Review on detection mechanisms of Sinkhole Attack on RPL

A report submitted towards partial completion of the degree of

Master of Technology

in

Information System Security and Engineering

by

Maitreyee Sinha

2018PGCAIS05

under the supervision of

Dr. Alekha Kumar Mishra

Assistant Professor

Dept. of Computer Applications
National Institute of Technology jamshedpur
Jamshedpur-831014, India

Contents

1 Introduction 2

2 Introduction to RPL 3

2.1 RPL control messages . . . 3

2.1.1 DODAG Information Object (DIO): . . . 3

2.1.2 DODAG Information Solicitation (DIS): . . . 4

2.1.3 Destination Advertisement Object (DAO): . . . 4

2.2 DODAG construction . . . 4

2.3 DODAG repair . . . 4

3 Security in RPL 5

4 Sinkhole Attack 6

5 Detection mechanisms of Sinkhole attack 7

5.1 Literary survey . . . 7

5.2 Problem statement . . . 10

6 Objective 10

7 Work done 10

8 Road Map 12

9 The proposed work (if any done so far) 12

10 Future work 12

REFERENCES 12

1 Introduction

IoT is a heterogenous network of constrained devices with properties like built-in sensing and communication interfaces as shown in Figure 1 that collects or exchanges data from users. The constrained devices are basically limited of energy, power and memory and are deployed randomly in the area of observation, resulting in Low power and Lossy Network(LLN).

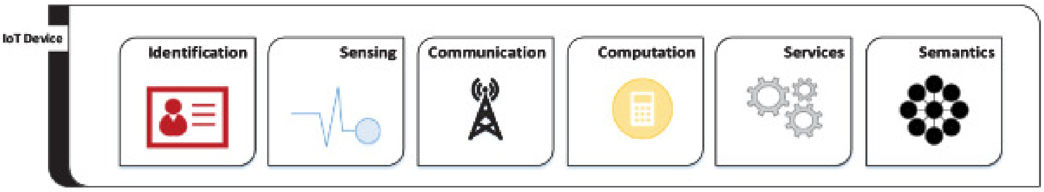


Figure 1: Properties of IoT devices

With such constrained devices to achieve the goal of a secure IoT network, the fundamental properties that has to be satisfied and maintained in the network are: Confidentiality, availability, authenticity, integrity and non-repudiation. To attain and maintain these properties, IoT implements the following security mechanisms. In [1] it has been observed that authentication is the mostly used security mechanism for countermeasures.

i) Authentication: It basically authenticates users in the application layer and also grant access to the devices in the IoT network. For the currently used architecture of IoT there are three authentication protocols designed: asymmetric-cryptosystem based protocols, symmetric-cryptosystem based protocols, and hybrid protocols. Recent reports in **??** shows that it is the most popular method to control unauthorized persons/devices from accessing the network but it still has some weaknesses and to consider this as a perfect solution to IoT security may not be right.

ii) Encryption: As the devices are constrained of resources they cannot support the standard cryptographic primitives so lightweight and low-cost algorithms are designed.

iii) Trust Management: This mechanism is also gaining popularity now. [2] studied few different trust models and provided a summary of each of them. Basically trust is the bonding between two nodes. One node(trustor) counts on the actions performed by another node(trustee). Trust can be evaluated on three categories: General, Situational and Basic Trust.

iv) Secure Routing: RPL is the routing protocol designed for multipoint communication while supporting point to point communication also in an LLN. It is able to meet the specific routing requirements of application areas including urban networks, building automation, industrial automation, and home automation. RPL is broadly discussed in Section 2

Although RPL is efficient in routing in LLN, it is still vulnerable to many security attacks. One such attack is sinkhole attack where the illegitimate node pretends to be more close to the root node.
In this report, first we have discussed about RPL in detail in Section 2 and sinkhole attack in Section 4. Then in Section 5 we have done a literary survey on different detection mechanisms designed to mitigate the sinkhole attack in RPL. And based on the characteristics of the previous techniques possible review is made in Section 7 and a new mechanism is proposed in Section 9.

2 Introduction to RPL

RPL is a distance-vector and a source routing protocol that is designed by IETF working group [ROLL WG] in RFC6550 [3] in March 2012 to operate on top of several link layer mechanisms including IEEE 802.15.4 PHY and MAC layers and an adaptation layer, 6LoWPAN. RPL starts with the generation of the topology through the construction of the Destination Oriented Directed Acyclic Graph(DODAGs).DODAG is tree shaped structure. Unlike traditional tree nodes DODAG nodes cannot have multiple parent nodes in a single DODAG.

2.1 RPL control messages

The construction of the topology is controlled by three control messages:

2.1.1 DODAG Information Object (DIO):

The DIO carries information that allows a node to find a RPL Instance, its relative distance to the root via the sender, learn its configuration parameters, compute the rank according to the objective function,select a DODAG parent set, and maintain the DODAG. DIO is triggered whenever the trickle timer expires [4].The format of DIO base object is in figure2:

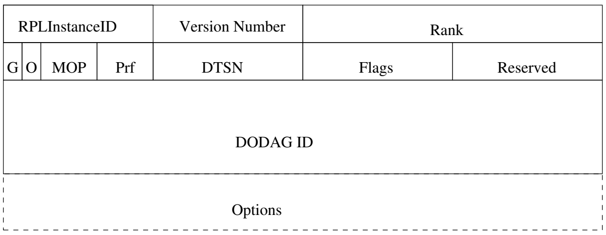


Figure 2: The DIO base object

2.1.2 DODAG Information Solicitation (DIS):

This message is sent when a new node wants to join the network and it cannot wait for the trickle timer to expire.

2.1.3 Destination Advertisement Object (DAO):

DAO is unicasted back to the root node after a DIO is received by the node.Now DAO is sent back either in Storing mode or in Non- Storing mode. In Storing mode, the parent node saves the DAOs from its child nodes and aggregate its address in the reverse route stack and forward it in upward direction. In Non-Storing mode, the node directly unicasts the DAO to root node, intermediate nodes just pass on the message in the upward direction without saving the DAO. Now when an intermediate node receives a DAO from a child node it sets the DelayDAO timer whose value is implementation dependent then checks the DAOSequence number and K-flag in the DAO and then the mode of the receiving node, as shown in figure 4. The format of DAO base object is in figure 3:

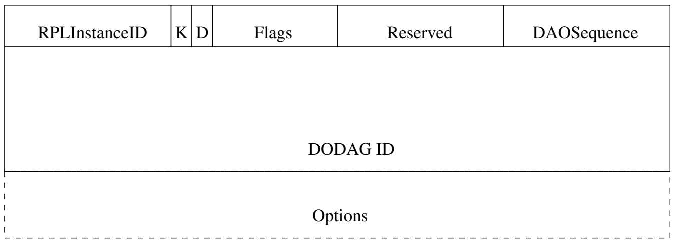


Figure 3: The DAO base object

2.2 DODAG construction

The root node of the tree starts the construction of the topology by broadcasting the DIO message. The node on receiving the DIO calculate its rank according to the objective function. Now objective function(OF) defines the set of rules which tells a node how to translate the metrics and constraints to evaluate the rank and select a parent with lowest rank and add the source node in the parent list and then further multicast it. Here Rank is a value that gives the relative position of a node to the root whose value monotically increases as we go deep in the tree. The rank of a node should be greater than the rank of all the parent nodes.

The full processing of DIO message while DODAG construction and maintainance is explained in the figure 5

2.3 DODAG repair

DODAG repairment includes two mechanisms: local repair and global repair. Firstly, when a network inconsistency is detected it initiates local repairment then when the

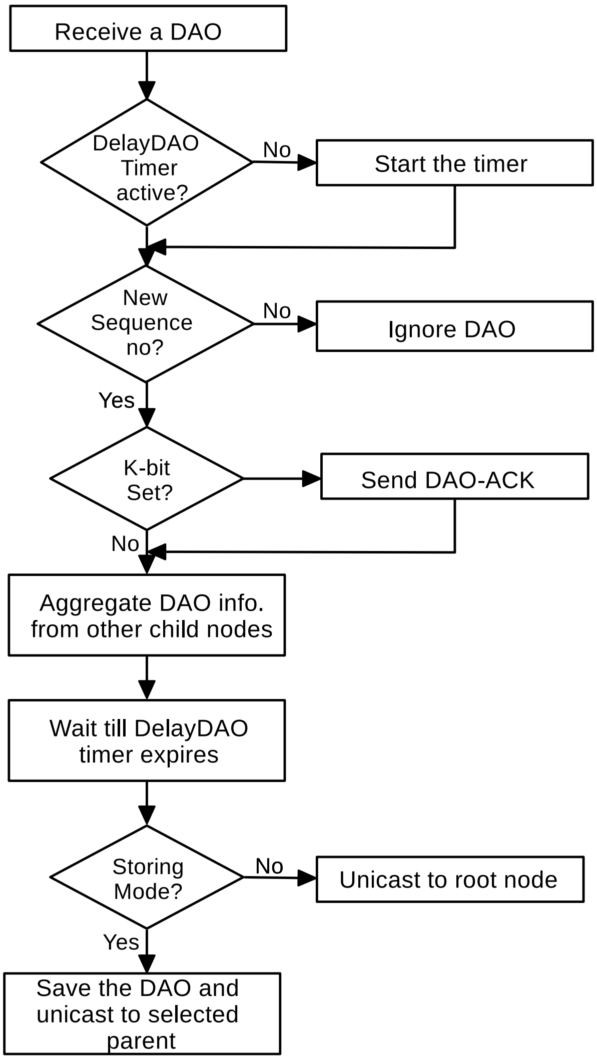


Figure 4: The DAO processing

latter is not able to control it, leading to multiple inconsistencies then global repairment takes place by simply sending DIOs again with incrementing the version number by one. Now, network inconsistency occurs when a loop is detected or the flags in the control message indicating the direction of the message flow doesn't match with the received direction(upward/downward). Loop detection and avoidance mechanisms are explained in [5]

3 Security in RPL

Though RPL meet all the criterias of LLN, it is still vulnerable to many security attacks. IETF ROLL has not proposed any specific security models for RPL.A security design weakness in the IETF RPL standard is the lack of specification of how the authentication and secure network connection among sensor devices running security critical missions

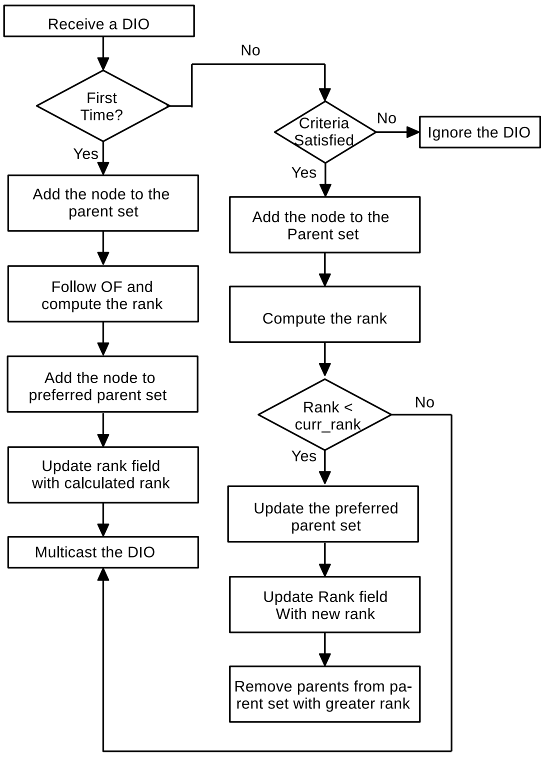
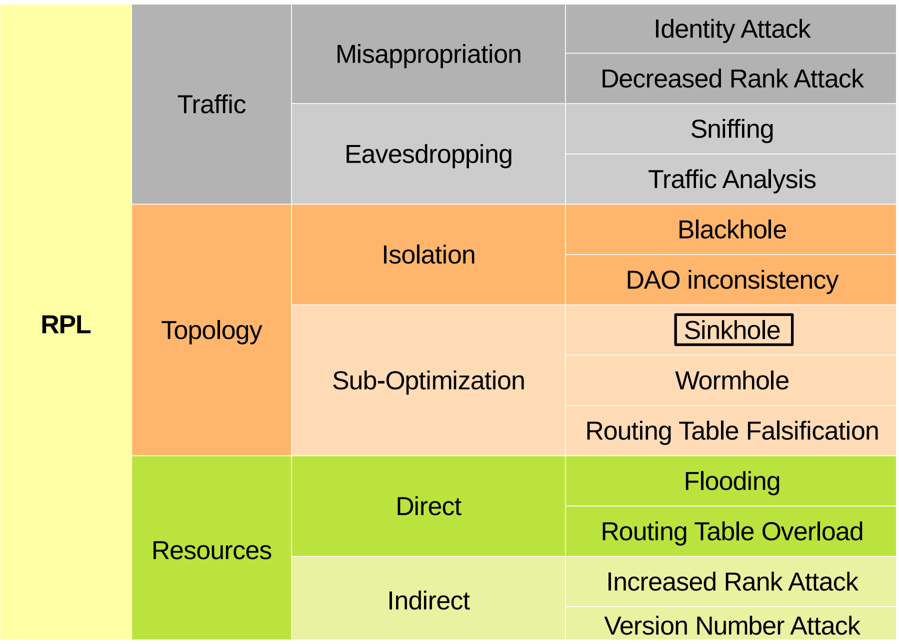

Receive a DIO
First Time?
No
Yes
Criteria Satisfied
No
Ignore the DIO
Yes
Add the node to the parent set
Add the node to the Parent set
Follow OF and compute the rank
Compute the rank
Add the node to preferred parent set
Rank < curr_rank
No
Yes
Update rank field with calculated rank
Update the preferred parent set
Multicast the DIO
Update Rank field With new rank
Remove parents from parent set with greater rank

Figure 5: DODAG maintainance

is defined. This exposes such devices to attacks such as Route Falsification, Byzantine, Rank, Routing Information Replay, Sybil, Selective Forwarding, Sinkhole, Blackhole, Greyhole and Version Number attacks. Based on the type of performing the attack, the attacks has been classified into various categories.The classification is shown in Figure 6. The attack encircled is the one that is discussed down in the next section.

4 Sinkhole Attack

An adversary launches the sinkhole attack[6] to hamper the working of the RPL by altering the control packets and advertises falsely about its position near to the sink node. As a result all the nodes within its reachability forward the message to it instead of sending it to the efficient route that is attracting the traffic towards itself. The adversary node that alters its rank is called the Sinkhole which creates a sphere of influence as shown in figure 7.



RPL	Traffic	Misappropriation	Identity Attack
			Decreased Rank Attack
		Eavesdropping	Sniffing
			Traffic Analysis
	Topology	Isolation	Blackhole
			DAO inconsistency
		Sub-Optimization	Sinkhole
			Wormhole
			Routing Table Falsification
	Resources	Direct	Flooding
			Routing Table Overload
		Indirect	Increased Rank Attack
			Version Number Attack

Figure 6: Attack Classification

5 Detection mechanisms of Sinkhole attack

5.1 Literary survey

In [7] says that the attacker can attack in two ways either by increasing the rank hence increasing the parent set size or by decreasing the rank which is more effective. The proposed mechanism can prevent an illegitimate node from sending DIO with increased version number and establishing lower rank. The mechanism calculates one-way version number hash chain and a rank hash chain for each node with version, V_i. DODAG root then send root of the version number hash chain, MAC(Max rank hash value associated with the next version number) using the next value of version number hash chain as the key and with its signature on it. Receiving the DIO, the next node verifies the authentication data and saves the values if success. So when their is a rank change in the DIO, with their revealed version-number hash value they can decode the saved Max Rank Hash value and then validate the new rank by performing $hash^{l-newrank}$ on new rank. If equal to the saved Max Rank Hash value then update the receiving node's rank according to the sender.

In [8] each node maintains a parent set with their rank, so when a new updated rank request come to a node 'i':-

1. The node 'i' calculates the average(R_{avg}) of the rank of the nodes in the parent set(including the new requested value) and maximum rank(R_{max})
2. Then threshold(i)= R_{avg}-($R_{max} * K$) and K is constant($0 < K < 1$) whose value is heuristically assumed.

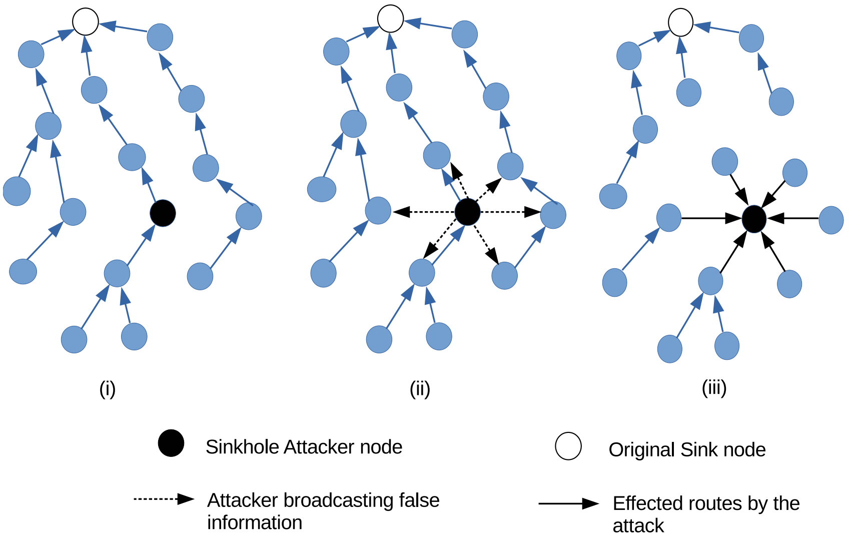

(i)
(ii)
(iii)
● Sinkhole Attacker node
○ Original Sink node
Attacker broadcasting false information
Effected routes by the attack

Figure 7: Sinkhole attack

3. If node(i)'s neighbor node's rank is lower than threshold it judges the neighbor node as an attacking node and excludes it from parent candidate list.This way it prevents the sinkhole node from being selected as the parent node.

In [9] Weekly and Pister employed two techniques- rank verification and parent-fail over. In rank verification one way hash function is computed starting with a random number picked by the root, which is hashed and forwarded to the next level and then the receivers perform hashing on the received value before forwarding it to the next and so on. So what the adversary node can at max do is forward the received value without hashing it as it can never know the previous hashed value received by its parent. Here each node stores the hashed value received from its preferred parent node. After the tree is converged, waiting for some reasonable time the root node securely broadcasts the random number picked by it at the starting. On receiving this initial value, the receiver performs hashing on this value for the new rank(p) number of times. If its unequal with the already saved value then it is assumed that the parent is falsifying its rank otherwise continue with normal process. And in the second technique, a UNS field is added to the DIO message. A threshold value is decided which is the amount of messages received by the root node from each node in a decided period of time. If the percentage is lower than that, the node is added to the UNS list. If the node on receiving next DIO finds itself included in the UNS, it will add its parent to a local blacklist which will prevent the latter from being selected as a parent.

It is mentioned in [10] that the two most preferred IDS approaches in RPL-based network are anamoly-based IDS and specification-based IDS. The authors in [10] have used specification-based IDS in their detection mechanism. Firstly, they simulate RPL in Contiki-Cooja simulator in ideal conditions to collect the trace files which are then passed through two designed algorithms to confine the whole process into different states with their flow of transitions. The IDS agents are placed into the network through hybrid method as the cluster-heads and in such a way that no cluster overlaps. After a time period, the cluster-head requests for topology information from all nodes(in the cluster) and cross-checks their information. While cross-checking, there is some time-synchronization issues which can mislead the detection so the authors proposed to add the sequence number information in the DIO and DIS messages to solve the issue. Then the IDS agent performs the detection algorithms on the collected information for validating DIO,DIS with their sequence numbers and checking rank consistency.

The mechanism in INTI[11] runs on four modules-

1. Cluster configuration where nodes are classified as members, associated and leaders which can change over time. The node with highest number of neighbor nodes is the leader and the rest are the members of the cluster. If a member is in the range of two leaders then that's the associated node.

2. Monitoring module where the "observer" node checks if the incoming and outgoing number of transmissions are equal if so, then the node is good or assumed to be malicious.

3. Detection module employs Beta probability density function to estimate the values of uncertainty, belief and disbelief and depending on them the confidence value is also calculated and updated periodically. This method uses the trust and reputation of a node to detect the sinkhole attack.

4. After the attacker is detected, in the Isolation module the attacker identity is disclosed in a restoration message for isolating the attacker and regaining the stability in communication.

The authors of [12] introduced three routing schemes- Far-sink reverse routing, equal-hop routing and left-hand routing rule. Each node sends probe packet to the sink after certain period of time. The intermediate nodes are to add their IDs in the packet and to forward it to the next hop so that when the sink node sends back the acknowledgement with its digital signature and timestamp on it can be routed back to the source node. On receiving acknowledgement, verifying the signature can assure the sink node being malicious node or not. If no acknowledgment is sent then the intermediate router can send acknowledgement with probable malicious node ID on it. The source node will then employ far-sink reverse routing till a possible max value 'h' and forward the probe packet to node with the max hop-count neighbor, then employ equal-hop routing with left hand routing rule till 'g' nodes and then continue with min-hop count routing again. If still no acknowledgement received then increase 'g' and 'h' values. If acknowledgement received with valid signature and timestamp then a packet is returned with the list of

compromised nodes back in the successful probe. On receiving it the sink isolate them from the network, till they are cleared, the bypass route can be used for routing to the real sink.
Finally, a brief comparison was provided in the 7

5.2 Problem statement

The problem is that sinkhole attack is a very effective attack to disturb the performance of an IoT network and when combined with other attacks like sybil attack, selective forwarding attack, etc. can majorly degrade the network performance so an efficient way needs to be developed for its prevention and detection.
With this as the objective the above mentioned work has been studied and a new method has been proposed as mentioned in Section 7

6 Objective

In relation to the above studied work, we are looking forward to do the following:

1. To study and analyse the sinkhole attack in RPL.
2. To model and design sinkhole attack for understanding its behaviour and characteristics.
3. To survey and analyse currently existing solutions for detecting and preventing sinkhole attack.
4. To come up with an improvement approach to an existing work.

7 Work done

Few previously developed mechanisms were analyzed and studied, and their advantages and drawbacks with regard to false positive rate and resource consumption were highlighted in the following table.

Table 1: Summary of mechanisms used to detect sinkhole attack on RPL

Title	**Merits**	**Demerits**
VERA [7]	Counters version attack also.	Computation overhead and resource exhaustion.
Secure parent [8]	Provides a way of selecting legitimate parent.	Does not prevent sinkhole attack fully.
Parent fail-over[9]	The combined mechanisms improve E2E performance of a network under attack	Network overhead and the sub-tree nodes to a sinkhole are added in the blocklist
Specification-Based IDS [10]	False positive rate is sufficient small and also have the ability to detect other topology attacks.	Due to IDS centralization high risk of system failure.
INTI[11]	INTI considers node mobil- -ity and almost all sinkhole nodes are detected.	Very less packet delivery rate.
PRDSA [12]	Provides an effective means to bypass the sinkhole attack	High network overhead, time taking and will have issues if sink-hole node and legitimate sink start almost at same time

8 Road Map

Time Duration	Work Plan
October 2019 - November 2019	Literature survey on Sinkhole attack and its prevention/detection mechanisms
December 2019	Sink attack and Rank Spoofing attack modeling
January 2020 - February 2020	Proposing improvised approach to an existing work
March 2020 - April 2020	Implement the proposed work via Cooja simulator
May 2020	Result analysis and report writing

9 The proposed work (if any done so far)

In this work, we propose a prevention mechanism to be used during rank update process. An IoT node not only maintains the neighbor set, parent set, and preferred parent set, but also the mean and standard deviation of each metric used in computation of rank of a node. The computation of mean and standard deviation of each metric is triggered each time a node receives a DIO packet from its neighbors. When the OF is used to compute the rank of a node according to a requesting parent, the newly computed rank is compared to its current rank. If the rank is lower than the current one, and probability of membership of each metric of the requesting parent is more than a predefined threshold, then the preferred parent is updated to the newer one. If the probability of membership for any metric is lower than the given threshold then the metrics are considered to be forged ones. In this case, it retains the original parent.

Metrics to be used: ETX, hop count, residual energy vs network time ratio.The latter is considered because if a new node joins the network then its residual energy will be high compared to its neighbours so if only residual energy will be considered then the new node might be assumed to be illegitimate therefore network time is considered along with it.

10 Future work

In relation to the above proposed work, we are looking forward to do the following:

i) To implement the proposed model using Contiki.

ii) Compare it with previous works.

iii) Writing the report.

References

[1] Mardiana binti Mohamad Noor and Wan Haslina Hassan. Current research on Internet of Things (IoT) security: A survey. *Computer Networks*, 148:283 – 294, 2019.

[2] V. Beltran and A. F. Skarmeta. Overview of Device Access Control in the IoT and its Challenges. *IEEE Communications Magazine*, 57(1):154–160, January 2019.

[3] A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur, and R. Alexander. Rpl: Ipv6 routing protocol for low-power and lossy networks. RFC 6550, March 2012.

[4] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko. Trickle algorithm. RFC 6206, March 2011.

[5] Patrick Olivier Kamgueu, Emmanuel Nataf, and Thomas Djotio Ndie. Survey on RPL enhancements: A focus on topology, security and mobility. *Computer Communications*, 120:10 – 21, February 2018.

[6] David Airehrour, Jairo A. Gutierrez, and Sayan Kumar Ray. SecTrust-RPL: A secure trust-aware RPL routing protocol for Internet of Things. *Future Generation Computer Systems*, 93:860 – 876, 2019.

[7] A. Dvir, T. Holczer, and L. Buttyan. Vera - version number and rank authentication in rpl. In *2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems*, pages 709–714, Oct 2011.

[8] I. Kenji, T. Matsunaga, K. Toyoda, and I. Sasase. Secure parent node selection scheme in route construction to exclude attacking nodes from RPL network. *IEICE Communications Express*, 4(11):340 – 345, 2015.

[9] K. Weekly and K. Pister. Evaluating sinkhole defense techniques in rpl networks. In *2012 20th IEEE International Conference on Network Protocols (ICNP)*, pages 1–6, October 2012.

[10] A. Le, J. Loo, A. Lasebae, A. Vinel, Y. Chen, and M. Chai. The impact of rank attack on network topology of routing protocol for low-power and lossy networks. *IEEE Sensors Journal*, 13(10):3685–3692, October 2013.

[11] C. Cervantes, D. Poplade, M. Nogueira, and A. Santos. Detection of sinkhole attacks for supporting secure routing on 6lowpan for internet of things. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 606–611, May 2015.

[12] Y. Liu, M. Ma, X. Liu, N. N. Xiong, A. Liu, and Y. Zhu. Design and analysis of probing route to defense sink-hole attacks for internet of things security. *IEEE Transactions on Network Science and Engineering*, 7(1):356–372, Jan 2020.