characteristic analysis of sybil attack in the Internet of Things is provided. The entire process of sybil attack is classified into three phases as compromise, deployment, and launching phase based on the nature of attack operations. The proposed sybil attack modeling can be used to better understand various states that a sybil attacker undergoes, and the possible action it takes in each state. However, it is quite challenging to adapt it in the design of sybil attack in RPL-based LLNs, where all nodes are constrained in terms of processing power, memory, and energy. The [28] first investigates advanced sybil attack in Vehicular Ad Hoc Networks (VANETs) where sybil attacker conducts power control to deliberately change transmission powers to avoid being detected by RSSI-value based countermeasure. Then, it proposes a power control identification sybil attack detection scheme to find anomalous variations in RSSI time series, which are used to identify sybil attackers via a linear Support Vector Machine classifier. Unlike prior RSSI-based techniques that assume a constant transmission power, the authors consider sybil attackers who perform power control to launch the attack. This is the first study that investigates sybil attack with varying transmission powers in VANETs. However, the proposed detection scheme is designed based on Support Vector Machine classification method to differentiate sybil nodes from normal nodes, which is not suitable for resource-constrained nodes in RPL-based LLNs. The [29] proposes a prediction system that can be leveraged in the manipulation of deep-learning solution model solving the problem of sybil attack in social network, where a data harvesting module, a feature extracting mechanism, and a deep-regression model are functioning in a systematic form to analyze and evaluate user's profiles. The proposed approach can provide an accurate sybil detection decision by analyzing and evaluating user's profiles in social network, but also introduces a long detection latency because of data collection. In [30], a distributed approach using the traffic flow theory is proposed to detect sybil attack in VANETs, where each vehicle will monitor its neighborhood in order to detect an eventual sybil attack by comparing between the real accurate speed of the vehicle and the one estimated using the vehicle-to-vehicle communications with vehicles in the vicinity. However, this approach requires implicit monitoring and a large amount of message exchanges between vehicles, which introduce a significant amount of energy consumption. Compared to the prior approaches, the proposed GINI countermeasure is designed with a lightweight detection technique without introducing extra energy consumption. In addition, the GINI can quickly respond to the change of network traffic to further reduce the impact of sybil attack in the network.

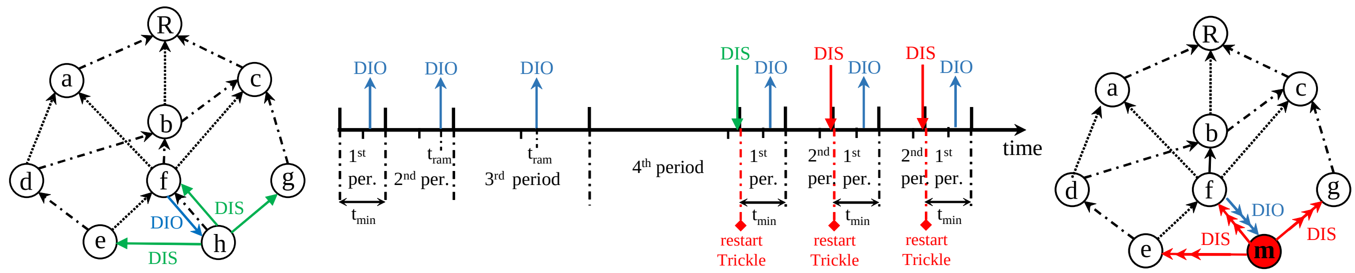
The [31] presents a survey on countermeasures offered to defend MANETs, WSNs, and WMNs from sybil attack. The investigated defense mechanisms are categorized into symmetric cryptography using a central authority, random key pre-distribution, radio resource testing, received signal strength indicator, time difference of arrival, neighborhood data, and energy trust-based system. A survey of sybil attacks and their defense schemes in the Internet of Things are provided in [32], where three types of sybil attacks with different characteristics including social structures and behaviors are identified. In [33], a detailed review of RPL standard along with recently identified security attacks and their mitigation methods are provided. The [34] investigates the important aspects of IoT cybersecurity, such as the state-of-the-art of the current position and potential future directions, the major countermeasures against IoT attacks, and the applications in industries. In addition, a four-layered IoT cybersecurity infrastructure and a taxonomy of attacks on IoT cybersecurity are provided.

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 [7]. 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.



(a) New Node n_h Joining Network (b) Trickle Algorithm at n_f (c) n_m Launching Sybil Attack

Fig. 1. (a) A new node n_h multicasts the DIS message to probe for the DIO message from adjacent nodes to join the network; (b) Example of Trickle algorithm at n_f, 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 launch sybil attack, where overlapped red and blue arrows represents multiple DIS messages and DIO messages, respectively.

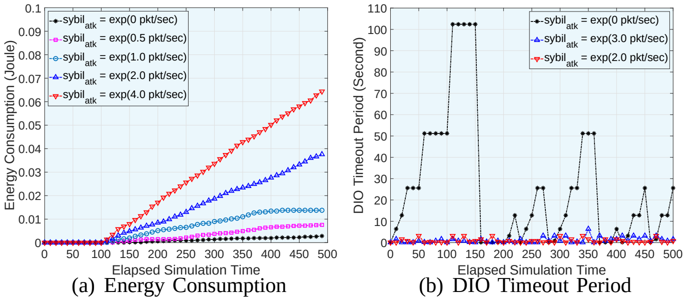
RPL relies on four types of control messages to establish and manage DODAG: DODAG Information Object (DIO), DODAG Destination Advertisement Object (DAO), DODAG Information Solicitation (DIS), and DODAG Destination Advertisement Object Ack (DAO-Ack). After the deployment of LLNs, the DODAG root will first issue a DODAG 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 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 DODAG Destination Advertisement Object (DAO) control message to propagate reverse route information and record the nodes visited along the upward routes. After receiving DAO message, the DODAG root replies a DODAG Destination Advertisement Object Ack (DAO-Ack) message as a unicast packet to the source of DAO message. 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 DODAG 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 DIO messages according to the Trickle algorithm [35]. 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 to achieve the goal of reducing energy consumption through minimizing the redundant DIO messages. In other words, the emission rate of DIO messages is dynamically adjusted according to the stability of routing information. More specifically, 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.

As shown in Subfig. 1(a), suppose that a new node n_h 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_h, 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 current time period. This will continue until the length of the time period reaches a previously defined maximum length t_{max}. 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 restarts 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



(a) Energy Consumption (b) DIO Timeout Period

Fig. 2. The performance of energy consumption and DIO timeout period against elapsed simulation time, where exp() is an exponential function.

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.

In Fig. 2, we measure the total energy consumption of n_e, n_f, and n_g and the DIO timeout period of n_f against simulation time by changing sybil attack rate ($sybil_{atk}$), where the network topology shown in Subfig. 1(c) is considered for the simulation of preliminary results. In this paper, sybil attack rate $sybil_{atk}$ is the frequency that malicious node multicasts a DIS message piggybacked with randomly generated fictitious identity. As shown in Subfig. 2(a), the total energy consumption of n_e, n_f, and n_g without sybil attack ($sybil_{atk}$ = exp(0 pkt/sec)) is significantly lower than the total energy consumption under sybil attack. This is because sybil attack launched by malicious node n_m causes n_e, n_f, and n_g to receive and broadcast more control messages, consuming a significant amount of energy. In addition, the total energy consumption of n_e, n_f, and n_g under sybil attack increases as the sybil attack rate increases from $sybil_{atk}$ = 0.5 pkt/sec to $sybil_{atk}$ = 4.0 pkt/sec. With a larger sybil attack rate, more DIS messages with fictitious identities are generated and multicasted by malicious node n_m. As a result, n_e, n_f, and n_g receive a large number of DIS messages and broadcast the corresponding number of DIO messages, which consumes more energy. In Subfig. 2(b), we observe the change of timeout period of emitting DIO message without and with sybil attack. As shown in Subfig. 2(b), without sybil attack, the timeout period of emitting DIO message is fluctuating between t_{min} = 0.1 second and t_{max} = 110 second because the network experiences regular topology changes. However, with sybil attack rate $sybil_{atk}$ = exp(2.0 pkt/sec) or exp(3.0 pkt/sec), the timeout period of emitting DIO message is always changing around the smallest time period t_{min}. This is because malicious node n_m frequently multicasts DIS messages with fictitious identifies to trigger the Trickle algorithm restart from t_{min}.

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 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][36]. 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 theory [37]. 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

[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.

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 VI. 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_h 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_h, 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.

```
Notations:
 • TT, N, $Gini(D)$, $c_{atk}$, $det_{atk}$, $c_{win}$, $\lambda^{dio}$, $\delta$, $\varphi$, $\gamma$, $\xi$, and $t_{cur}$: Defined before.
 • $mac_i$ and $d^*$: The number of samples in class i and the number of total samples.
 • $\omega_i^k$: The timeout of the $k^{th}$ observation window period at node $n_i$.
 • $pkt[n_{id}, type]$: A packet containing a node ID ($n_{id}$) and packet type (type). Here, type can be DIS, DIO, Alarm, or Isolate.
 • rec($n_i$, $n_j$, type): The node $n_i$ receives the type of packet from node $n_j$.
◇ if rec($n_i$, $n_j$, DIS) = true
    $TT_i$ = $TT_i$ ∪ [$n_j$, $t_{cur}$];
◇ while $t_{cur} < \omega_i^k$
    for i = 1 to N
        $p_i = \frac{mac_i}{d^*}$;
    $Gini(D)$ = 1 - $\sum_{i=1}^{N} p_i^2$;
    if $\frac{Gini(D_i)-Gini(D_{i-1})}{Gini(D_{i-1})} > Th_{Gini,i}$
        $c_{atk}$ += 1;
        Broadcast Alarm packet;
        $det_{atk}$ = $\frac{c_{atk}}{c_{win}}$;
        $\lambda^{dio}$ = $\delta$ + $\varphi$ · $e^{1-det_{atk}\cdot\gamma}$;
    if $c_{atk} > \xi$
        Broadcast Isolate packet;
```

Fig. 3. The pseudocode of the proposed *GINI* algorithm.

Second, when an observation window ω ends, each node measures the dispersity of new nodes' identities based on Gini index theory [37]. 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 samples from N classes, and p_i is the relative frequency of samples in class i in D. The Gini impurity $Gini(D)$ is then defined as

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

The Gini impurity reflects the impurity level of a set of information. Specifically, when the samples 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 samples 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 observation window period ω. 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

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

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,

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

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 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,

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

where δ, φ, and γ are system parameters. Here, δ is an asymptote to ensure that the λ^{dio} never reach 0, so that the δ

number of DIS messages can be replied even when the sybil attacker aggressively performs attack. φ is used to quickly increase the DIO replying rate if there is no sybil attack. 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. In Section VI, δ = 3, φ = 5, and γ = 0.5 are adopted. 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. When the total number of detected sybil attack reaches a certain threshold value ξ, the detecting node will broadcast an *Isolate* packet to its one-hop neighbor nodes to prevent them from receiving any DIS messages from the local area. 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. Major operations of the *GINI* are summarized in Fig. 3.

C. Analysis of The Gini Index Theory

In this subsection, we analyze the Gini index theory in terms of the value of Gini index by changing the number of sampled classes S^*, the number of classes *N*, and the number of total samples d^*. As shown in Fig. 4, a set *D* is equally divided into *N* classes, where each class is denoted by S_i ($i \in [1, N]$). The d^* samples are uniformly distributed among *k* consecutive classes ($k < N$). As shown in Subfig. 5(a), the overall Gini index value increases as the number of sampled classes increases. Here, the increasing number of sampled classes indicates that the samples will spread over more classes. Since the samples distribute over classes more evenly, the Gini index value is increased. However, as the total number of samples d^* increases, the Gini index value does not change. In Subfig. 5(b), as the number of classes *N* increases from 10 to 20, the overall Gini index value significantly decreases. This is because the increasing number of classes makes the samples more centralized in a certain number of classes, as a result, the Gini index value decreases based on the Gini index theory. In addition, a larger Gini index value is observed with a larger number of sampled classes. Compare to a smaller number of sampled classes, a larger number of sampled classes makes samples more evenly distribute over more classes. Thus, the Gini index value is larger than that of a smaller number of sampled classes.

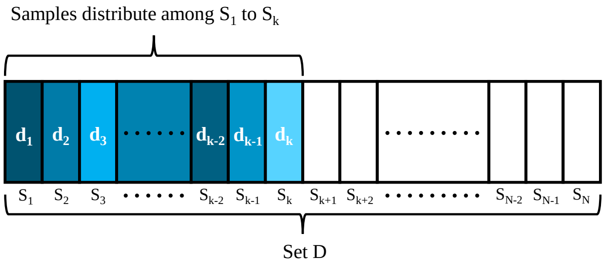


Fig. 4. An example of centralized sample distribution. Here, a set *D* is divided into *N* classes, and samples distributed among class S_1 to S_k.

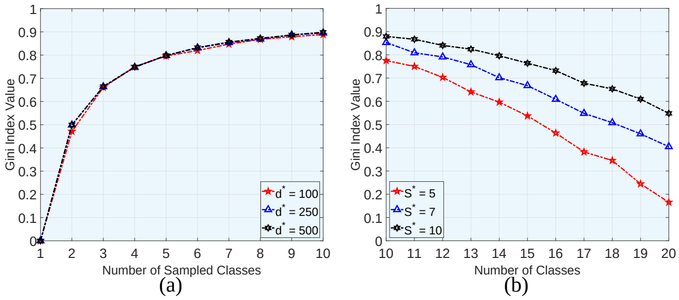


Fig. 5. The change of Gini index value against the number sampled classes S^* and the number of classes *N*.

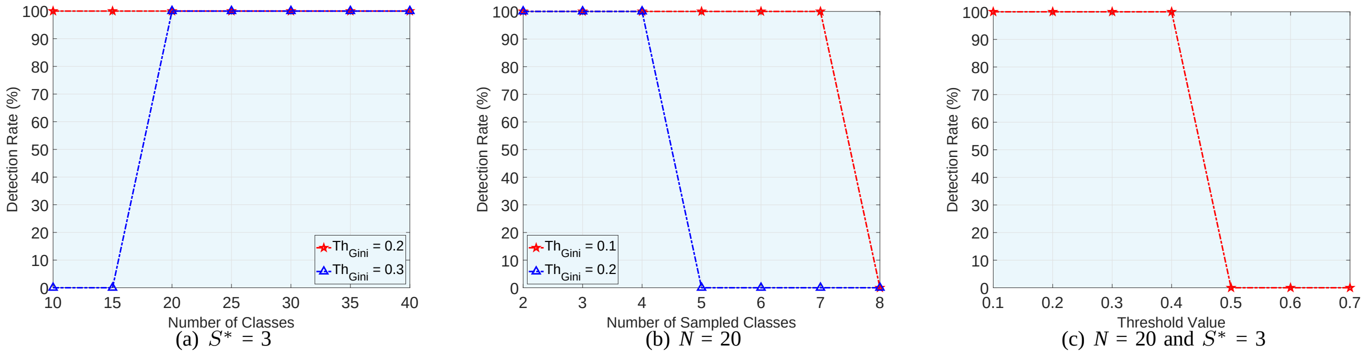
V. Analysis of the Proposed Countermeasure

In this section, we analyze the *GINI* countermeasure in terms of detection rate. According to Eq. 2, if the relative increment of the Gini impurity of current observation window period to that of previous observation window period is larger than a threshold value, the sybil attack can be detected. Otherwise, it results in the miss detection of sybil attack. For example in Subfig. 1(c), suppose that the malicious node n_m starts to launch sybil attack from the $i+1^{th}$ observation window period ω^{i+1}, where the d^*_{i+1} number of DIS messages piggybacked with randomly generated different fictitious identities are multicasted. Here, it is reasonable to assume that the d^*_{i+1} number of fictitious identities are equally distributed among all *N* classes of the entire MAC address space. This is because the malicious node n_m does not know how the legitimate MAC addresses are assigned, it only can randomly generate the fictitious identities and piggyback them in the DIS messages. In addition, let d^*_i and S^* be the number of DIS messages with valid identities from legitimate nodes within the i^{th} observation window period ω^i and a small set of consecutive classes where the d^*_i number of valid identities are generated, respectively.

First, in the observation window period ω^{i+1}, since the d^*_{i+1} number of fictitious identities are equally distributed among all *N* classes, the number of fictitious identities in class *j*, denoted as mac_j, can be approximated by

$$mac_j \approx \frac{d^*_{i+1}}{N} \tag{6}$$

So, the relative frequency of fictitious identities in class *j* can



(a) $S^* = 3$ (b) $N = 20$ (c) $N = 20$ and $S^* = 3$

Fig. 6. Detection rate against the number of classes N, the number of sampled classes S^*, and the threshold value Th_{Gini}.

be represented as

$$p_j = \frac{mac_j}{d^*_{i+1}} = \frac{\frac{d^*_{i+1}}{N}}{d^*_{i+1}} = \frac{1}{N} \tag{7}$$

Thus, the Gini impurity of the d^*_{i+1} number of fictitious identities within the $i+1^{th}$ observation window period ω^{i+1} can be observed as

$$\begin{aligned} Gini(d^*_{i+1}) &= 1 - \sum_{i=1}^{N} p_i^2 \\ &= 1 - \sum_{i=1}^{N} (\frac{1}{N})^2 \\ &= 1 - N \cdot (\frac{1}{N})^2 \\ &= 1 - \frac{1}{N} \end{aligned} \tag{8}$$

Second, using the same idea, the relative frequency of the d^*_i number of valid identities in class j within the observation window period ω^i can be represented as

$$p_j = \begin{cases} \frac{1}{S^*}, & j = 1, 2, \ldots, S^* \\ 0, & j = S^*+1, S^*+2, \ldots, N \end{cases} \tag{9}$$

Thus, the Gini impurity of the d^*_i number of valid identities within the i^{th} observation window period ω^i can be observed as

$$\begin{aligned} Gini(d^*_i) &= 1 - \sum_{i=1}^{N} p_i^2 \\ &= 1 - \Big(\sum_{i=1}^{S^*} p_i^2 + \sum_{i=S^*+1}^{N} p_i^2\Big) \\ &= 1 - \sum_{i=1}^{S^*} p_i^2 \\ &= 1 - S^* \cdot (\frac{1}{S^*})^2 \\ &= 1 - \frac{1}{S^*} \end{aligned} \tag{10}$$

Third, according to Eq. 8 and 10, the relative increment of the Gini impurity between observation window period ω^i and ω^{i+1}, denoted as Imp^i, can be represented as

$$\begin{aligned} Imp^i &= \frac{Gini(d^*_{i+1}) - Gini(d^*_i)}{Gini(d^*_i)} \\ &= \frac{(1-\frac{1}{N}) - (1-\frac{1}{S^*})}{1-\frac{1}{S^*}} \\ &= \frac{\frac{1}{S^*} - \frac{1}{N}}{1-\frac{1}{S^*}} \end{aligned} \tag{11}$$

Finally, the detection rate and miss detection rate of sybil attack can be represented as

$$\begin{cases} \frac{\frac{1}{S^*}-\frac{1}{N}}{1-\frac{1}{S^*}} > Th_{Gini,i} & \Longrightarrow detection \\ \frac{\frac{1}{S^*}-\frac{1}{N}}{1-\frac{1}{S^*}} <= Th_{Gini,i} & \Longrightarrow miss\ \ detection \end{cases} \tag{12}$$

In Fig. 6, we show numerical results of detection rate of sybil attack against the number of classes N, the number of sampled classes S^*, and the threshold value Th_{Gini}. Here, the number of classes N varies from 10 to 40, while the number of sampled classes S^* is set to [2, 8]. The threshold value Th_{Gini} is between 0.1 and 0.7. As shown in Subfig. 6(a), with the threshold value Th_{Gini} = 0.2, the detection rate can be maintained as high as 100% as the number of classes N increases. When Th_{Gini} = 0.3, the sybil attack cannot be successfully detected as the number of classes N varies from 10 to 15. Since the number of sampled classes S^* is close to the number of classes N, the Gini impurity of the identities distributed over S^* and N is also very close. As a result, the adversarial scenario and normal scenario cannot be clearly distinguished and the sybil attack is not able to be detected. In Subfig. 6(b), the overall detection rate decreases as the number of sampled classes S^* increases. With a larger S^*, the identities are distributed over more classes in the entire MAC address space, resulting in a larger Gini impurity within the observation window period. As a result, the relative increment of two consecutive observation window periods becomes smaller, and the sybil attack cannot be successfully detected when the relative increment is less than the threshold value. As the Th_{Gini} increases to 0.2, a lower detection rate is observed with a varying S^*. In Subfig. 6(c), as the threshold value Th_{Gini} increases, the overall detection rate decreases. Since a larger Th_{Gini} requires a larger increment of Gini impurity between two consecutive observation window periods to detect sybil attack, a lower detection rate is observed.

VI. Performance Evaluation and Analysis

A. Simulation Testbed and Benchmark Schemes

We conduct extensive simulation experiments using OMNeT++ [12] to evaluate the performance of *GINI*. A 100 × 100 m^2 square network area is considered, where 20 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 [38]. The communication range of each node is 30 meters. To emulate the scenario that a node runs out of its battery or is damaged and new nodes are deployed, 1 to 3 legitimate nodes are randomly generated and join the network. 1 to 3 malicious nodes are randomly located in the network, and multicast malicious DIS messages with sybil attack rate from 0.1 to 3.0 pkt/sec. Here, sybil attack rate is the number of DIS messages with fictitious identities broadcasted by a malicious node. The total simulation time is set to 1000 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, isolation latency, and energy consumption by changing key simulation parameters, including sybil attack rate ($sybil_{atk}$) and the size of observation window (ω).

We revisit prior SecRPL scheme [10] and two-step detection approach (Two_Step) [11], and modify them to work in the framework for detecting sybil attack. The original idea of these two benchmark schemes are briefly discussed below:

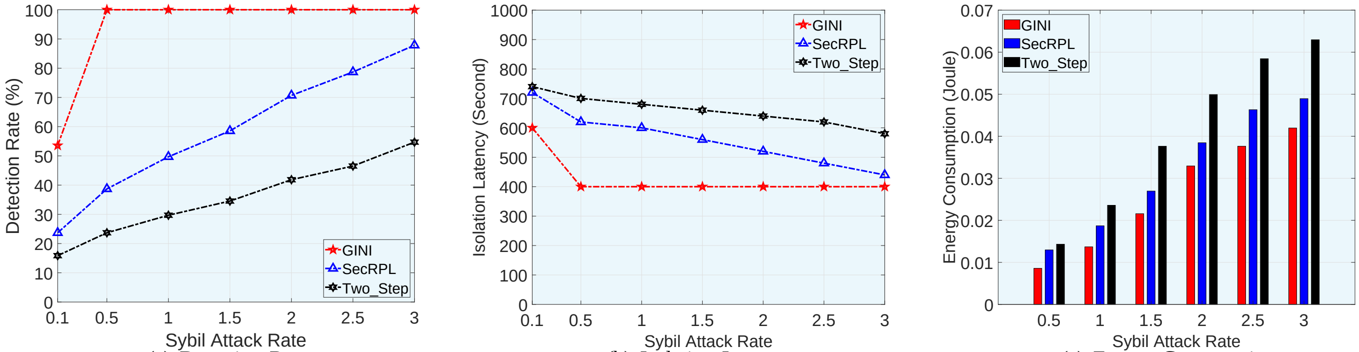
- In order to defend against a DAO insider attack in RPL, a defense mechanism named SecRPL is proposed. In SecRPL, each parent node associates a counter with every child node in its sub-DODAG. When the number of forwarded DAO messages for a child node exceeds a pre-specified threshold value, the parent node discards any received DAO messages piggybacked with the prefix of the respective child node. To avoid blocking DAO messages from legitimate nodes, the counter of forwarded DAO messages is reset between each two consecutive DIO messages. In other words, the counters for all child nodes will be reset when the parent node sends out a DIO message.
- A two-step detection approach consisting of local monitoring and global verification is proposed to defend against blackhole attack in RPL-based networks, where a malicious node silently drops all the received Data packets. In particular, each node observes the forwarding behaviors of one-hop neighbor nodes by overhearing Data packets transmitted by its neighbor nodes. If a node does not overhear the Data packet transmitted by a neighbor node, the misbehaving activity of neighbor node is increased by one. When the detected misbehaving activity events exceed the threshold value, the monitoring node suspects the forwarding misbehaviors of the neighbor node and sends the verification packet to the DODAG root to verify whether the sending or forwarding packet was received or not.

B. Simulation Results and Analysis

First, we measure the detection rate by changing the sybil attack rate in Subfig. 7(a). As the sybil attack rate increases, the overall detection rate of *GINI*, SecRPL, and Two_Step significantly increase. This is because more malicious DIS messages with fictitious identities are generated and broadcasted with a larger sybil attack rate, more sybil attack attempts can be detected. As a result, the overall detection rate increases. The highest detection rate is observed by *GINI* because each node calculates the Gini index value of two consecutive observation windows to detect any abnormal change of DIS message receiving rate, more sybil attacks can be detected and a larger detection rate is achieved. The SecRPL deploys a pre-specified threshold value of DIS messages to detect the sybil attack. However, when the number of received malicious DIS messages is less than the threshold value, the sybil attack can not be detected. As a result, the total number of detected sybil attack decreases, and the detection rate of SecRPL is lower than that of *GINI*. The Two_Step shows the lowest detection rate. In the Two_Step, a node collects the DIS message receiving rate from all its one-hop neighbor nodes to detect the potential sybil attack. However, the malicious node actually broadcasts malicious DIS messages, which will increase the DIS message receiving rate of all neighbor nodes. Subsequently, the average DIS receiving rate of neighbor nodes increases, the detection node can not successfully detect sybil attack by comparing its DIS message receiving rate with average rate. Therefore, the detection rate of Two_Step falls behind *GINI* and SecRPL.

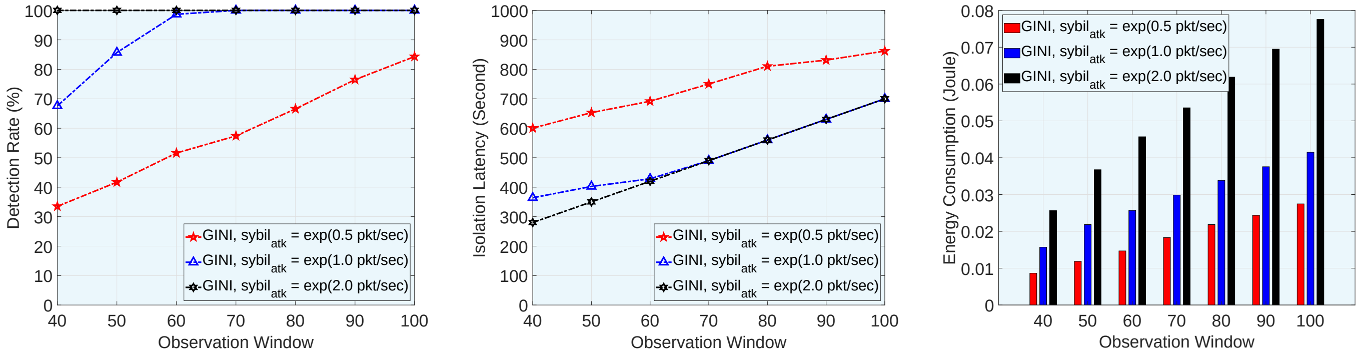
Second, we measure the isolation latency by varying the sybil attack rate in Subfig. 7(b). The isolation latency is the amount of elapsed time when *Isolate* packet is broadcasted. It is shown that the lowest isolation latency is obtained by the *GINI*. Since a larger number of sybil attack can be detected within a short amount of time, when the number of detected sybil attack reaches a threshold value, an *Isolate* packet can be broadcasted earlier. The SecRPL provides a larger isolation latency than that of *GINI*. This is because the SecRPL has lower detection rate and some sybil attack attempts can not be successfully detected. As a result, the number of detected sybil attack needs a long period of time to reach the threshold value to broadcast *Isolate* packet, and a longer isolation latency is observed. The highest isolation latency is obtained by the Two_Step because it has the lowest detection rate and more time are needed to detect enough sybil attacks to isolate the malicious nodes.

Third, the energy consumption of *GINI*, SecRPL, and Two_Step are measured by changing the sybil attack rate in Subfig. 7(c). In this paper, the detection cost of the *GINI* can be categorized into communication cost and computation



(a) Detection Rate (b) Isolation Latency (c) Energy Consumption

Fig. 7. The performance of detection rate, isolation latency, and energy consumption against sybil attack rate.



(a) Detection Rate (b) Isolation Latency (c) Energy Consumption

Fig. 8. The performance of detection rate, isolation latency, and energy consumption against observation window.

cost. Communication cost is the energy consumption due to message forwarding and receiving, which is measured based on the number of forwarded and received messages (e.g., DIS and DIO messages). Computation cost comes when calculating the Gini impurity of current observation window period and comparing it with the Gini impurity of previous observation window period. However, the energy consumption of internal operations is negligible compared to that of communication operations, such as transmitting and receiving operations executed in a wireless network interface card (NIC). Thus, the energy consumption is only measured based on the number of forwarded and received messages [39]. As the sybil attack rate increases, the energy consumption of three schemes increase. With a larger sybil attack rate, more malicious DIS messages will be broadcasted by the malicious nodes. Thus, the legitimate nodes receive more DIS messages as well as broadcast more DIO messages, which results in a significant energy consumption. However, the energy consumption of *GINI* is lower than that of SecRPL and Two_Step. This is because the *GINI* can isolate the potential sybil attack earlier than SecRPL and Two_Step, the legitimate nodes in *GINI* will receive and reply a less number of control messages (e.g., DIS and DIO) and a less amount of energy will be consumed. The highest energy consumption is achieved by Two_Step because the sybil attack can not be mitigated at the earliest convenience, more energy has to be spent on receiving and replying control messages.

Fourth, we measure the detection rate of *GINI* by changing the size of observation window in Subfig. 8(a). When the sybil attack rate is large enough (e.g., $sybil_{atk}$ = 2.0 pkt/sec), the detection rate of *GINI* can be maintained as high as 100% without being affected by the size of observation window. This is because a large number of malicious DIS messages are generated and broadcasted by malicious nodes with a larger sybil attack rate, the difference between the Gini index value within two consecutive observation windows is always larger than the threshold value, all sybil attack attempts can be detected. With a lower sybil attack rate such as $sybil_{atk}$ = 0.5 pkt/sec, as the size of observation window increases, the detection rate of *GINI* increases as well. Since a larger observation window makes the legitimate node receive more malicious DIS messages, more sybil attack can be detected and an increment can be achieved in the detection rate.

Fifth, the performance of isolation latency and energy consumption of *GINI* are shown in Subfig. 8(b) and Subfig. 8(c), respectively. In Subfig. 8(b), as the size of observation window increases, the isolation latency of *GINI* increases. Since each node can only detect the sybil attack attempt at the end of observation window, when the observation window is extended, the detection of sybil attack in each observation window will be delayed. As a result, the overall isolation latency is increased as the size of observation window increases. With a larger sybil attack rate, a lower isolation latency can be achieved because the required number of sybil attack detection can be achieved within a less number of observation windows, and the *Isolate* packet can be broadcasted earlier. In Subfig. 8(c), the energy consumption increases as the size of observation window increases. Within a longer observation window, each legitimate node will receive more malicious DIS messages and reply more DIO messages when the malicious nodes increase sybil attack rate. As a result, each legitimate node will consume more energy and the overall energy consumption will be increased.

VII. Discussion

In this section, we first investigate the proposed *GINI* in terms of design features, design constraint, and possible improvement. Then, we discuss the immunity of the proposed *GINI* against other security attacks in the Internet of Things.

A. Design Features, Design Constraint, and Possible Improvement

In summary, the *GINI* is designed with three desirable features: lightweight detection, quick adaptability, and complete isolation. First, according to the nature of detection mechanism, the *GINI* belongs to monitor-based approach [16], where each node only passively overhears and records DIS messages from adjacent nodes to detect potential sybil attack. Compared to acknowledgment-based approach, where a large number of explicit control packets (i.e., acknowledgment (Ack) packet) are generated to detect the potential adversary, the *GINI* can significantly reduce the number of generated control packets, achieving the goal of energy saving. Second, the *GINI* is sensitive and adaptive to the change of network state and can quickly react to different sybil attack patterns. When the sybil attack is aggressive by broadcasting a large number of DIS messages with fictitious identities, the *GINI* can quickly detect the sybil attack within two consecutive observation windows, and significantly reduce the DIO message replying rate to reduce the impact of sybil attack. However, if there is no sybil attack and the network structure experiences normal change, the DIO message replying rate can be maintained at a relatively high level. Third, in order to completely isolate the potential adversary, when the number of detected sybil attack reaches a threshold value, the detection node will broadcast an *Isolate* packet to prevent adjacent nodes from accepting any DIS messages from new nodes nearby. After a long period of time, the *GINI* will slowly increase the DIO replying rate to probe for network state.

The *GINI* has one design constraint that needs to be further discussed for further improvement. The *GINI* is designed based on an assumption that the MAC addresses of all legitimate nodes in the network are assigned and centralized in a small set of consecutive classes, rather than randomly distributing among all classes of the entire MAC address space. Since the adversary does not know exactly how the MAC addresses of legitimate nodes are assigned, it only can use randomly generated fictitious and unreachable destination MAC address to create DIS messages with fictitious identifier. As a result, the MAC addresses in the received DIS messages randomly spread out in the entire MAC address space, which causes the Gini index value to increase significantly. However, if the MAC addresses of all legitimate nodes in the network are also randomly assigned and distributed in the entire MAC address space, the proposed Gini index-based countermeasure probably cannot accurately detect sybil attack because the Gini index value in normal and attack scenarios does not have significant difference.

To see the full potential and overcome the intrinsic weakness of the *GINI*, we plan to investigate the following for further improvement and extension. We propose to let each node estimate the variation of received signal strength indication (RSSI) based on Chebyshev inequality [40], [41] to assist the Gini index-based approach with detecting sybil attack. In any data sample or probability distribution, Chebyshev inequality indicates that the strictly positive expectation $E(X)$ and the variance $var(X)$ have the following inequality with the discrete variable X:

$$P\{|X - E(X)| < \varepsilon\} \geq 1 - \frac{var(X)}{\varepsilon^2}. \tag{13}$$

Eq. 13 reflects that the random variable X is relative stable, when variance $var(X)$ tends to be zero. In addition, we can obtain

$$var(X) = E(X^2) - E(X)^2 \tag{14}$$

and

$$E(X) = \sum_i \frac{X_i}{n}. \tag{15}$$

Thus, $var(X)$ can be represented as

$$var(X) = \Big(\sum_i \frac{X_i^2}{n}\Big) - \Big(\sum_i \frac{X_i}{n}\Big)^2. \tag{16}$$

Since most radio transceivers contains RSSI register, the signal strength of received packet can be easily obtained [42]. Thus, when a node receives an excessive number of DIS messages from adjacent nodes, it can easily obtain the RSSI information of received DIS messages to detect whether there is a sybil attack. Here, the RSSI can be used to replace the variable X in Eq. 16. If the observed RSSI values of DIS messages are very stable, $var(X)$ will be approaching to zero, which indicates that all DIS messages might be broadcasted by the potential adversary. However, the essential prerequisite for using this RSSI-based approach is that the adversary does not vary transmission power of DIS messages to trick the receiver.

B. Immunity to Other Attacks

In this subsection, we discuss the *GINI* to see whether it is immune to other three well-known attacks: DAO insider attack [10], DIO suppression attack [43], and energy depletion attack [44].

1) DAO Insider Attack: In RPL, the DODAG Destination Advertisement Object (DAO) messages are generated by the child nodes and sent to their parent nodes to build downward routes. However, an adversary can send fake DAO messages to its parent nodes periodically to trigger parent nodes to forward the fake DAO messages upward to the root node. This attack is similar to sybil attack and it can be easily detected by the *GINI*. For example in Subfig. 1(c), suppose that adversary n_m periodically generates and sends fake DAO messages with fictitious identifies to legitimate node n_f. However, n_f can record a trace of the received DAO messages during each observation window and measures the dispersity of identities of DAO messages. If the difference of the Gini index value between two consecutive observation windows is larger than a threshold value, the DAO insider attack can be detected, and n_f can significantly reduce the DAO messages forwarding rate to defend against DAO insider attack.

2) DIO Suppression Attack: A malicious node can periodically replay previously overheard DODAG Information Object (DIO) messages to induce victim nodes to suppress the transmission of DIO messages, which are the RPL control messages necessary to build the routing topology. The DIO suppression attack can cause a degradation of the routes' quality, and eventually leads to network partitions. Unlike other RPL attacks, the DIO suppression attack does not require the adversary to steal cryptographic keys from legitimate nodes. Thus, it is not trivial to avoid DIO suppression attack, but this attack can be detected by the *GINI*. For example in Subfig. 1(c), suppose n_m periodically broadcasts the overheard DIO messages with a fixed interval to suppress the transmission of DIO message of n_f. In the *GINI*, n_f can record the identifiers of the received DIO messages from one-hop neighbor nodes within an observation window, and then calculates the Gini index value. If the Gini index value is close to zero, n_f can detect the DIO suppression attack. This is because the adversary n_m only replay the previously overheard DIO messages, the identifiers of all received DIO messages will be centralized in one class. As a result, the Gini index value tends to be zero and the DIO suppression attack can be detected.

3) Energy Depletion Attack: In RPL, point-to-point routing mechanism is primarily used to initiate data transfer, send end-to-end acknowledgments, or carry out infrequent network diagnostics. Unfortunately, the vulnerability of point-to-point routing mechanism can be exploited by adversary to launch energy depletion attack [42], where an adversary intentionally sends a large number of packets with fictitious destination addresses to excessively consume the energy resource of intermediate nodes located along the forwarding path, and finally makes the resource-constrained nodes suffer from denial of service. The energy depletion attack primarily targets point-to-point routing mechanism's vulnerabilities by violating an implicit assumption that all intermediate nodes faithfully and collaboratively forward the received packets to destination node. It is not trivial to eliminate energy depletion attack, but this attack can be successfully detected by the GINI. In Subfig. 1(c), suppose that the malicious node n_m sends a packet with the source route ([n_m, n_e, n_d, n_a, n_R, n_X]) to destination node n_X. Here, n_X is the fictitious destination node address that does not exist in the network. In the GINI, n_e can record the destination node addresses of received packets within a time period, and then calculate the Gini index value. If the Gini index value is larger than the threshold value, it claims to detect energy depletion attack performed by child node n_m.

VIII. 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, detection latency, and energy consumption, indicating a viable approach against sybil attack in IoT. 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 Startup grant in the Weisberg Division of Computer Science and 2019 Summer Research Award at Marshall University. This work was published in part at the 2019 IEEE/AFCEA Military Communications Conference (MILCOM) [45]. The author would like to thank the four anonymous reviewers of MILCOM for providing their constructive comments to improve the overall quality of the manuscript.

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] L. Horwitz, *The Future of IoT Miniguide*, https://www.cisco.com/c/en/us/solutions/internet-of-things/future-of-iot.html.

[4] J. Pittman, *Forget The Consumer Internet Of Things*, https://www.ge.com/reports/forget-consumer-internet-things-iiot-really/.

[5] Cisco, *Smart-Grid Last-Mile Infrastructure*, https://www.cisco.com/c/en/us/products/collateral/routers/1000-series-connected-grid-routers/white-paper-c11-730860.html.

[6] “Cisco Connected Grid Security for Field Area Network - White Paper,” 2012.

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

[8] 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.

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

[10] B. Ghaleb, A. Al-Dubai, E. Ekonomou, M. Qasem, I. Romdhani, and L. Mackenzie, “Addressing the DAO Insider Attack in RPL's Internet of Things Networks,” *IEEE Commun. Lett.*, vol. 23, no. 1, pp. 68–71, 2018.

[11] F. Ahmed and Y. Ko, "Mitigation of black hole attacks in Routing Protocol for Low Power and Lossy Networks," *Security and Communication Networks*, vol. 9, no. 18, pp. 5143–5154, 2016.

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

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

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

[19] 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.

[20] 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.

[21] G. Glissa, A. Rachedi, and A. Meddeb, "A Secure Routing Protocol Based on RPL for Internet of Things," in *Proc. IEEE GLOBECOM*, 2016, pp. 1–7.

[22] A. Rachedi and A. Benslimane, "Multi-objective optimization for security and QoS adaptation in Wireless Sensor Networks," in *Proc. IEEE ICC*, 2016, pp. 1–7.

[23] A. Rachedi and A. Hasnaoui, "Advanced quality of services with security integration in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 15, no. 6, pp. 1106–1116, 2015.

[24] H. Nguyen-Minh, A. Benslimane, and A. Rachedi, "Jamming Detection on 802.11p under Multi-channel Operation in Vehicular Networks," in *Proc. IEEE WiMob*, 2015, pp. 764–770.

[25] S. Murali and A. Jamalipour, "A Lightweight Intrusion Detection for Sybil Attack under Mobile RPL in the Internet of Things," *IEEE Internet of Things Journal (Early Access)*, pp. 1–1, 2019.

[26] D. Airehrour, J. Gutierrez, and S. Ray, "SecTrust-RPL: A secure trust-aware RPL routing protocol for Internet of Things," *Future Generation Computer Systems*, vol. 93, pp. 860–876, 2019.

[27] A. Mishra, A. Tripathy, D. Puthal, and L. Yang, "Analytical Model for Sybil Attack Phases in Internet of Things," *IEEE Internet of Things*, vol. 6, no. 1, pp. 379–387, 2019.

[28] Y. Yao, B. Xiao, G. Yang, Y. Hu, L. Wang, and X. Zhou, "Power Control Identification: A Novel Sybil Attack Detection Scheme in VANETs using RSSI," *IEEE J. Sel. Areas Commun.*, vol. DOI: 10.1109/JSAC.2019.2933888, pp. 1–1, 2019.

[29] M. Al-Qurishi, M. Alrubaian, S. Rahman, A. Alamri, and M. Hassan, "A prediction system of Sybil attack in social network using deep-regression model," *Future Generation Computer Systems*, vol. 87, pp. 743–753, 2018.

[30] M. Ayaida, N. Messai, S. Najeh, and K. Ndjore, "A Macroscopic Traffic Model-based Approach for Sybil Attack Detection in VANETs," *Ad Hoc Networks*, vol. 90, p. 101845, 2019.

[31] A. Vasudeva and M. Sood, "Survey on sybil attack defense mechanisms in wireless ad hoc networks," *Journal of Network and Computer Applications*, vol. 120, pp. 78–118, 2018.

[32] K. Zhang, X. Liang, R. Lu, and X. Shen, "Sybil Attacks and Their Defenses in the Internet of Things," *IEEE Internet of Things*, vol. 1, no. 5, pp. 372–383, 2014.

[33] A. Raoof, A. Matrawy, and C. Lung, "Routing Attacks and Mitigation Methods for RPL-Based Internet of Things," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1582–1606, 2019.

[34] Y. Lu and D. Li, "Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics," *IEEE Internet of Things*, vol. 6, no. 2, pp. 2103–2115, 2019.

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

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

[37] 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.

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

[39] X. Tang and J. Xu, "Extending Network Lifetime for Precision-Constrained Data Aggregation in Wireless Sensor Networks," in *Proc. IEEE INFOCOM*, 2006, pp. 1–12.

[40] V. Csiszár and T. Móri, "A Bienaymé-Chebyshev Inequality for Scale Mixtures of the Multivariate Normal Distribution," *Math Inequal Appl*, vol. 12, no. 4, pp. 839–844, 2009.

[41] A. Moussaoui, F. Semchedine, and A. Boukerram, "A link-state QoS routing protocol based on link stability for Mobile Ad hoc Networks," *Elsevier Net. and Comp. Appl*, vol. 39, pp. 117–125, 2014.

[42] C. Pu, "Link-Quality and Traffic-Load Aware Routing for UAV Ad Hoc Networks," in *Proc. IEEE CIC*, 2018, pp. 71–79.

[43] 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.

[44] V. Nguyen, P. Lin, and R. Hwang, "Energy Depletion Attacks in Low Power Wireless Networks," *IEEE Access*, vol. 7, pp. 51 915–51 932, 2019.

[45] B. Groves and C. Pu, "A Gini Index-Based Countermeasure Against Sybil Attack in the Internet of Things," in *IEEE Proc. MILCOM*, 2019, pp. 672–677.