Analyzing the Impact of Topology on Flooding Attacks in Low-power IoT Networks

Jack Zhao
Dalhousie University
jackz@dal.ca

Xinyu Liu
Dalhousie University
xinyu.liu@dal.ca

Michael Baddeley
Technology Innovation Institute (TII)
michael@ssrc.tii.ae

Israat Haque
Dalhousie University
israat@dal.ca

***Abstract*—Low-power Internet of things (IoT) networks can support various applications like smart agriculture or smart manufacturing. These devices usually rely on the commonly-used *routing protocol for low-power and lossy networks (RPL)* protocol to exchange messages. RPL suffers from various attacks, where the DODAG Information Solicitation (DIS) flooding attack is the most common but effective attacking method. To figure out the different factors in this attack, we analyze how the number of attackers and their location from DODAG root can influence the DIS flooding attack as such systematic analysis is missing in existing works. Extensive evaluation over Contiki-NG-based Cooja simulator reveals that attackers' number significantly damages devices' energy consumption and packet delivery ratio than the position of attackers.**

***Keywords*—**Low-power IoT, RPL, DIS flooding attack, Contiki-NG, Cooja

I. Introduction

Low-power IoT networks have a wide range of applications such as smart health, smart home, and smart agriculture. We can use low-power wireless sensors to monitor air quality [1], reduce energy consumption in manufacturing companies [2], or improve the collaboration among objects for better energy utilization [3]. These low-power sensors can form multihop communication networks and deploy standard *Routing Protocol for Low-Power and Lossy Networks (RPL)* for message exchanges [4]. However, researchers have shown RPL suffers from many vulnerabilities that can significantly degrade the overall network performance (e.g., packet delivery ratio) [5] [6] [7].

Moreover, compared with the legacy networks, wireless networks have more chance to be silently modified by the attacker in order to change the data transmission or increase packet overhead of the network. It could even have the potential to cause massive attacks to disable partial or entire networks. For example, if the targeted network is part of the power plant infrastructure, it might cause power failure or blackout for a large portion of the community [8]; or if the target network is embedded in a traffic control system, it might cause traffic jams or car accidents. Therefore, the attacks need to be mitigated and eliminated.

Our study shows that among different RPL attacks, the DIS flooding attack is one of the most common and influential ones to deploy from outside the targeted IoT networks. There are different mitigation approaches that exist [9] [10] [11]. But most of them only evaluate their results using grid or random topology; there is no systematic analysis on how the topology structure influences the effectiveness of RPL attacks. Therefore, in this study, we want to analyze the influence of the network topology, i.e., the hop distance and the number of attackers on the DIS flooding attack. The analysis will allow designing useful attack mitigation schemes.

The rest of this report is organized as follows. In Section II, we provide the necessary background on RPL and DIS flooding attacks. Section III briefly describes the related work. Then we write about the system design of our study in Section IV. The evaluation is presented in Section V, along with the discussion and the challenges faced in Section VI. Finally, we conclude the paper in Section VII.

II. Background

RPL. RPL is an IPv6-based distance vector and source routing protocol that specifies how to build a Destination Oriented Directed Acyclic Graph (DODAG) using an Objective Function (OF) to meet a set of metrics and constraints [5]. The RPL control message includes four elements: DODAG Information Solicitation (DIS), DODAG Information Object (DIO), DODAG Advertisement Object (DAO), DAO-ACK.

Figure 1 shows a brief demonstration of the RPL control messages exchange mechanism. When a new node wants to join the network, it first sends out the DIS to request information from nearby DODAG. After that, the network sends back a DIO message in response to the DIS containing the network's information. Then the new node will send DAO, which includes the direction of DODAG to join the original network. After the original network received the DAO, it then replies with the DAO-ACK. Since there is no mechanism to validate control messages in the original RPL, all RPL attacks exploit manipulating control messages. Therefore, RPL uses an adaptive timer mechanism called *Trickle timer* to limit the control traffic in the network and reduce the influence of attacks that manipulate the control messages [12].

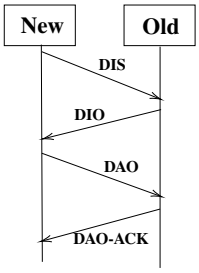

Fig. 1. RPL control messages.

DIS Flooding attack. In the DIS Flooding attack, a compromised node periodically sends DIS messages to neighbors within its transmission range. In return, a victim node resets its timer and replies with DIO messages. This process can be done either by sending unicast or multicast DIS messages. In the unicast DIS flooding attack, the attacker sends out unicast DIS messages, any node that receives this message needs to send back a DIO message. Similarly, in the multicast DIS flooding attack (in Figure 2), the attacker sends out multicast DIS messages; all nodes that receive this message send DIO messages to their neighbors. It leads to an increase in control packet overhead, node energy exhaustion, and routing disruption. We use multicast DIS flooding attacks to create the most significant impact.

Figure 2 shows a DIS flooding attack scenario. In this case, node 1 is the root node; nodes 2 to 10 are normal nodes. Node 11 is the

attacker that sends multicast DIS messages in its range. Nodes 2, 3, 5, 6, 8, and 9, after receiving that DIS message reset their trickle timer. Finally, they send multicast DIO messages to their neighbors.

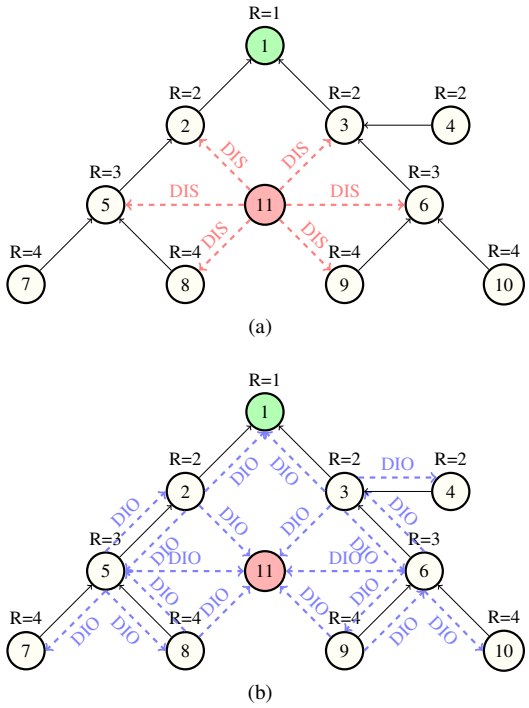

Fig. 2. DIS Flooding Attack. (a) The attacker, node 11, sends out multicast DIS messages within its range. (b) The target nodes 2, 3, 5, 6, 8, and 9 received the DIS message, reset their trickle timer and send out multicast DIO messages; the attack increases network overhead and power consumption.

There are four attacks taking advantage of the RPL design to deploy attacks from outside an IoT network: DIS flooding, Hello flooding, Clone ID, and DODAG Inconsistency attack [5] [6] [7]. However, our further studies show that all these four attacks directly or indirectly use the DIS flooding attack. The Hello flooding is inherited from Wireless Sensor Networks (WSN) [5], where an attacker sends out unnecessary Hello packets to its neighbor, causing unexpected damage in the entire network. In an RPL based network, DIS message is the only Hello packet. The Hello flooding attacker sends out the DIS messages (Hello packets) to its neighbor to trigger those neighbors to send out DIO messages causing unnecessary energy consumption [6] [13].

The Clone ID attack takes advantage of the lack of security and confidentiality implementation of the RPL-based network. The attacker first gets the DODAG configuration directly from the target IoT network, then it listens and clones the identity information from one of the nodes in the network. The attacker uses the collected data to imitate this selected node and access the target network. Then, the attacker uses the DIS flooding attack for the actual damage [11]. DODAG Inconsistency attack creates an inconsistency to trigger a targeted node to start sending DIS messages [9]. Therefore, we conclude that the DIS flooding attack is the most effective and essential component in an RPL attack.

III. Related Work

Smith et al. show how attackers can use various attacks to drain battery [13]. First, in a scenario of one server, five normal nodes, and one attacker, the results show that flooding attacks will vastly increase energy consumption and affect the operation of the equipment. Second, it shows that even if a small number of packets are sent, the battery of a typical IoT device can quickly run out within an hour [13].

In addition, the distance affects the efficiency of attacks. It finds that devices close to malicious nodes are more vulnerable to attacks than other devices. However, the paper only focuses evaluations on different types of attacks with random topology and lacks analysis of the influence on hop distance and number of attackers. Thus, we analyze the attacker's impact for different hop distances and their number.

IV. System Design

This study focuses on two factors that can change the DIS flooding attack's impact on IoT networks: 1) the hop distance from the attacker to the root/server and 2) the number of attackers. We construct a customized grid topology to get the exact hop distance and attacker number in the topology. In that topology, senders generate UDP traffic and deploy RPL protocol to convey that traffic to the root. These senders periodically generate the UDP messages for the root.

Our study uses the state-of-the-art Contiki-NG to implement our experiments in the Cooja simulator. Contiki-NG is an open-source, cross-platform operating system designed for IoT devices. It focuses on dependable (secure and reliable) low-power communication and standard protocols, such as IPv6/6LoWPAN and RPL [14]. To get the power consumption, we use the energy measure module in the Contiki-NG called Energest and calculate the exact energy cost for each specific hardware according to the datasheet specification. Our script could measure the packet loss rate and power consumption for each node. While our current experiment focuses on the total energy consumption, we plan to provide a per node analysis in the future study.

V. Evaluation

A. Evaluation setup

Our project uses Ubuntu 18.04.4 LTS operating system in a VirtualBox with two cores processor and 4096MB memory. We use the latest Contiki-NG and the Cooja (x86 32-bit) simulator [15]. Our project code is available on GitHub [16]. In the Cooja simulator, we measure the packet loss rate and power consumption using the Z1 sensor mote to provide real-life performance results. Each node has a radio range of 20m, and the edge between nodes is 15m. Thus, all the nodes can only reach their neighbors in the topology. We then put those nodes in a 5×5 grid topology and place the server or root in the center of the topology.

Figure 3 shows the grid topology used in our study. In the experiment, node 13 is selected to be the server node, and all other nodes are client nodes or attackers. First of all, our program randomly selects one attacker node with hop distance 1 to the server node (those nodes are nodes 12, 8, 14, 18 in the Figure 3). Then the program select node accordingly with hop distance 2 (node 11, 7, 3, 9, 15, 19, 23, 17), hope distance 3 (node 6, 2, 4, 10, 20, 24, 16, 22), and hop distance 4 (node 1, 5, 25, 21). For each test, the test period is set to 5 minutes and repeat ten times. After we get all the results for one attacker, we increase the number of attackers to two and use the same methodology for the above hop distances. The number of attackers we evaluate is up to four.

B. Evaluation Results

Figure 4 shows the packet loss rate for different numbers of the attackers. Our results show that, in general, as the number of attackers increases, the packet loss rate also increases. In addition, as the hop distance increases, the packet loss rate decreases. Those matched our expectations; since the attacker(s) move closer to the server/root or increase the numbers, the attack should be more damaging. Our studies show that the hop distance plays a major role in the DIS flooding attack's effect in a grid topology. The attack from one hop distance away is the most damaging one, and it can cause the packet

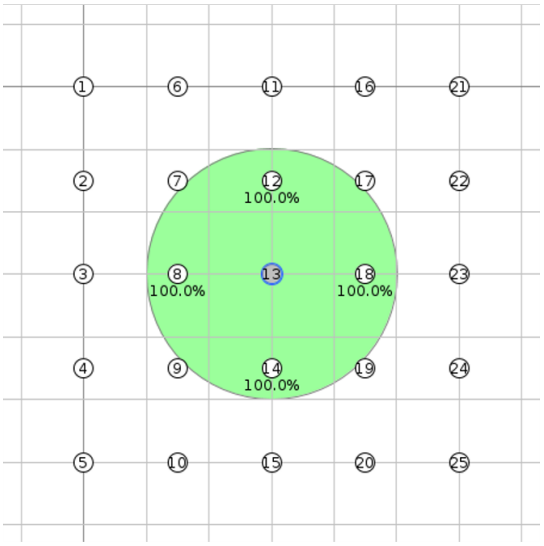

Fig. 3. Evaluation Topology

loss rate to increase more than 102% compared to the attackers from two hop distances away.

Moreover, from the attacker results from hop distance 1, we can see a monotonically increasing packet loss rate when the number of attackers increases. The growth rate gets faster for more attackers. There is only one server with four different node positions around it in this specific topology; therefore, when we place the attacker(s) in those positions, it is more likely to influence the server directly than attackers in the other hop distances. And when there are four attackers in hop distance 1, those four attackers occupy all paths to other nodes.

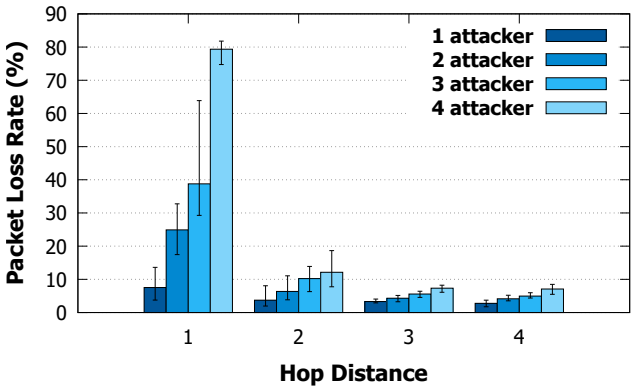

Fig. 4. The average packet loss.

Figure 5 shows the energy consumption from different hop distance for different attackers. The results show that the larger the number of attacks the higher the energy consumption. From the hop distance perspective, we can see that different hop distances have less impact on energy consumption than changing attacker numbers. For the attacker results of hop distance 1, the one attacker result is 2.2% higher than two attackers and 2.6% more from two to three attackers; then it drops to 2.07% increase from three to four attackers. For hop distance 2, growing the attacker's number from one to two can create 2.39% more energy consumption; when the attacker number extends from two to three, the energy rises by 3.8%; it only shows 0.6% improvement from three to four attackers. Thus, the three attackers' situation has a better energy consumption gain; however, most energy consumption occurs between 2–3% when adding an attacker.

Our study shows that when the attacker is closer to the server or the number of attackers is more prominent; the attack consumes more energy. Also, attacks from two hop distances can create the most energy consumption on average, but its output is also unstable compared to other scenarios. We believe this is because the 2-hop distance scenario is more complex than others. Although all the nodes have the same hop distance 2, there are two different groups of physical distance in this scenario, node 3, 11, 15, 23 are 30m away from the root, and node 7, 9, 17, 19 are $15\sqrt{2}$m away. We can see the physical distance difference as the potential influence of this instability; this impact of the attack will be addressed in the future study, alongside different topologies. After the two-hop distance scenario, the energy consumption decrease when the hop distance increases.

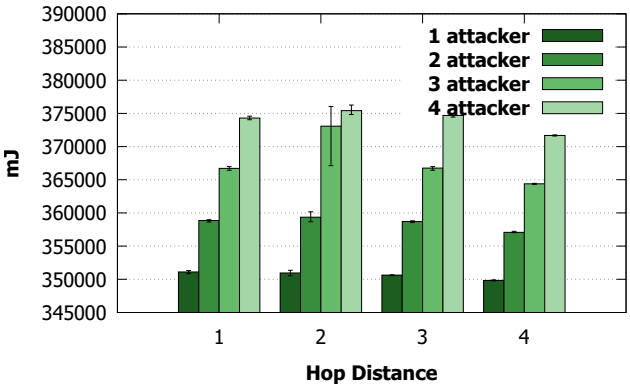

Fig. 5. The average energy consumption.

VI. Conclusion and Future Work

In conclusion, our research indicated that all the state-of-the-arts attacks from outside the IoT networks are directly or indirectly use the DIS flooding attack. Therefore, we evaluated how to deploy the most effective attacks by varying the number of attackers and their distance from the root in a grid topology. We have found that the number of attackers plays a dominant role in increasing the packet loss rate and energy consumption for the outside attacks in an IoT network with grid topology. And the attackers who only intended to influence the packet loss rate should be as closest to the root.

As part of future work, we will first increase the scale of the evaluation on various topologies. Also, we will deploy machine learning-based attack prediction. As the mitigation strategy, we plan to deploy multiple roots in different positions and keep changing their locations to decrease the influence of the attack.

References

[1] H. Hromic, D. Le Phuoc, M. Serrano, A. Antonić, I. P. Žarko, C. Hayes, and S. Decker, "Real time analysis of sensor data for the internet of things by means of clustering and event processing," in *2015 IEEE International conference on communications (ICC)*. IEEE, 2015, pp. 685–691.

[2] F. Shrouf and G. Miragliotta, "Energy management based on internet of things: practices and framework for adoption in production management," *Journal of Cleaner Production*, vol. 100, pp. 235–246, 2015.

[3] K. Wang, Y. Wang, Y. Sun, S. Guo, and J. Wu, "Green industrial internet of things architecture: An energy-efficient perspective," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 48–54, 2016.

[4] M. Baddeley, "Software defined networking for the industrial internet of things," Ph.D. dissertation, University of Bristol, 2020.

[5] A. Verma and V. Ranga, "Security of rpl based 6lowpan networks in the internet of things: A review," *IEEE Sensors Journal*, vol. 20, no. 11, pp. 5666–5690, 2020.
[6] S. M. Muzammal, R. K. Murugesan, and N. Jhanjhi, "A comprehensive review on secure routing in internet of things: Mitigation methods and trust-based approaches," *IEEE Internet of Things Journal*, 2020.
[7] A. Mayzaud, R. Badonnel, and I. Chrisment, "A taxonomy of attacks in rpl-based internet of things," *International Journal of Network Security*, vol. 18, no. 3, pp. 459–473, 2016.
[8] S. Soltan, P. Mittal, and H. V. Poor, "Blackiot: Iot botnet of high wattage devices can disrupt the power grid," in *27th {USENIX} Security Symposium ({USENIX} Security 18)*, 2018, pp. 15–32.
[9] A. Mayzaud, A. Sehgal, R. Badonnel, I. Chrisment, and J. Schönwälder, "Mitigation of topological inconsistency attacks in rpl-based low-power lossy networks," *International Journal of Network Management*, vol. 25, no. 5, pp. 320–339, 2015.
[10] A. Verma and V. Ranga, "Mitigation of dis flooding attacks in rpl-based 6lowpan networks," *Transactions on emerging telecommunications technologies*, vol. 31, no. 2, p. e3802, 2020.
[11] C. D. Morales-Molina, A. Hernandez-Suarez, G. Sanchez-Perez, L. K. Toscano-Medina, H. Perez-Meana, J. Olivares-Mercado, J. Portillo-Portillo, V. Sanchez, and L. J. Garcia-Villalba, "A dense neural network approach for detecting clone id attacks on the rpl protocol of the iot," *Sensors*, vol. 21, no. 9, p. 3173, 2021.
[12] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, "The trickle algorithm," *Internet Engineering Task Force, RFC6206*, 2011.
[13] R. Smith, D. Palin, P. P. Ioulianou, V. G. Vassilakis, and S. F. Shahandashti, "Battery draining attacks against edge computing nodes in iot networks," *Cyber-Physical Systems*, vol. 6, no. 2, pp. 96–116, 2020.
[14] "contiki-ng." [Online]. Available: https://github.com/contiki-ng/contiki-ng
[15] "usdn-ng." [Online]. Available: https://github.com/mbaddeley/usdn-ng/tree/master/tools/vagrant
[16] "rpl_attacks." [Online]. Available: https://github.com/printfer/rpl_attacks