People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
Kasdi Merbah Ouargla University
Faculty of new information technologies and communication
Department of Computer Science and Information Technology

Professional MASTER Thesis

Field: Computer Science and information technology

Branch: Computer Science

Specialty: Administration and Network Security

Presented by: Kouchy Djihad, Ammari Ferdous

Theme

Detection of DIO suppression attacks for IOTs Networks

Committee Members:

-	Dr. Boukhamla Akram	**Supervisor**	**UKM Ouargla**
-	Dr.Djedia.H	**Examiner**	**UKM Ouargla**
-	Dr.Khaldi.A	**President**	**UKM Ouargla**

Year University: 2021/2022

Dedication

dedicate this work especially:

To my beloved parents, my mother Habiba and my father Azzeddine who have always encouraged, supported and guided me to work hard to succeed in my studying and life, shared their words of moral has been my source of inspiration when I thought of giving up. your prayers are the reason behind making me able to get such honour.

To my sisters Anfal and Meriam for their support and positive feedback.

To my brothers, who stood by my side until the last minute.

To my supportive partner Djihad, I could say; thanks for this wonderful coincidence that made me know you more closely.

It was great working on this thesis with you.

To my dearest friends Meriem and Sarah and to my lovely cousin Marwa for their supports.

To my friends Yasmin, Aicha, Monaim and Zohire.

To all my university friends and colleagues to all whom encouraged me during my dissertation.

Ferdous

Dedication

I would like to dedicate this dissertation:

To my beloved parents, my mother Nora and my father Saber

, whom always been there for me I would never forget their

prayers, love and support I could never reach this moment without

them.

To my dear sibllings , Mahammed, Adem

,of course my nephew Rayan and specialy my sweetheart sister Rihab ,who support me and every day with me

May Allah bless them for me.

To my lovely colleague, Ferdous I would never regret a

moment by choosing you to be my partner in our short but amazing

journey.

To my dearest friends, Monaim ,Yasmin, Aicha , Zohire , Taha , Sonia and Nor the ones

whom always been there and prayed for me.

To all my university friends and colleagues.

To my neighbors Hakima, Wissam.

Last and not least to all whom encouraged me during my dissertation

preparation.

Djihad

Acknowledgment

First and foremost, we thank God who gave us the strength and will to accomplish this work.

We would like to extend our sincere thanks to our supervisor

Mr. BOUKHAMLA Akram who helped us during our work with his patience, precious advice, guidance and all these constructive observations for the smooth running of our project.

We would like to thank all the faculty members in the Department of Computer Science for the training they provided us throughout our academic career at the university

We'd also like to extend our gratitude to our loving parents, family and close friends for their support and assistance by providing comfort conditions.

Abstract

The Internet of Things has become the revolution of the era, as it makes the things around us smart by giving them communication with each other. The Internet of things is the connection of an unlimited number of devices to the Internet, where they share a huge number of data through sensors, which help in data delivery, analysis and Getting to the perfect solution. Internet of things applications are increasingly used in many areas and systems, and with this wide increase in their uses and methods of employment, new problems have begun to appear in these networks, among the problems is the problem of faster energy drain, due to the increase in data traffic in the network. Several protocols have been proposed as a solution to fix this problem and adapt to these networks. One of the most important protocols that has been introduced is the RPL protocol, which is the de-facto standard for routing in low-power and lossy networks (LLNs). However, RPL is vulnerable to a number of attacks related to cross-control messages that are a top priority to improve the security of future IoT systems. In this thesis, we propose an algorithm capable of detecting a DIO suppression attack and identifying malicious nodes to improve the security of the RPL routing protocol. The algorithm was implemented using the cooja simulator. After studying a number of scenarios that depended on time and also malicious nodes, we can say that we reached a satisfactory result, through which we can identify all malicious nodes in the topology. On the other hand, there are some minor errors that lie in considering a percentage that does not exceed 10 % of harmful nodes, which can be improved in the future.

Keywords: IoT, RPL,LLN, DIO suppression attack, Cooja.

Résumé

L'Internet des objets est devenu la révolution de notre époque, car il rend les objets qui nous entourent intelligents en leur permettant de communiquer entre eux. L'Internet des objets est la connexion d'un nombre illimité d'appareils à Internet, où ils partagent un énorme nombre de données via des capteurs, qui aident à la livraison et à l'analyse des données, et à la solution parfaite. Les applications de l'Internet des objets sont de plus en plus utilisées dans de nombreux domaines et systèmes, et avec cette forte augmentation de leurs utilisations et méthodes d'emploi, de nouveaux problèmes ont commencé à apparaître dans ces réseaux,

parmi les problèmes figure le problème de la consommation d'énergie plus rapide, en raison de la augmentation du trafic de données dans le réseau Plusieurs protocoles ont été proposés comme solution pour résoudre ce problème et s'adapter à ces réseaux. L'un des protocoles les plus importants qui a été introduit est le protocole RPL, qui est la norme de facto pour le routage dans les réseaux à faible puissance et avec perte (LLN).Cependant, RPL est vulnérable à un certain nombre d'attaques liées au contrôle croisé. messages prioritaires pour améliorer la sécurité des futurs systèmes IoT. Dans cette thèse, nous proposons un algorithme capable de détecter une attaque de suppression DIO et d'identifier les nœuds malveillants pour améliorer la sécurité du protocole de routage RPL. L'algorithme a été implémenté à l'aide du simulateur cooja. Après avoir étudié un certain nombre de scénarios qui dépendaient du temps et aussi des nœuds malveillants, nous pouvons dire que nous sommes arrivés à un résultat satisfaisant, grâce auquel nous pouvons identifier tous les nœuds malveillants de la topologie.D'autre part, il y a quelques erreurs mineures qui se trouvent dans considérant un pourcentage qui ne dépasse pas 10 % de nœuds nuisibles, qui peuvent être améliorés à l'avenir.

Mots clés: IoT, RPL, LLN, attaque de suppression DIO, Cooja.

ملخص

اصبحت انترنت الاشياء ثورة العصر فهي تجعل الأشياء من حولنا ذكية من خلال منحها التواصل مع بعضها البعض .انترنت الأشياء هي اتصال عدد غير محدود من الأجهزة بشبكة الانترنت ,حيث تتشارك بعدد هائل من البيانات من خلال أجهزة الاستشعار , التي تساعد في تسليم البيانات و تحليلها و الوصول الى الحل بشكل مثالي . يتم استخدام تطبيقات انترنت الأشياء في العديد من المجالات و الأنظمة بشكل متزايد, و مع هذا التزايد الواسع لاستخداماتها و طرق توظيفها بدأت تظهر مشاكل جديدة في هذه الشبكات , من بين المشاكل مشكلة استنزاف الطاقة بشكل اسرع, و ذلك بسبب زيادة حركة البيانات في الشبكة .تم طرح العديد من البروتوكولات كحل لإصلاح هذه المشكلة و التكيف مع هذه الشبكات . من اهم البروتوكولات التي تم طرحها هو بروتوكول RPL حيث هو المعيار الواقعي للتوجيه في الشبكات ذات الطاقة المنخفضة و المفقودة LLNs. مع ذلك فان RPL عرضة لعدد من الهجومات المتعلقة برسائل التحكم المتبادلة التي تعتبر أولوية قصوة لتحسين امن أنظمة انترنت الأشياء IoT المستقبلية . في هذه الاطروحة , نقترح خوارزمية قادرة على كشف هجوم قمع DIO و تحديد العقد الخبيثة لتحسين أمن بروتوكول التوجيه RPL تم تنفيذ الخوارزمية باستخدام محاكي cooja. بعد دراسات عدد من السيناريوهات التي اعتمدت على الوقت و ايضا العقد الخبيثة ,نستطيع القول ان توصلنا الى نتيجة مرضية ,التي من خلالها نستطيع التعرف على كل العقد الخبيثة في الطبولوجيا.و من جهة اخرى يتواجد بعض الاخطاء الطفيفة التي تكمن في اعتبار نسبة لا تتجاوز 10% من العقد الضارة ,التي يمكن التحسين فيها مستقبلا .

الكلمات المفتاحية : RPL ,LLNs , DIO suppression attack,IoT, Cooja .

I. Contents

I. Chapter 1 Internet of things 4
I.1 Introduction 4
I.2 Definition 4
I.3 Architectures 5
I.4 IOT Element 6
I.4.1 Identification 7
I.4.2 Sensing 7
I.4.3 Communication 7
I.4.4 Computation 7
I.4.5 Services 8
I.4.6 Semantics 8
I.5 Applications 9
I.6 protocols of IoT 12
I.7 IOT network 13
I.7.1 Low-power and lossy network (LLN) 13
I.7.2 Wireless sensor networks (WSN) 14
I.7.3 Wireless mesh networks (WMN) 15
I.8 The advantages and disadvantages of Internet Of Things (IoT) 15
I.8.1 Advantages 15
I.8.2 Disadvantages 16
I.9 Conclusion 17
II Chapter 2 The RPL protocol 19
II.1 Introduction 19
II.2 RPL Protocol 19
II.3 RPL control messages 19
II.3.1 DIO (DODAG Information Object) 20
II.3.2 DIS (DODAG Information Solicitation) 20
II.3.3 DAO (Destination Advertisement Object) 20
II.3.4 DAO-ACK (Destination Advertisement Object Acknowledgement) 21
II.4 RPL Rank 21
II.5 Construction of DODAG 21
II.6 The operating modes of the RPL protocol 22

II.6.1 Mode non storing 22
II.6.2 Mode storing 23
II.7 Traffic Flows Supported by RPL 23
II.7.1 Multipoint-to-Point (MP2P) 23
II.7.2 Point-to-Multipoint (P2PM) 23
II.7.3 Point-to-Point (P2P) 24
II.8 Objective Function 24
II.9 Trickle Timer 25
II.9.1 Parameters and Variables [44] 25
II.9.2 Algorithm Description [44] 25
II.10 RPL security 26
II.10.1 Self-healing mechanisms 26
II.10.2 Security features 27
II.11 ATTACKS ON RPL TOPOLOGY 28
II.11.1 Sub-optimization Attacks 28
II.11.2 Isolation Attacks 30
II.12 Conclusion 31
III Chapter3 Conception and Implementation 33
III.1 Introduction 33
III.2 Conception 33
III.2.1 Model DIO suppression attack 33
III.2.2 trickel timer 34
III.2.3 Model Of Solution 34
III.2.4 Pseudo code of solution 36
III.3 Implementation 36
III.3.1 Contiki OS 36
III.3.2 Cooja Simulation 37
III.3.3 Performance evaluation 38
III.4 Conclusion 44

List of figures

Figure I.1 IoT architecture: (a) Three-layer. (b) Four-layer. (c) Five-layer. 5
Figure I.2 IoT elements[11]..... 7
Figure I.3 Top 10 IoT segments in 2018 [17] 10
Figure I.4 Applications of Internet of Things..... 11
Figure I.5 Advantages and Disadvantages of IoT 17
Figure II.1. Flow of RPL control messages 20
Figure II.2 RPL topology based on ranks[40]..... 22
Figure II.3 Storing mode and non-storing mode[42] 23
Figure II..4 Traffic patterns supported by RPL. Lines with arrows indicate the traffic flow, while dotted lines without arrows indicate the links of the routing topology.[42]..... 24
Figure II.5 Trickle algorithm during two intervals illustrating the transmission, listen-only period and the suppression mechanism 26
Figure II.6 Taxonomy of attacks on RPL topology 28
Figure II.7 Illustration of DIO Suppression Attack[50]..... 29
Figure II.8 Illustration of WormHole Attack[52]..... 29
Figure II.9 Illustration of SinkHole Attack[53] 30
Figure II.10 Illustration of BlackHole Attack[55] 31
Figure III.2 Pseudo code of solution..... 36
Figure III.3 Cooja simulator interface 37
Figure III.4 Network topology and node placement for simulation..... 39
Figure III5. Simulation results of the proposed method: (a) TPR for different simulation runtime (b) FPR for different simulation runtime (c) TPR for different number of malicious nodes (d) FPR for different number of malicious nodes..... 41
Figure III.6 Detection accuracy 42
Figure III7. Energy consumption..... 43

Tables list

Table 1 The elements and key technologies of IoT. 9
Table2 IOT PROTOCOLS 12
Table 3 Evaluation parameters 40
Table 4 Parameters used for three scenarios..... 40
Table 5 Parameters used for three scenarios..... 43

List of Abbreviations

6LoWPAN	ipv6 Low power Wireless Personal Area Network
DAO	Destination advertisement Object
DAO-ACK	Destination advertisement Object Acknowledgement
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
DODAG	Destination Oriented Directed Acyclic Graph
ETX	Expected Transmission Count
IEEE	Institute of Electrical and Electronics Engineers
Imax	Interval Maximum
IETF	Internet Engineering Task Force
Imin	Interval Minimum
IoT	Internet Of Things
IP	Internet Protocol
IPv6	Internet Protocol Version 6
K	redundancy constant
LLN	Low-Power and Lossy Network
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
OF	Objective Function
OF0	Objective Function Zero
RFC	Request For Comments
ROLL	Routing Over Low-power and Lossy
RPL	Routing Protocol for Low-Power
WSN	Wireless Sensor Networks
WMN	Wireless Mesh Network
QOS	Quality of service

General Introduction

The Internet is one of the most important inventions in human history. The Internet of Things represents the next development of the Internet. It is the new global technological directive that has become the integrated solution to meet the daily and growing needs of people, by relying on the Internet without resorting to human intervention. The Internet of Things consists in linking devices of different types so that they can all communicate and exchange information among themselves, using multiple types of communication protocols. Where each device or element of the Internet is assigned a unique Internet Protocol (IP) address. The Internet of things contributes greatly By providing many applications for various sectors such as smart cities, smart buildings, smart networks and healthcare that improve daily life and also improve the distribution of global resources. However, there are several obstacles that threaten to slow down the development of the Internet of Things, where the correct operation of the network is linked The Internet of Things strongly supports the correct operation of the communication protocol. In the Internet of Things, routing of information in a network can only be ensured thanks to routing protocols that include reliable data delivery. To this end, several protocols have been proposed to perform routing in WSNs. Among these protocols is RPL, the routing protocol for Low Power and Lossy Networks (LLNs). Recent standardization efforts are enhancing the role of RPL as the standard routing protocol for IPV6-based wireless sensor networks. However, many attacks have been observed while routing data packets between devices by malicious nodes. In this document , we introduce a DIO suppression attack, which is a new degradation attack. Service against RPL and greatly reduce its routing service. Unlike other RPL attacks in the literature, a DIO suppression attack does not require stealing cryptographic keys from some legitimate nodes, it is enough to periodically return messages that were previously heard. The goal of a DIO suppression attack is to interrupt Or slow down the transmission of DIO messages in the network. To achieve this goal, the DIO suppression mechanism of the Trickle algorithm is exploited. In this attack, the victim nodes are instigated to suppress the transmission of DIO messages, which are the messages necessary to build the routing topology. This causes some nodes to remain hidden

It also leads to a general deterioration in the quality of the paths, or in the worst case, the splitting of the network.

Our main goal of this graduation project is to detect a DIO suppression attack, to improve the performance of the RPL protocol.

This thesis is organized into three chapters :

The first chapter will be devoted to introducing the Internet of Things, as well as introducing some basic concepts about IOT.

In the second chapter, we introduce the RPL (IPv6 Routing Protocol for Low Power and Lossless Networks) communication protocol that is most widely used in the IoT domain.

And in the last chapter, we describe the experimental framework used to evaluate the proposed method and show the results of the evaluation in the Cooja simulator.

Chapter 1 :

Internet Of Things

I. Chapter 1 Internet of things

I.1 Introduction

Over the past few years, the IoT has gained significant attention since it brings potentially tremendous benefits to the human. The concept of the IoT has been introduced by Kevin Ashton in 1999, it aims to connect anything at anytime in anyplace [1]. ”Things” in IoT are embedded with sensing, processing and actuating capabilities and cooperate with each other to provide smart and innovative services autonomously. The IoT spans many diverse application domains such as home automation, environmental monitoring, healthcare, and so on [2]. The primary objective of the IoT is unification of these numerous diverse application domains under the same umbrella referred as smart life [2]. The architecture of IoT supports a large number of heterogenous devices and integrates various communication technologies that enable the connectivity of IoT devices to provide the required services to end-users. The present chapter provides an overview of fundamental concepts of IoT. It introduces the IoT definition, potential applications and architecture including major elements and protocols used in IoT.

I.2 Definition

The Internet of Things (IoT) refers to the evolutionary stage of the internet, which makes a global communicating infrastructure between humans and machines.[3]The Internet of Things (IoT) describes the network of physical objects—“things”—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet.[4]

Internet of Things is a concept and a paradigm that considers pervasive presence in the environment of a variety of things that through wireless and wired connection and unique addressing schemes are able to interact with each other and cooperate with other things to create new applications and reach common goals [5].

The goal of the internet of things is to enable things to be connected every time, everywhere, with anything and anyone ideally using any path/network and any service.

I.3 Architectures

There is no single consensus on architecture for IoT, which is agreed universally. Different architectures have been proposed by different researchers.

Due to the different understanding and definition of IoT, correspondingly, the IoT architecture has been proposed in many different types (shown in Figure I.1). One famous mode is three-layer architecture [6], including Application, Network and Perception layers. In [7], a four-layer reference mode was proposed, consisting of Application layer, Service support and Application support layer, Network layer and Device layer. Another architecture I want to mentioned here is composed of five layers [8]: a Business layer, an Application layer, a Processing layer, a Transport layer, and a Perception layer.

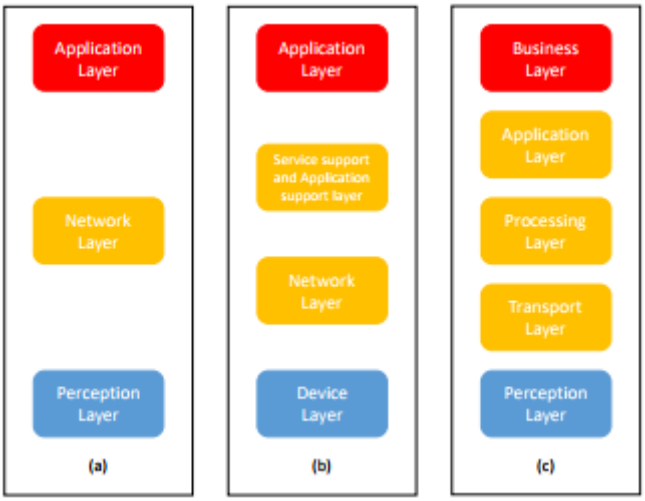


Figure I.2 IoT architecture: (a) Three-layer. (b) Four-layer. (c) Five-layer.

Three- and Five-Layer Architectures

- The perception layer is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment.
- The network layer is responsible for connecting to other smart things, network devices, and servers. Its features are also used for transmitting and processing sensor data.

- The application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health. The three-layer architecture defines the main idea of the Internet of Things, but it is not sufficient for research on IoT because research often focuses on finer aspects of the Internet of Things. That is why, we have many more layered architectures proposed in the literature. One is the five-layer architecture, which additionally includes the processing and business layers [9–10]. The five layers are perception, transport, processing, application, and business layers (see Figure 1). The role of the perception and application layers is the same as the architecture with three layers. We outline the function of the remaining three layers.
    - The transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as wireless, 3G, LAN, Bluetooth, RFID, and NFC.
    - The processing layer is also known as the middleware layer. It stores, analyzes, and processes huge amounts of data that comes from the transport layer. It can manage and provide a diverse set of services to the lower layers. It employs many technologies such as databases, cloud computing, and big data processing modules.
    - The business layer manages the whole IoT system, including applications, business and profit models, and users' privacy. The business layer is out of the scope of this paper. Hence, we do not discuss it further.

I.4 IOT Element

Internet of things offers numerous advantages and services to the users. Therefore, to use them correctly, some elements are needed. The IoT elements will be discussed in this section. Figure I.3 shows the elements required to provide IoT functionalities.

Figure I.4 IoT elements[11]

I.4.1 Identification

The identification is essential for the development of the IoT and is important to ensure the correct identification of objects in order to match services with their demand. Actually, many identification methods exist such as electronic product codes (EPC) and ubiquitous code (uCode). Object identification refers to its name or designation and addressing refers to its IP address for communication on the network. Addressing methods include today's IPv4, IPv6, and 6LoWPAN that provide compression on IPv6 headers [12]. With the large address space provided by IPv6, all the addressing needs of the IoT are thought to be taken care of.

I.4.2 Sensing

In IoT, sensing refers to acquiring data from the environment and sending it to a database, remote, local, or in a cloud, and as an example of IoT sensors, one can find smart sensors, actuators, or wearable sensors.

I.4.3 Communication

To achieve smart services, IoT communication techniques communicate heterogeneous artifacts. One of the main goals of Internet of things is Communication in which various devices connect and communicate with each other. In the communication layer, devices can transfer and deliver messages, documents and other in-formation. There are many methods which facilitate communication, for example Bluetooth , radio frequency identification (RFID, long term evolution (LTE), Wi-Fi and nearfield communication (NFC)

I.4.4 Computation

In the computation step, the information collected from various objects in IoT applications must go through a processing procedure. The information is filtered to specify the useful

information and remove unnecessary one. Several hardware and software platforms are involved in performing this task, such as Arduino, Raspberry Pi, Friendly ARM, Intel Galileo, Beagle Bone, WiSense, Mule, etc., and software like Tiny OS, Lite OS, Android, etc. These

operating systems play an important role in processing, and also offer lightweight OS that is suitable for designing IoT environments. Furthermore, the cloud is considered an important computational part of the IoT. That it provides facilities for smart objects to send their data to the cloud, allows big data to be processed in real-time, and enables end-users to benefit from the knowledge extracted from the collected big data [13] [14].

I.4.5 Services

IoT applications can mainly provide four types of services; Identity-related services, Information aggregation services, Collaborative Aware services, and Ubiquitous services. The Identity-related services are concerned with object identity information, whereas Information aggregation is used to collect, summarize, and process all the information from objects and send it back to the application. Moreover, Collaborative Aware services are used to turn the collected information into a decision and send appropriate responses to the devices. The last service is Ubiquitous services, which are responsible for providing Collaborative-Aware services immediately to anyone at any time and place [14] [15] [16].

I.4.6 Semantics

It is the responsibility of IoT to facilitate users by performing their tasks. It is the most important element of IoT to fulfill its responsibilities. It acts as the brain of IoT. It gets all information and makes appropriate decisions to send responses to the devices.

Table 1 is shown to describe the key technologies involved in each element of IoT.

Table 1 The elements and key technologies of IoT.

IoT Elementes	Tecnologies
Identification Naming Addressing	Electronic ,Prodect Code ,Ucode Ipv4 ,and Ipv6
Sensing	Smart,Sensor,RFID tags,Wearable Sensing Devices and Actuators
Communication	Radio Frequency Identification ,wirless Sensor Network ,Near Field Communication (NFC), Bluetooth, Long Term Evolution (LTE)
Compulation Hardware Software	Audrino,Raspherry Pi,Intel Galil Operating System
Services	Identity-Related, Information Aggregation ,Collaborative-Aware and Ubiquitous
Semantics	RDF ,OWL,EXI

I.5 Applications

The Internet of Things has many application domains, and due to its rapid development, the number of domains is increasing. Figure I.5 [17] shows the top 10 IoT segments in 2018.

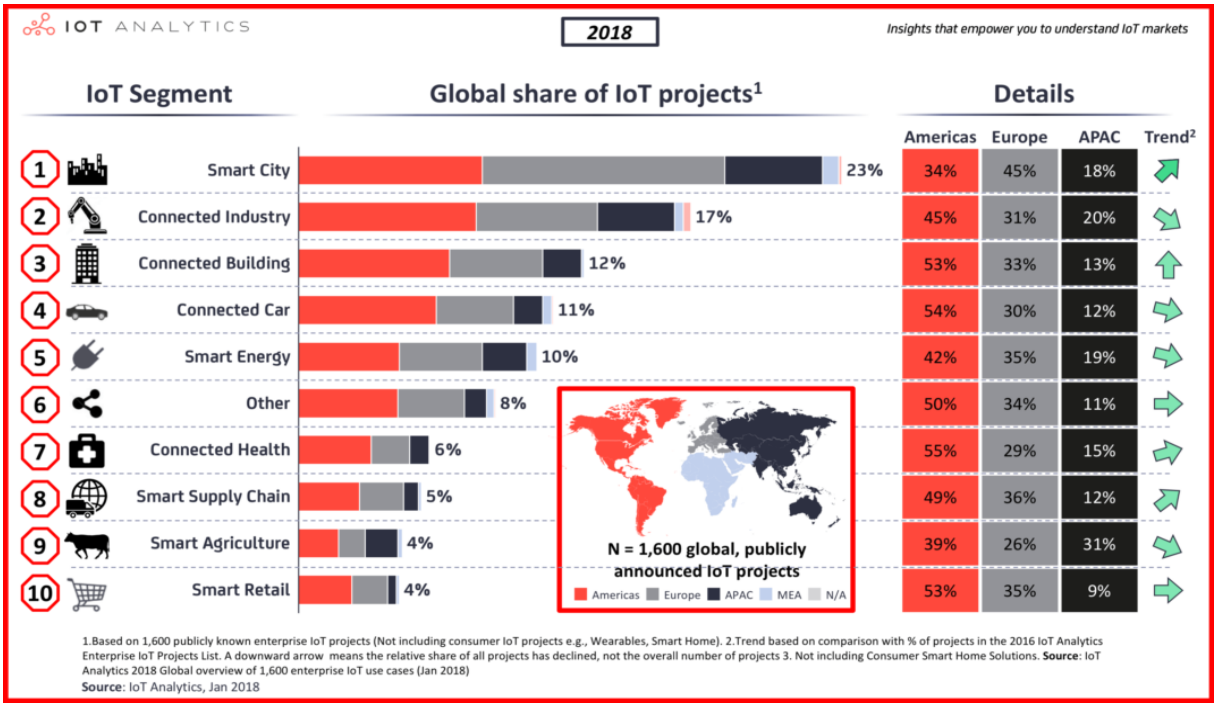


Figure I.6 Top 10 IoT segments in 2018 [17]

The use of applications of IoT in every field is increasing all over the world. The major IoT applications driving countries are China, North America, and Western Europe[6]. IoT industry can grow the revenue from $892 billion in 2018 to $4 trillion by 2025 [18].

The applications of the Internet of Things have infinite areas such as fitness, home automation, transport management, education, entertainment, health care, environmental monitoring, transport systems, and energy conservation. These applications make human life easy as shown in Fig. 4. A smart parking application has been designed for smart cities where dynamic resource caching technique has been implemented using CoAP protocol. This technique reduce energy consumption of servers [19]. One more method is presented by Santos to reduce the resource consumption in smart cities applications with the use of Integer Linear Programming [20]. Smart industry applications reduce human-efforts and security is a big challenge in smart industry applications. So, a model is presented to transmit, store and

record the information used in industry applications which provides secure, structured and efficient data communication. This model is integrated of order preference and additive weight techniques [21]. IoT plays a major role to improve the health services. A hybrid model is formed based on cloud computing and IoT to improve the CPU utilization, optimization,

storage and retrieval of data. This model improves the real time data retrieval around 5.2 % [22]. Due to smart agriculture IoT applications there is rapid growth in farm yields. A life cycle is presented for agriyields for plantation in controlled environment, open and closed – field farming and breeding etc. This life cycle is formed of green IoT system which has higher yield growth expectancy [23]. Smart traffic management helps to manage, monitor, and tracking the traffic in a well manner using sensors and detection technology. It makes easy to pass emergency vehicles. One smart traffic management system has been presented by integrating IoT with the agent technology. This smart traffic information system is flexible, scalable and of low cost [24]. Smart homes make our comfort life style in all the ways like security using cameras, kitchen appliances, entertainments and other households. One approach using IoT and Big Data analytics approach deals with smart home management system [25].

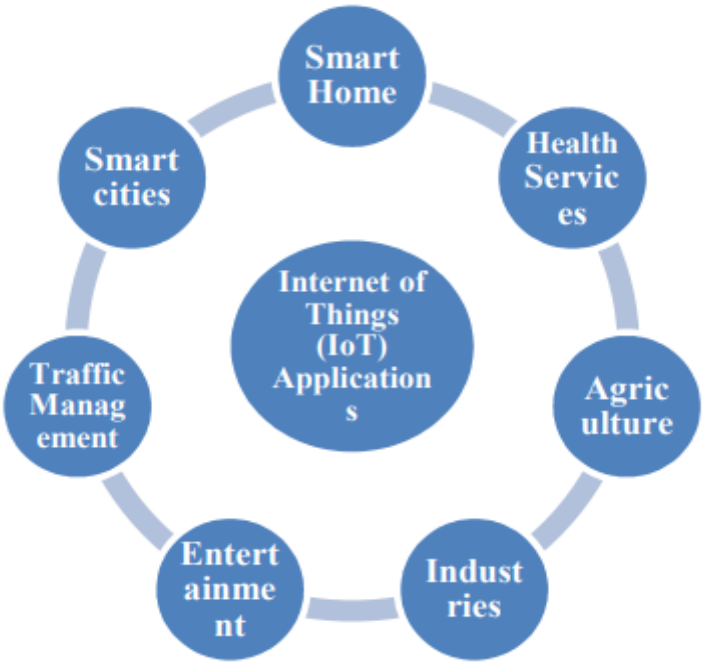


Figure I.7 Applications of Internet of Things

I.6 protocols of IoT

No single, alternative protocols have been developed by Internet Engineering Task Force (IETF) for communication between IoT devices [26]. Internet Protocol for Smart Object (IPSO) Alliance has published various papers for explaining the communication protocols and standards [27], [28] and [29]. These protocols as per the layers are shown in TABLE 2

Table2 IOT PROTOCOLS

Layers	Protocols				Services
Physical Layer	IEEE 802.15.14				Infrastructure
Adaption Layer	6LOWPAN		IPV6		Device Identification
Network Layer	RPL				Routing Services
Transport Layer	TCP		UDP	DTLS	End to End Communication Security Services
Application Layer	HTTP	AMQP	COAP	MQTT	Data Protocols

Internet layer technologies (OSI Layer 3) identify and route packets of data. Technologies commonly adopted for IoT are related to this layer and include IPv6, 6LoWPAN, and RPL.

- **IPv6**

At the Internet layer, devices are identified by IP addresses. IPv6 is typically used for IoT applications over legacy IPv4 addressing. IPv4 is limited to 32-bit addresses, which only provide around 4.3 billion addresses in total, which is less than the current number of IoT devices that are connected, while IPv6 uses 128 bits, and so provides 2 128 addresses. In practice, not all IoT devices need public addresses. Of the tens of billions of devices expected to connect via the IoT over the next few years, many will be deployed in private networks that

use private address ranges and only communicate out to other devices or services on external networks by using gateways.

- **6LoWPAN**

The IPv6 Low Power Wireless Personal Area Network (6LoWPAN) standard allows IPv6 to be used over 802.15.4 wireless networks. 6LoWPAN is often used for wireless sensor networks, and the Thread protocol for home automation devices also runs over 6LoWPAN.

- **RPL**

The Internet Layer also covers routing. IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) is designed for routing IPv6 traffic over low-power networks like those networks implemented over 6LoWPAN. RPL (pronounced “ripple”) is designed for routing packets within constrained networks such as wireless sensor networks, where not all devices are reachable at all times and there are high or unpredictable amounts of packet loss. RPL can compute the optimal path by building up a graph of the nodes in the network based on dynamic metrics and constraints like minimizing energy consumption or latency

I.7 IOT network

I.7.1 Low-power and lossy network (LLN)

Low-Power and Lossy Network. Typically composed of many embedded devices with limited power, memory, and processing resources interconnected by a variety of links, such as IEEE 802.15.4 or low-power Wi-Fi. There is a wide scope of application areas for LLNs, including bindustrial monitoring, building automation (heating, ventilation, and air conditioning (HVAC), lighting, access control, fire), connected home, health care, environmental monitoring, urban sensor networks, energy management, assets tracking, and refrigeration.

RFC 7228 further says, Low-Power and Lossy Network often exhibit considerable loss at the Physical Layer, with significant variability of the delivery rate, and some short-term unreliability, coupled with some medium-term stability that makes it worthwhile to both construct directed acyclic graphs that are medium-term stable for routing and do

measurements on the edges such as Expected Transmission Count (ETX) RFC 6551. Not all LLNs comprise low-power nodes RPL-DEPLOYMENT.

Low-Power and Lossy Networks typically are composed of Constrained Nodes; this leads to the design of operation modes such as the "non-storing mode" defined by RPL (the IPv6 Routing Protocol for Low-Power and Lossy Networks RFC 6550). So, in the terminology of the present document, an LLN is a Constrained Node Constrained Network with certain network characteristics, which include constraints on the network as well.

I.7.2 Wireless sensor networks (WSN)

Wireless sensor network (WSN) is one of the rapidly growing sections in networking today [30]. It is a self-configured wireless network consisting of independent devices which are spatially distributed [31]. Formally, a WSN can be defined simply as a network of low-size and low complexity devices known as nodes that use wireless communication link to sense the environment and send the data gathered from the monitored environment to control unit for further processing and decisions. Moreover, the collected data is forwarded via multiple hop relaying, to a sink or base station, which is considered as an interface between users and the network, the sink can use the data locally, or is connected to other networks (e.g., the Internet) via a gateway [32] [33].

Typically a wireless sensor network contains thousands of sensor nodes, which can communicate among themselves using radio signals, so it is equipped with sensing and computing devices, radio transceivers and power components. All the nodes in a wireless sensor network are resource constrained: they have limited storage capacity, processing speed, and communication bandwidth [32].

WSN usually uses sensors for monitoring physical or environmental conditions, such as motion, temperature, vibration, pressure, sound, or pollutants. Furthermore, sensors can observe other phenomena, not just ambient conditions. These sensors are deployed widely in many remote sensing applications, such as infrastructure monitoring, habitat monitoring, intrusion detection, target tracking and surveillance....

The diversity of fields and environments where the WSN can be implemented makes it necessary to have enough knowledge of communication and signal processing, hardware technologies, embedded system design, and software engineering [33]. When the need arises

to deploy WSN in any environment, a wireless sensor network is considered a subpart of IOT and one of its most important elements [31].

I.7.3 Wireless mesh networks (WMN)

Wireless mesh network is a network which comprises various wireless nodes with access points. Each node in the network acts as a forwarding node to transfer the data. Since the network is decentralized, forwarding of data is possible only to the neighboring node. This results in the network structure simple and easy. [34]

Wireless mesh network is the architecture which provides less mobility with low cost within a radio range. [34]

WMN is a promising technology in providing high bandwidth network coverage. WMNs will greatly help the users to be always on-line anywhere anytime by connecting to wireless mesh routers [35].

I.8 The advantages and disadvantages of Internet Of Things (IoT)

I.8.1 Advantages

- **Communication**

Since IoT has communication between devices, in which physical devices are able to stay connected, and hence total transparency is available with lesser inefficiencies and greater quality.

- **Automation and Control**

Without human involvement, machines are automating and controlling a vast amount of information, which leads to faster and timely output.

- **Monitoring saves money and time**

Since IOT uses smart sensors to monitor various aspects in our daily life for various applications which saves money and time.

- **Better Quality of Life**

IoT based applications increases comfort and better management in our daily life; thereby improving the quality of life.

- **New business opportunities**

Creates new business for IoT technology, hence increases economic growth and new jobs .

- **Better Environment**

Saves natural resources and trees and helps in creating a smart greener and sustainable planet.

I.8.2 Disadvantages

- **Compatibily**

As devices from different manufacturers will be interconnected in IoT, presently, there is no international standard of compatibility for the tagging and monitoring equipment.

- **Complexity**

The IoT is a diverse and complex network. Any failure or bugs in the software or hardware will have serious consequences. Even power failure can cause a lot of inconvenience.

- **Privacy/Security**

IoT has involvement of multiple devices and technologies and multiple companies will be monitoring it. Since lot of data related to the context will be transmitted by the smart sensors, there is a high risk of losing private data.

- **Lesser employment of menial staff**

With the advent of technology, daily activities are getting automated by using IoT with less human intervention, which in turn causes fewer requirements of human resources. This causes unemployment issue in the society.

- **Technology Takes Control of Life**

Our lives will be increasingly controlled by technology, and will be dependent on it. The younger generation is already addicted to technology for every little thing. With IoT, this dependency will spread amongst generations and in daily routines of users. We have to decide how much of our daily lives are we willing to mechanize and be controlled by technology.

Figure I.8 Advantages and Disadvantages of IoT

I.9 Conclusion

The IoT has drawn significant attention in recent years since it has made revolutionary changes in human life. The IoT enables the exchange of information in a wide variety of applications such as smart buildings, smart health, smart transport, and so on.

In this chapter, we introduced the definition of IoT network and as well as their main fields of application, the IoT architecture and the different types of networks such as LLN, WSN, WMN ... etc. the RPL protocol will be the subject of the next chapter.

Chapter 2:

The RPL protocol

II Chapter 2 The RPL protocol

II.1 Introduction

RPL was developed by the ROLL Working Group and adopted by the IETF (Internet Engineering Task Force) as the standard routing protocol for LLNs.

RPL builds the network topology through a graph called a Directed Acyclic Graph (DAG), which consists of one or more graphs called Destination Oriented Directed Acyclic Graph (DODAG) , each DODAG represents a routing tree created by the root node, also known as the network information receiver node. RPL, unlike other well-known routing protocols, uses more parameters to calculate the best path, for example, routing metrics and OF (objective function).In the following sections, we will talk about RPL topology, routing mechanism, control messages, OF (objective function), Trickle Timer, and attacks on RPL topology.

II.2 RPL Protocol

RPL is one of the well-known IPv6 routing protocols for Low Power Loss Networks (LLNs) and was established by ROLL in RFC 6550 to address limitations of LLNs such as low processing power, battery , and memory. A key feature of RPL is that it represents a specific routing solution for low power and lossy networks. The protocol was designed to be highly adaptive to network conditions and to provide alternate routes whenever default routes are inaccessible. RPL provides a way to distribute data across a dynamically constructed network topology.

II.3 RPL control messages

The RPL use ICMPv6 control message to send the source address that is a link-local address, and destination address that can be a link-local uncast address of the destination or multicast address of all RPL nods, There are different types of control messages in RPL for information exchange and topology maintenance[36] ,these control messages are:

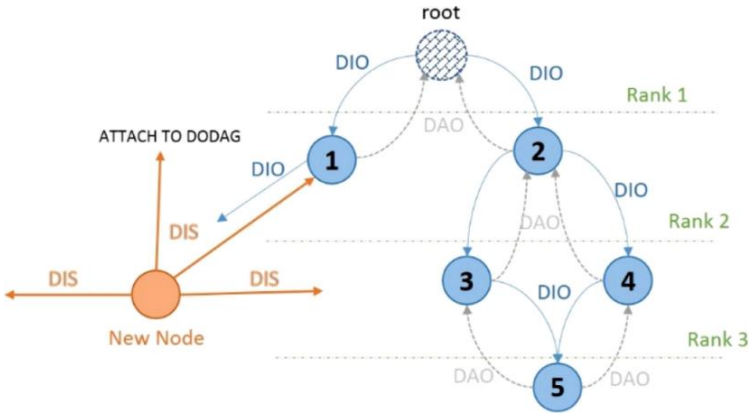


Figure II.1. Flow of RPL control messages

II.3.1 DIO (DODAG Information Object)

The DIO message (DODAG Information Object): The DIO message structure contains all the information concerning the network that allows any node to find an RPL Instance, learn its configuration parameters, choose a DODAG parent packet, and keep DODAG updated. Initially, the DODAG root broadcasts a downward DIO message to neighbor nodes and enables point-to-multipoint traffic in an upward direction. DIO messages contain the RPL Instance ID, routing metrics, rank, intent function, and DODAGID. These messages are sent periodically with a cumulative number in order to start the process of choosing a parent[37].

II.3.2 DIS (DODAG Information Solicitation)

Each new node (not belonging to any DODAG) sends this message to solicit the necessary information in order to join the DODAG. The receiving node responds by sending a DIO packet containing the information of DODAG.

II.3.3 DAO (Destination Advertisement Object)

the DAO message is used to relay reverse track information to record upstream nodes. DAO messages are sent to add routing tables with their children's prefixes by some node other than DODAG root and advertise their child's addresses and prefixes. After this DAO message passes through the default DAG path from a specific node to the DODAG root, a complete path between the DODAG root and the DODAG node is created.

II.3.4 DAO-ACK (Destination Advertisement Object Acknowledgement)

The DAO-ACK message is sent as a unicast packet by a DAO recipient (a DAO parent or DODAG root) in response to DAO message.

II.4 RPL Rank

In Figure 2, an example of a DODAG is shown. The nodes in the RPL-network are assigned with a rank, depending on where in the network you are. Rank 1 is the border router connected to the internet. The rank is used to discover network loops. A set of candidate neighbors exists for each node in the DODAG, which is a subset of the nodes that can be contacted via link-local multicast. The parent set is a restricted subset of this candidate neighbor set, and the preferred parent is a member of the parent set [38]. A node must not advertise a rank lower than any of the members of the parent set. The parent of a node is decided based on the rank of the parent. There exist some special cases where the node's parent is not the node with the lowest rank in the parent set. However, a common way to determine a node's parent is to choose the node with the minimum rank as its appropriate parent, because this means that it is closer to the border router. A node calculates its rank frequently and takes actions based on the new rank.

II.5 Construction of DODAG

The administrator configures the DODAG root node, which is responsible for constructing the complete DODAG topology, initially, the root node [39] calculates RPL instance ID, DODAGID, DODAG version number, base rank, objective function (OF), routing cost, and related information. Sink node multicasts this information in DIO control message to the neighboring nodes. Neighboring nodes extract and use the received information on receiving DIO messages to update their rank, to join DODAG and based on best rank chooses preferred parent. Instantly, they send a DAO message to their preferred parent representing that they have joined the network.

Each preferred parent is an intermediate node, which operates as a router between the child node and the root node. The root node is the preferred parent of all initial hop nodes in the network. The initial hop nodes further transmit DIO messages in the downward direction, whereas, the receiving nodes send DAO message upwards to their preferred parent.

In Fig II.2, DODAG root (node 1) sets its rank to (R = 1) and fills all other essential information in the DIO control message and multicasts it to the neighbors.

Neighbor nodes (5, 12, and 19) calculate their own ranks bearing in mind the objective function and they decide to add DODAG root as their preferred parent. After being a part of DOADG nodes 5, 12, and 19 multicasts their DIO messages with rank 2 for neighbors. The rank increases downwards in the DODAG. DODAG root ignores DIO from these nodes because of higher rank value depicting that it is coming from downward. Node 12 can add nodes 5 and 19 as a candidate parent if it is in the radio range of node 12. Noticeably, all further downward nodes receive DIO messages from multiple neighboring nodes but they choose a preferred parent, which has the best rank. Until all the nodes join the DODAG, topology formation continues.

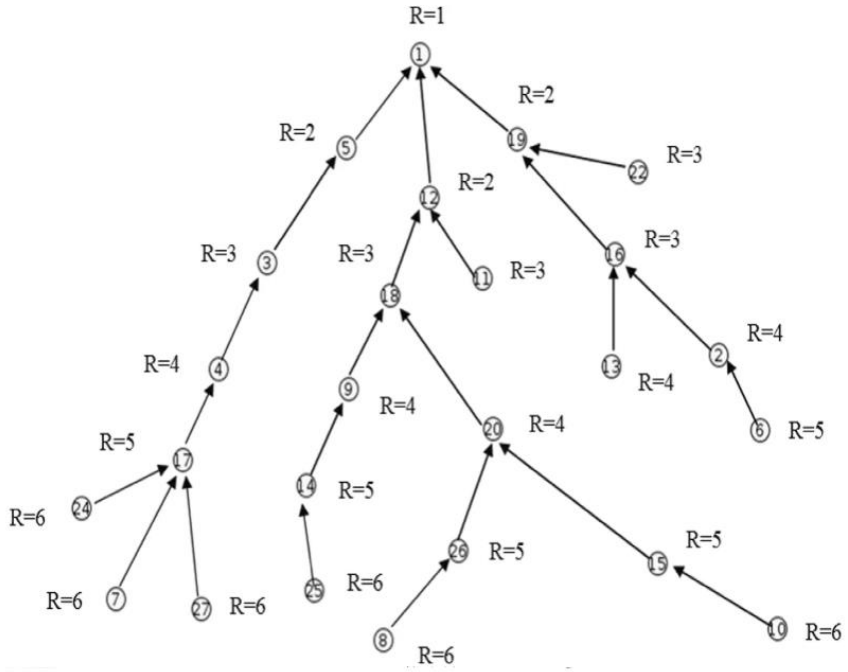


Figure II.3 RPL topology based on ranks[40]

II.6 The operating modes of the RPL protocol

Two modes of operations are supported by the RPL protocol in order to maintain downward routes. In this section, storing and non-storing mode of the RPL protocol [41] are highlighted.

II.6.1 Mode non storing

In the non-storing mode, leaf nodes unicast DAO message to the DODAG root node. Unlike storing mode, intermediate router nodes do not store any information from DAO message; instead, they only append their address to it and forward to the parent. It is done to form a reverse routing path. Thus, only the DODAG root knows a path to every node in the network.

II.6.2 Mode storing

In the storing mode, downward routes start to propagate from leaf nodes to root node through intermediate router nodes. Every child node sends DAO message to its parent who initially stores information contained in that message and later sends a new DAO message containing aggregated reachability information to its parent. Thus, each node knows the path to every other node in the RPL network.

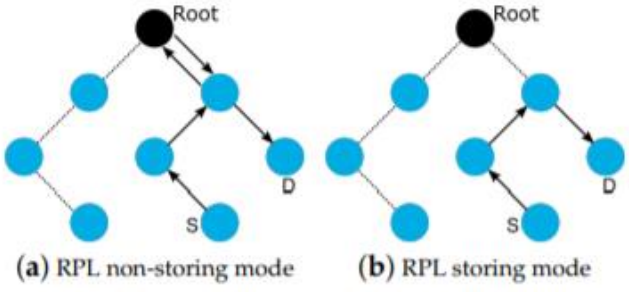


Figure II.4 Storing mode and non-storing mode[42]

II.7 Traffic Flows Supported by RPL

II.7.1 Multipoint-to-Point (MP2P)

RPL was designed primarily to optimize the Multipoint-to-Point (MP2P) type of traffic flow, this MP2P communication was provided by constructing the routes from each node to the DODAG root using DIO's of the preferred parent d a node these are “Upward Routes” MP2P stream destinations are designated nodes that have some importance to the application.

II.7.2 Point-to-Multipoint (P2PM)

In P2MP, a device sends messages downward, from one device to a set of other devices. P2MP is a traffic pattern required by several LLN applications. RPL supports P2MP traffic by using a destination advertisement mechanism that provisions Down routes toward destinations (prefixes, addresses, or multicast groups), and away from roots.

II.7.3 Point-to-Point (P2P)

For P2P traffic, the construction of routes depends on the operating mode of the RPL protocol. If Non-Storing mode case the packet is directing to a root, then the root will perform routing to the destination, if Storing mode case the packet is flowing towards the root until it reaches an ancestor who has a known route to the destination. This common ancestor may be the root DODAG. In other cases, it may be a node closer to the source or destination.

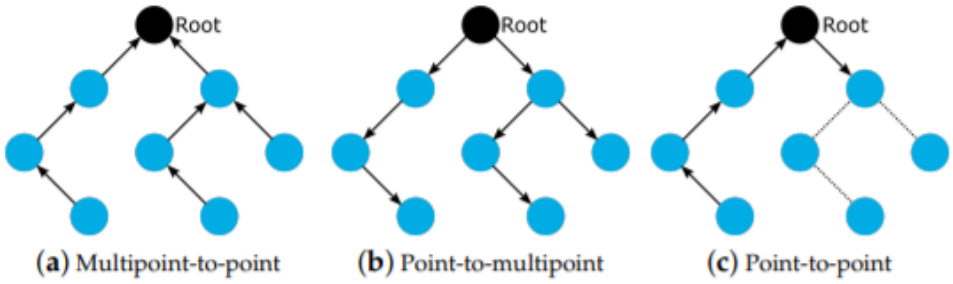


Figure II..5 Traffic patterns supported by RPL. Lines with arrows indicate the traffic flow, while dotted lines without arrows indicate the links of the routing topology.[42]

II.8 Objective Function

One of the main features of RPL is the objective function (OF). Due to the OF, RPL builds the roots in the network. The OF is based on the routing metrics. These metrics are translated into a rank value. The routing metrics are divided into two categories:

node metrics and link metrics. The node metrics are related to the consumption of nodes or its position as remaining energy and hop count metrics while the link metrics are related to the path between nodes. The link metric can be the number of expected transmission count (ETX), throughput, latency, and link quality level (LQL). The RPL specification is that it has flexibility in defining metrics that can be used in the OF according to the user needs.

The routing metrics can be used as a single metric [43] (based on one metric) or composite metrics where more than one metric can be used [44]. Until now, the ROLL working group has specified two OFs: objective function zero (OF0) [45] and the minimum rank with hysteresis objective function (MRHOF) [46]. OF0 selects the best parent based on the minimum number of hop count while MRHOF uses the minimum ETX as a criterion to select the optimal route toward the sink node.

II.9 Trickle Timer

The emission of DIOs is regulated by the Trickle algorithm [47]. Trickle was originally designed for polite gossiping in wireless networks, to reduce the power consumption of the nodes by minimizing the redundant messages and by dynamically adapting the transmission rate. In particular, the emission rate of DIOs is tuned according to the stability of routing information. If the information included in DIOs from the neighbors is consistent with internal routing information, then the emission rate is reduced.

II.9.1 Parameters and Variables [47]

A trickle timer runs for a defined interval and has three configuration parameters:

The minimum interval size is Imin, the maximum interval size is Imax, and a redundancy constant k.

The minimum interval size, Imin, is defined in units of time (e.g., milliseconds, seconds).

The maximum interval size, Imax, is described as a number of doublings of the minimum interval size (the base-2 log (max/min)).

The redundancy constant, k, is a natural number (an integer greater than zero).

In addition to these three parameters, Trickle maintains three variables:

- I, the current interval size.
- t, a time within the current interval.
- c represents a counter.

II.9.2 Algorithm Description [47]

The Trickle algorithm has six rules:

1) When the algorithm starts execution, it sets I to a value in the range of

[**Imin**, **Imax**]—that is, greater than or equal to **Imin** and less than or equal to **Imax**. The algorithm then begins the first interval.

2) When an interval begins, Trickle resets **C** to 0 and sets t to a random point in the interval, taken from the range [**I**/2, I], that is, values greater than or equal to **I**/2 and less than i. The interval ends at **I**.

3) Whenever Trickle hears a transmission that is "consistent," it increments the counter **C**.
4) At time t, Trickle transmits if and only if the counter **C** is less than the redundancy constant **K**.

5) When the interval **I** expires, Trickle doubles the interval length. If this new interval length would be longer than the time specified by Imax, trickle sets the interval length to be the time specified by Imax.
6) If Trickle hears a transmission that is "inconsistent" and I is greater than **Imin**, it resets the Trickle timer. To reset the timer, Trickle sets it to **Imin** and starts a new interval as in step 2. If **I** am equal to **Imin** when Trickle hears an "inconsistent" transmission, Trickle does nothing. Trickle can also reset its timer in response to external "events."

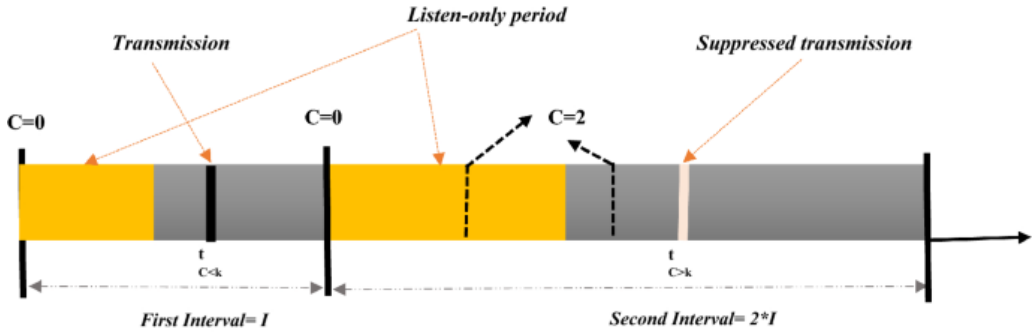


Figure II.6 Trickle algorithm during two intervals illustrating the transmission, listen-only period and the suppression mechanism

II.10 RPL security

The current RPL specification includes a few self-healing mechanisms, like loop detection and avoidance, global and local repair mechanisms. Furthermore, it defines security features like cryptographic security modes that are presented in the following subsections.

II.10.1 Self-healing mechanisms

- Loop detection and avoidance

Data packets must be transmitted upward from a child to its parent, where the parent has a lower Rank value than its child. This is known as the Rank rule. Hence, nodes could use the packet direction and Rank information conveyed in control messages to discover and avoid possible loops [48].

- Global and local repairs

RPL provides global repair and local repair mechanisms to fix links and node failures, and detect loops and other inconsistencies. Global repair is instituted by incrementing the DODAG Version Number field within the DIO message.

Only the BR (DODAG root) could trigger this mechanism. However, any non-root node that detects an inconsistency (e.g., loop or link failure) can start a local repair. The node should poison its routes by announcing a rank of INFINITE RANK.

Thus, it detaches itself from the DODAG and then reattaches to the DODAG as a new joining node using a DIS message [48]. Malicious nodes could exploit both global and local repairs to trigger specific attacks against RPL networks.

II.10.2 Security features

The self-organizing, self-healing, and resource-constrained, as well as unreliable links, limited physical security, and dynamic topology of RPL networks, expose them to various internal and external threats. The RFC 6550 [48] states that RPL could use link-layer security mechanisms when they are available to secure message transmission. Furthermore, the RPL specification defines the following optional cryptographic security modes that nodes within an RPL network can adopt to ensure communication security.

- **Unsecure mode**

In this mode, RPL control messages are transmitted without any additional security features [48]. In this case, RPL relies on other layer security primitives, such as the MAC layer, to satisfy the network's security requirements [48]

- **Pre-installed security mode**

In this mode, the nodes have pre-installed keys to generate and process RPL secured messages [48].

- **Authenticated security mode**

Like the pre-installed mode, the nodes have pre-installed keys; nevertheless, they may only use the keys to join the network as a leaf. A router that needs to enter an RPL network requires another key from an authentication authority [48]. Despite the modes mentioned above, RPL networks remain vulnerable to existing and newly designed threats that have been

extensively studied in the literature. Precisely, Rank attacks, Neighbor attack, DIO suppression attack, Sinkhole and Blackhole attacks, Wormhole attack.

II.11ATTACKS ON RPL TOPOLOGY

There are several types attacks that can be executed in a RPL network. We only focus on the topology of the attacks.

Attacks on Topology include attacks against RPL network topology. These attacks aim to disrupt the normal operation of the network. These may lead to the isolation of one or more nodes. This category can also be divided into two sub-categories:

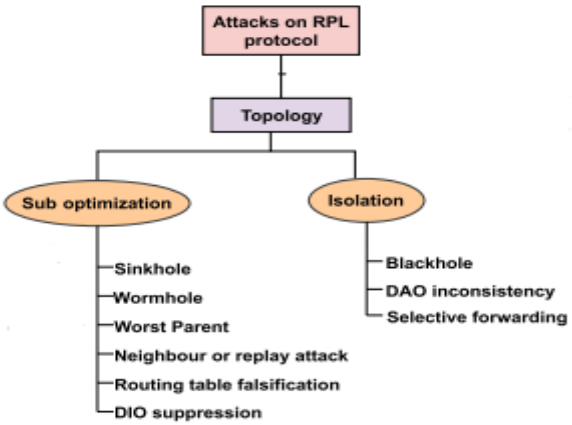


Figure II.7 Taxonomy of attacks on RPL topology

II.11.1 Sub-optimization Attacks

II.11.1.1DIO Supression Attack

In case of sub-optimization attacks, the network will not converge to the optimal form (i.e optimal paths) inducing poor performance.

The purpose of the DIO suppression attack is to disrupt or slow down the network's transmission of DIO messages. For this purpose, Trickle's DIO suppression method is used. During this attack, the adversary continuously sends a DIO message that the receiving nodes regard as consistent. Suppose the nodes get a sufficient number of consistent DIOs. In that case, they disable their own DIO transmission, resulting in a general decrease in the quality of the routes or, in the worst-case scenario, a network breakdown [49].

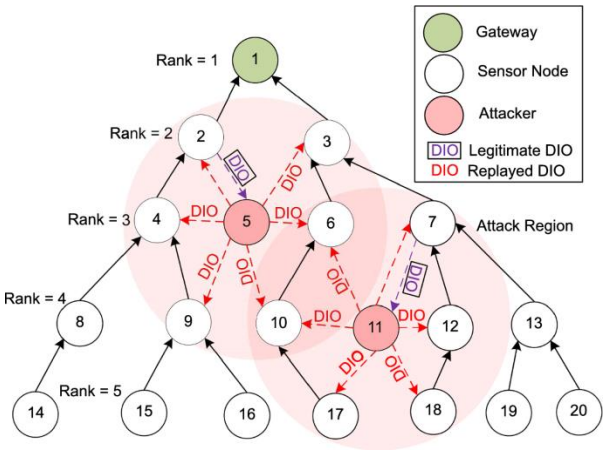


Figure II.8 Illustration of DIO Suppression Attack[50]

II.11.1.2 WormHole Attack

In a WormHole attack, two or more attackers collaborated to establish a virtual tunnel between them to pass the traffic, entirely or selectively, through it instead of its original route. Therefore, such an attack disrupts the network topology, exhausts network resources, and provides the attackers with access to sensitive information [51].

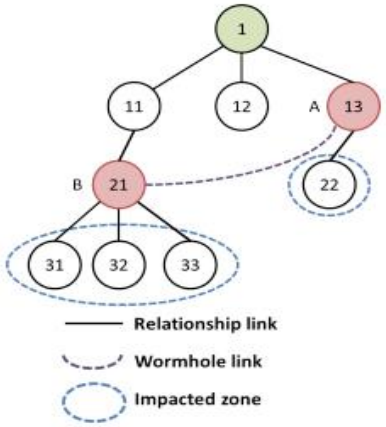


Figure II.9 Illustration of WormHole Attack[52]

II.11.1.3 SinkHole Attack

A malicious node broadcasts itself as the best convenient route (optimal path) to be a preferred parent for the surrounding nodes. Then, the network traffic of the child nodes will be forwarded to the SinkHole node. Therefore, this attack disrupts the communication and leads to other kinds of attacks.

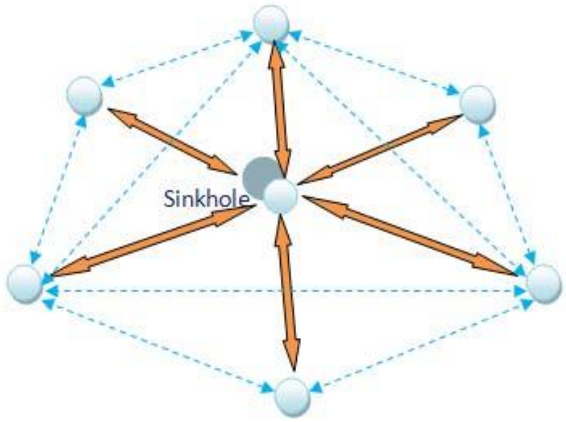


Figure II.10 Illustration of SinkHole Attack[53]

II.11.2 Isolation Attacks

The attacks against the topology also serve as a support for isolating a node or a subset of nodes in the RPL network which means that those nodes are no longer able to communicate with their parents or with the root.

II.11.2.1 BlackHole Attack

In a BlackHole attack, the pernicious node announces itself as the shortest route to the destination. All the packets arriving at this node will be dropped and, thus, prevented from reaching their destinations. Therefore, this attack will create a hole in the network without the senders being aware of their packets delivery status[54].

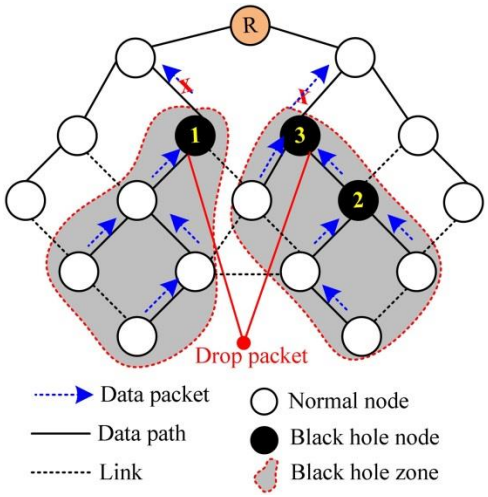


Figure II.11 Illustration of BlackHole Attack[55]

II.12 Conclusion

This chapter is devoted to the detailed description of the RPL protocol. we explained the build procedure for DODAG, and the types of control messages used (DIO, DAO, DIS), operating modes (non-storting and storting), the objective function as well as Trickle Timer algorithm. At the end of this chapter, we explained RPL security and types attacks that can be executed in a RPL network and we only focused on the topology of the attacks. In the next chapter, we will evaluate the RPL protocol against routing metrics.

Chapter 3:

Conception and Implementation

III Chapter3 Conception and Implementation

III.1 Introduction

The Internet things has many problems but the most severe one is how to save energy while providing QoS-effective communications with acceptable security services. The goal of the DIO suppression attack is to general degradation of the routes' quality or in the worst case a partition of the network and that is why they are considered among the dangerous attacks in the IoT.

This chapter consists of two parts: the first one presents the assumed network model, as well as the threat model. The second part of the chapter provides detailed presentation of evaluation contexts of our solution.

III.2 Conception

In this parts we start by the design of the network, threat, solution.

III.2.1 Model DIO suppression attack

The goal of the DIO suppression attack is to interrupt or slow down the transmission of the DIO messages in the network. To this aim, the DIO suppression mechanism of the Trickle algorithm is exploited. In this attack, the attacker requires the adversary to transmit only k DIO messages at each Trickle period.

The adversary transmits repeatedly a DIO message that is considered consistent by the receiving nodes. If the nodes receive enough consistent DIOs, they will suppress their own DIO transmission. Since DIO messages are exploited to discover neighbors and the network topology, their continuous suppression can cause some nodes to remain hidden and some routes to remain undiscovered.

A simple way to mount a DIO suppression attack is to eavesdrop a DIO message from a legitimate node and then replay it many times with a fixed frequency. The surrounding legitimate nodes will consider the replayed DIOs consistent. Indeed, receiving a DIO equal to the last received one will cause no changes in their parent set, their preferred parent, or their distance to the root.

We proposed the solution detect this attacker will been also detailed in the next part.

III.2.2 trickel timer

is algorithm that controls the emission of Data Information Objects (DIOs) which are the control traffic messages responsible for constructing the upward routes in RPL routing protocol.

Trickle uses two primitive and simple operations to regulate its transmissions. First, a node in Trickle suppresses scheduled transmission should hear enough number of its neighbouring nodes that transmit the same piece of information. Second, a node should increase the frequency of data transmission whenever an inconsistent data has been received (e.g. its parent change its rank) for quickly resolving the resulting inconsistency, and exponentially decreases data transmission rate each time it hears a consistent data

The Trickle timer has three configuration parameters:Imin,Imax,k

III.2.3 Model Of Solution

The specific variables and structures of the detection mechanism are initialized., tcurrent stores the current system time. Moreover, array namely node table are maintained to store nodes information respectively.

The node information (entry in node table) is maintained using a defined structure of two elements, i.e., [<from, timestamp>].

Where, from stores DIO sender, timestamp stores time of DIO receipt. The instructions are Incorporated in the default Contiki RPL's "rpl_process_dio" method. starts by storing the IP of DIO sender and the time of DIO receipt in tcurrent. The investigation is performed by the receiving node if the DIO sender is the same as the previous DIO sender, in case it is the same sender is also checked if the time difference between them equals 128 millisecond If the checked is valid then this sender will be considered an attacker, and the detection scheme is executed every time a node receives a DIO message.

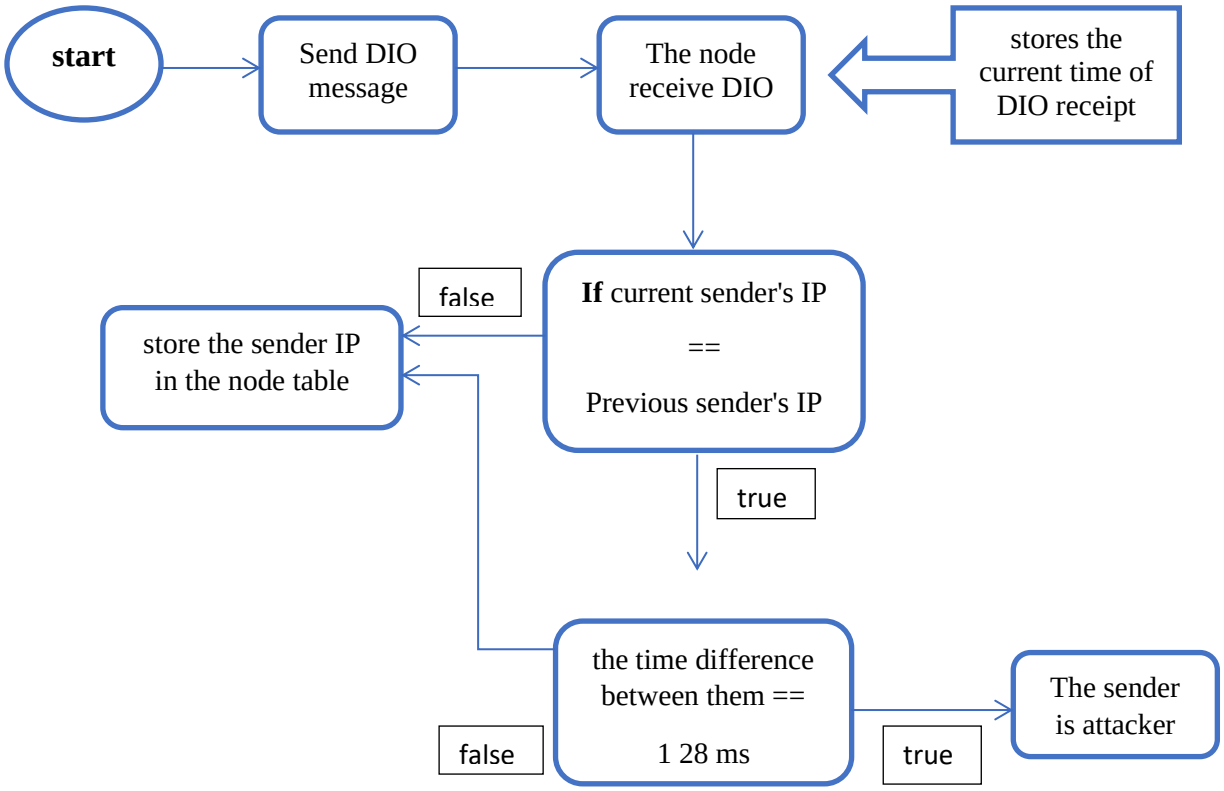


Figure III.1 The architecture of DIO suppression attack detection model.

III.2.4 Pseudo code of solution

```
} else {
  if(p->rank == dio->rank) {
    PRINTF("RPL: Received consistent DIO\n");
    if(dag->joined) {
      instance->dio_counter++;


     if(node_table[f].senderip== from && (new-node_table[f].timestamp )==128){
      PRINT6ADDR(from);
      PRINTF(" is an attack\n");
       } else {

        node_table[f].senderip= from;
        node_table[f].timestamp=tcurrent;

              }
```

Figure III.1 Pseudo code of solution

III.3 Implementation

This part contains the platform used for this simulation, and discuss the evaluation results for this simulation.

III.3.1 Contiki OS

Contiki is a lightweight IoT operating system that specifically targets small IoT devices with limited memory, power, bandwidth and processing power, developed at the Swedish Institute of Computer Science By Adam Dunkels and written in the C programming language. Contiki OS requires a minimum of 2KB (RAM) and 30KB (ROM).It supports technologies that assist in coding smart object solutions. It supports libraries for connection stripping and memory allocation. Instant Contiki is an Ubuntu Linux virtual machine that can run on Windows or Linux pre-installed with a VM Ware player running the Linux virtual machine.

The main advantage of Contiki is that it works on a concept that lies between multi-threading and event-driven programming, this allows processes to share the same execution context and therefore improves memory usage. and energy. This is the concept of Protothreads. Contiki

supports both IPv6 and IPv4 stack implementations, as well as less advanced wireless standards like 6lowpan, RPL, CoAP, and the Rime stack.

Contiki contains two communications : uIP and Rime

- The Rime Layer: enables for a conversation with nearby sensors to generate route diagrams.
- The uIP layer : The Internet-oriented uIP layer provides the IP protocol's fundamental services but consumes more resources than Rime.

III.3.2 Cooja Simulation

Cooja is a network simulator offered by Contiki. It is a Java simulator that enables for the simulation of various sensors on which the operating system and applications will be installed. Cooja then lets you simulate network connections and sensor interactions. Nodes run the same code that is loaded on sensors and physical nodes. Exp5483, z1, wismote, micaz, sky, jcreate, sentilla-usb, and esb are among the sensors supported by Cooja. Furthermore it is flexible in the sense that many parts of the simulator can easily be replaced or extended with additional functions The simulation interface consists of many plug-ins that are in the form of windows, which are five as shown in Figure III.2.

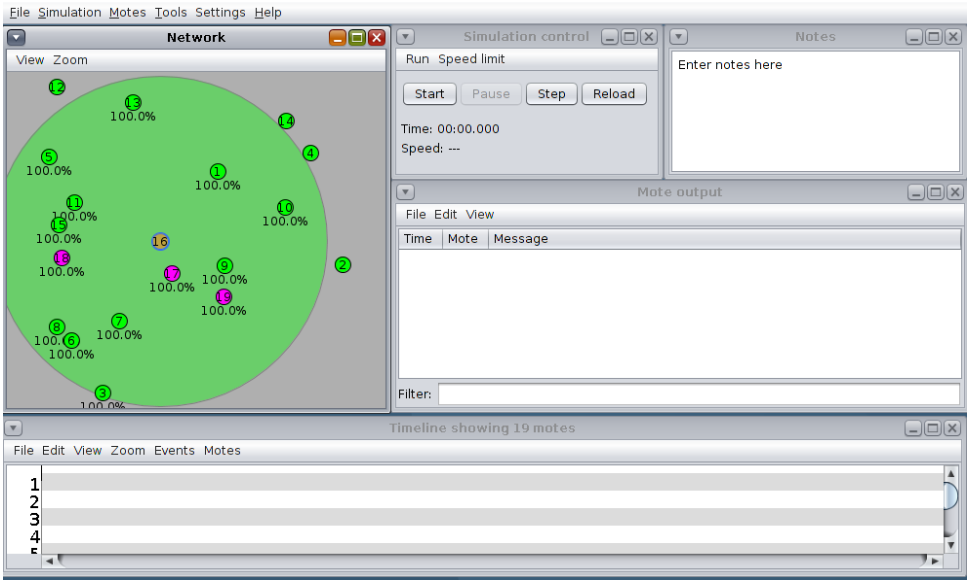


Figure III.3 Cooja simulator interface

- The network window : This area allows you to view each node in the network and to display its status (ID, address, position, etc.) When the simulation is initialized, this area is empty and nodes must be added to it.
- The simulation control window : This area is used to start, reload, or jump to the simulation. There we also find the execution time in addition to the speed.
- The notes window : This is a space for taking notes on the current simulation.

- The motes output window: It's where it shows all the output from the different node interfaces, where we can see everything that happens between the network nodes and the messages they exchange.
- The timeline window: It's where we can see the radio communications (transmission, reception, collision), as well as the awake and sleeping states of the sensor nodes over time.

III.3.3 Performance evaluation

In this section, we describe the experimental framework used to evaluate the proposed method and show the results from the evaluation.

III.3.3.1 Performance Metrics

The criteria identified for analysis are false positives, false negatives, and speed of detection and detection accuracy. A simulation is run with the attack and the detection.

- true positive and false positive

true positive rate (TPR), or detection rate, and false positive rate (FPR).

TPR and FPR (formula 1)

$$TPR = \frac{TP}{TP+FN} \qquad FPR = \frac{FP}{FP+TN} \qquad (1)$$

where true positive (T P) is the total number of successfully detected malicious nodes, false negative (F N) is the number of events that misjudged a malicious node as a legitimate node, false positive (F P) is the number of false alarms raised by misjudging a legitimate node as a malicious node, and True Negative (T N) is the number of events that correctly judged a legitimate node as normal.

- Prediction Accuracy

Prediction accuracy is the number of correct predictions made by the model divided by all the predictions made (formula 2)

$$\text{accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \qquad (2)$$

III.3.3.2 The simulation environment

In this paper, Zolertia 1 (Z1) platform is utilized to act as a 6LoWPAN node. Table1 presents the simulation parameters considered in the experiments. Figure.18 shows the topology setup used in the experiment, in order to simulate three scenarios. The sink node has nodeID 1, and nodeIDs 25.the runtime from 5 min to 15 min.

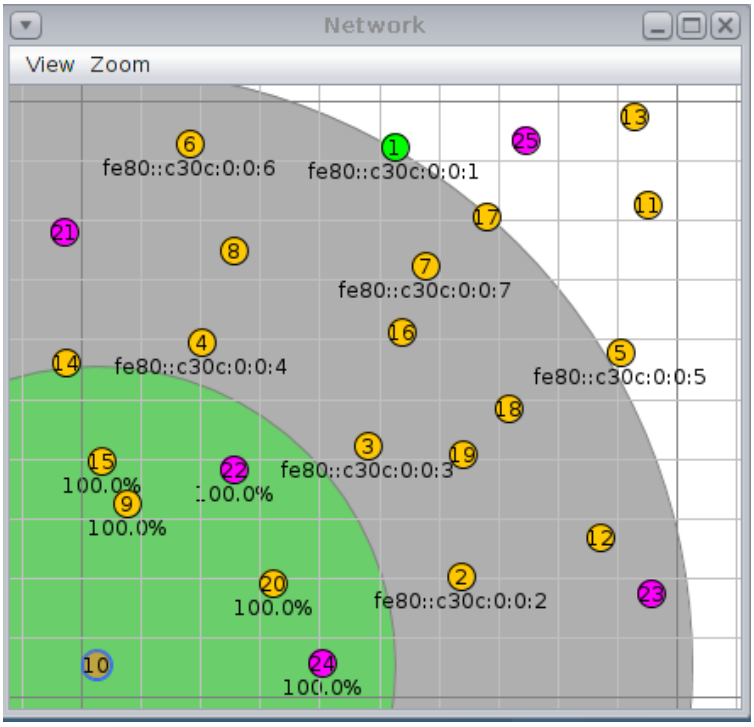


Figure III.4 Network topology and node placement for simulation

Table 3 Evaluation parameters

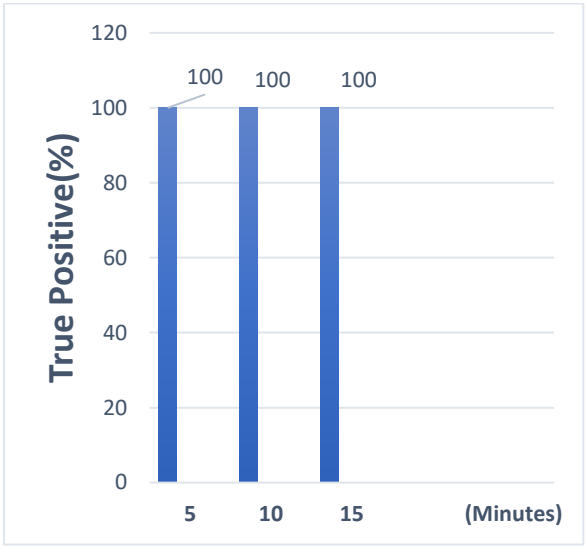
Parameter	Value
Evaluation platform	Cooja Contiki 3.0
Number of nodes	1 server, 25 clients
Number of malicious nodes	2,5,6
Deployment area	100 × 100 m
Simulation runtime	5, 10, 15 minutes

Table 4 Parameters used for three scenarios

Scenario 1	Scenario 2	Scenario 3
2attack	5attacks	6attacks

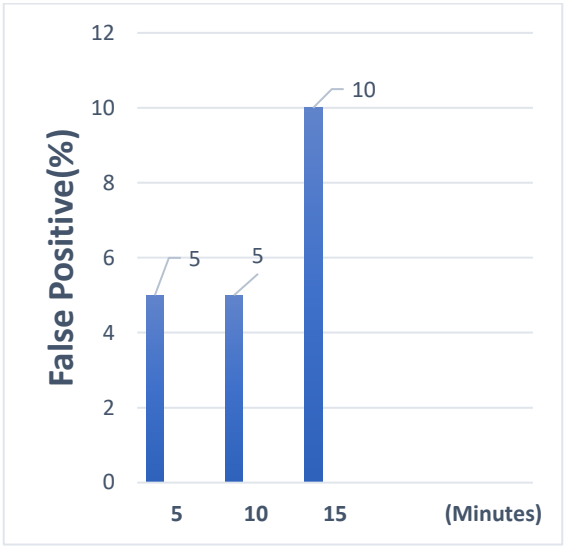
III.3.3.3 Simulation results

We measured TPR and FPR by varying the simulation runtime and the number of malicious nodes. The obtained results are presented in Figure III5.



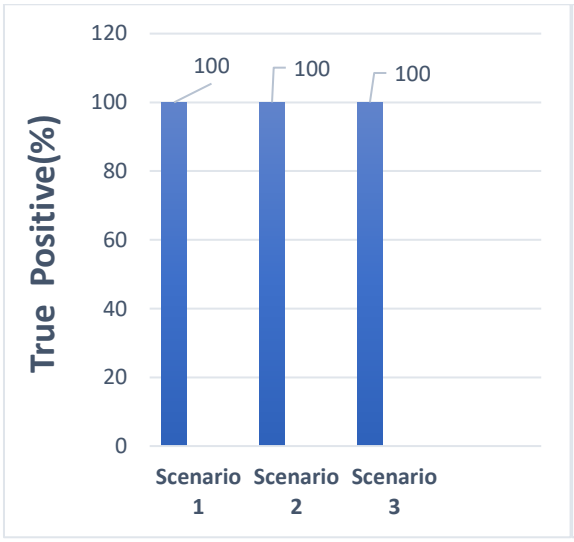
Simulation runtime (minutes)

(a)



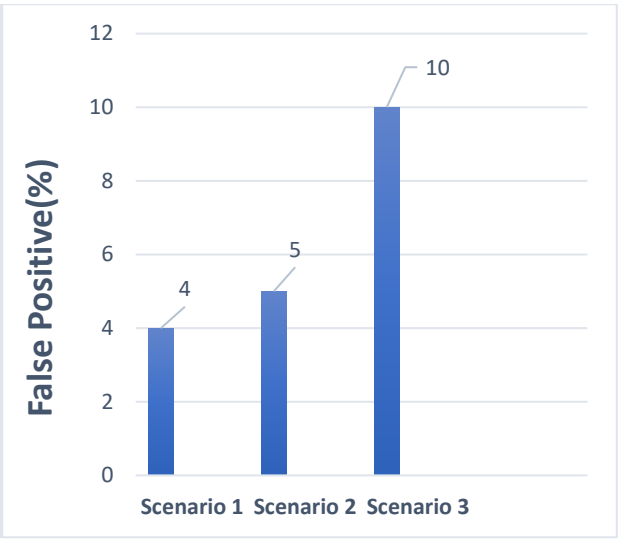
Simulation runtime (minutes)

(b)



The number of malicious nodes

(c)



The number of malicious nodes

(d)

Figure III6. Simulation results of the proposed method: (a) TPR for different simulation runtime (b) FPR for different simulation runtime (c) TPR for different number of malicious nodes (d) FPR for different number of malicious nodes

We first ran experiments for 5, 10,15 min ,to determine the optimized simulation runtime.This is because it needs time to reach a stable network operation. Figures III.5 (a) and (b) show (TPR)as successful detection rates and FPR as false alarm rates, respectively, for different simulation runtimes. In every case, the detection rate is 100% and the false alarm rate is less than 10%.

To see the influence of the number of malicious nodes, we then run experiments by varying the number of malicious nodes. For every case, simulations were run for 15 min, simulation results according to 2,5,6 malicious nodes are shown in Figures III.5 (c) and (d).

Experimental results show that the TPR is 100% and the FPR is less than 10% in all cases. The results show that the higher the number of malicious nodes and the longer the duration, the higher the false alarm rate. This is due to network congestion.

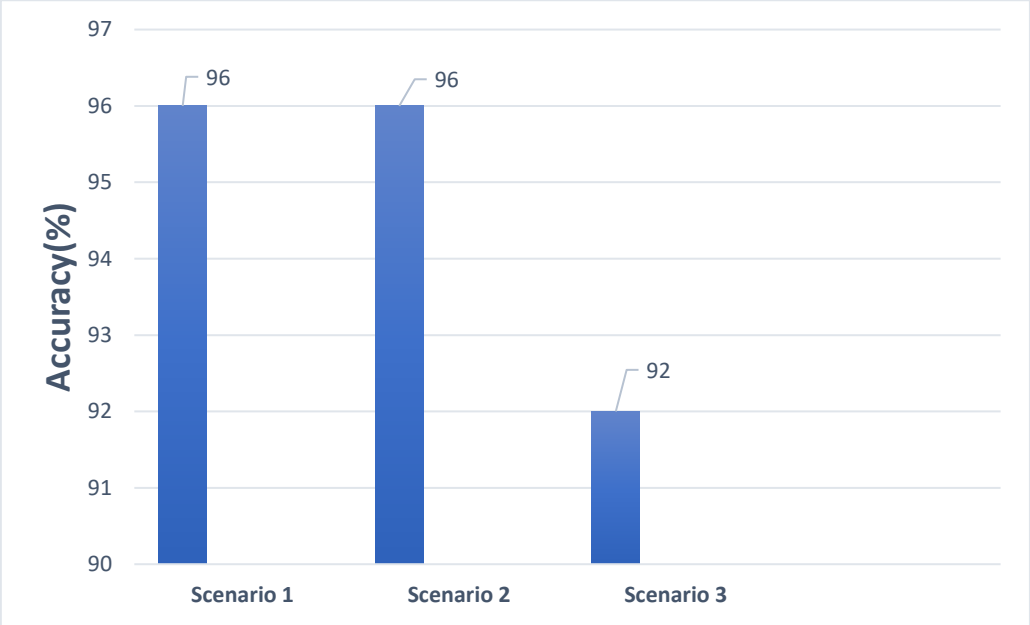


Figure III.7 Detection accuracy

It can be observed from the accuracy of the method performs better in the network by achieving a maximum of 96% and a minimum of 92%.

- **Impact on energy**

Table 5 Parameters used for three scenarios

Scenario 1	Scenario 2	Scenario 3
Simulation without attackers	Simulation with attackers	Simulation with attackers and detection

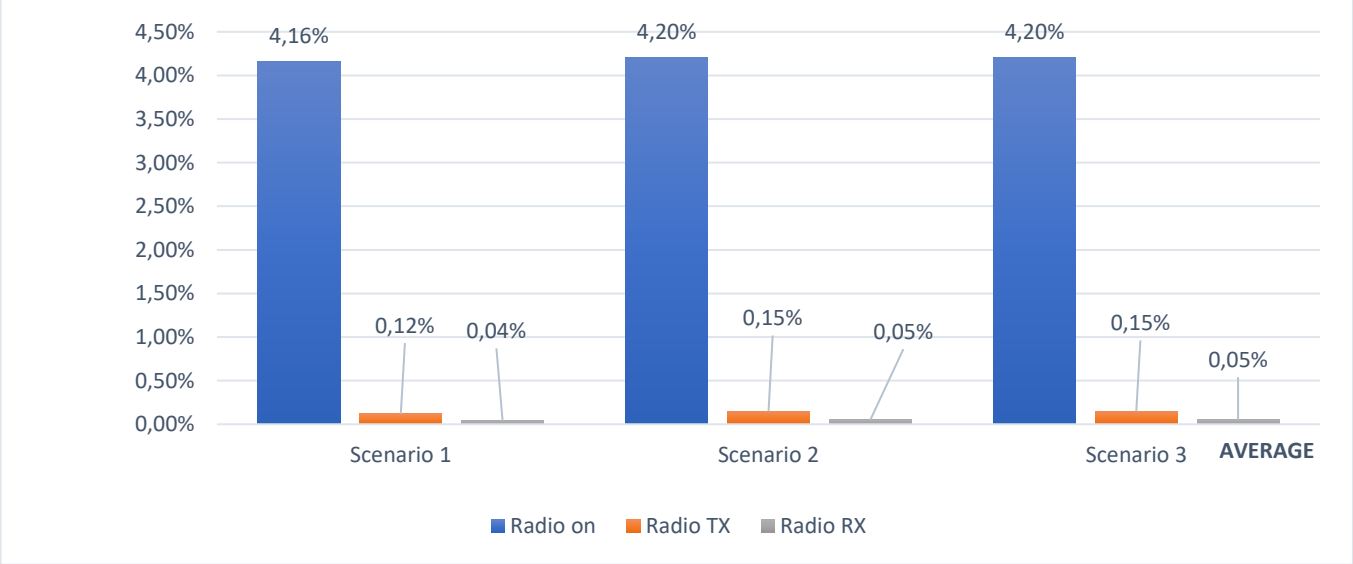


Figure III8. Energy consumption

- Radio on : Energy consumption
- Radio TX : transmission energy consumption
- Radio RX : recieving enrgy consumption

The objective of the study of energy consumption is to compare energy changes before and after the application of the detection process.

Where we note in figure III.7 in the second scenario, in the case of simulation with attackers, the energy consumption increases compared to the first one and this is considered a realistic thing. But with the addition of the method of detecting the attackers, there is no difference compared to the second scenario, indicating that the method of detection did not affect the energy consumption at all.

III.4 Conclusion

In this chapter, we have presented the essence of our work which is about a security mechanism for RPL communications in the IoT and explained the simulator Cooja.

we proposed a detection method for DIO suppression attacks in RPL-based networks and we verified the performance of our detection method based on experiments using Cooja-Contiki. The results showed that the detection method has very good performance: high detection rates 100%, and low false alarm rates (less than 10%) in every case.

General Conclusion

Nowadays, the Internet of Things is a major part of our daily lives. Billions of intelligent and independent beings around the world are connected and communicate with each other. The Internet of Things system uses wireless communication technologies to connect intelligent and autonomous objects. These objects have the ability to collect, analyse, process, create and exchange information in order to provide advanced services. Wireless networks have many limitations such as power and memory, which makes routing in them complex, so RPL was chosen as the actual routing protocol to address the limitations of LLN networks from low processing power and battery and memory. However, RPL is vulnerable to many attacks related to cross-control messages. In this document , we propose an algorithm capable of detecting a DIO suppression attack and identifying malicious nodes to improve the security and performance of the RPL protocol. This is based on the DIO surrender, storing the surrender time, and then executing the action by calculating the time difference between each message and a message from the same node in the sequence. We used cooja as a development environment to implement our solution and after we implemented the proposed solution , which lies in detection of DIO suppression attack and we came to a satisfactory result.

As a future work, we intend to improve our method of not detecting legitimate nodes as malicious and also implement a removal method that can remove malicious nodes from the network. Unfortunately, there was not enough time.

Reference

[1] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. Future generation computer systems, 29(7):1645–1660.

[2] Ovidiu Vermesan and Peter Friess. Internet of things: converging technologies for smart environments and integrated ecosystems. River Publishers.

[3] https://www.researchgate.net/publication/351554927_Internet_of_things_IoT.

[4] https://www.oracle.com/internet-of-things/what-is-iot/

[5] O.Vermesan,P.Friess,internet of things-Converging Technologies for Smart Environments and Integrated Ecosystems.

[6] K. Zhao and L. Ge. A survey on the internet of things security, in Com- putational Intelligence and Security (CIS), 2013 9th International Con- ference, on pp.

[7] https://www.geeksforgeeks.org/architecture-of-internet-of-things-iot/.

[8] M. Wu, T. J. Lu, F. Y. Ling, J. Sun, and H. Y. Du. Research on the architecture of Internet of Things, in Proc. 3rd ICACTE.

[9] I. Mashal, O. Alsaryrah, T.-Y. Chung, C.-Z. Yang, W.-H. Kuo, and D. P. Agrawal, "Choices for interaction with things on Internet and underlying issues," Ad Hoc Networks, vol. 28, pp. 68–90.

[10] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future internet: the internet of things architecture, possible applications and key challenges," in Proceedings of the 10th International Conference on Frontiers of Information Technology (FIT '12), pp. 257–260.

[11]https://fardapaper.ir/mohavaha/uploads/2018/10/Fardapaper-Internet-of-Things-A-Survey-on-Enabling-Technologies-Protocols-and-Applications.pdf

[12] Jara, A.J., Zamora, M.A., Skarmeta, A.: Glowbal IP: an adaptive and transparent IPv6 integration in the Internet of Things. Mob. Inf. Syst. 8(3) .

[13] Burhan, M., Rehman, R., Khan, B. and Kim, B. (2018) IOT Elements, Layered Architectures and Security Issues: A Comprehensive Survey. Sensors, 18, 2796-2812.

[14] Wani, U. (2019) An Introduction to IOT, Its Architecture and Various Protocols. IEEE Conference Paper, ID: 33482413.

[15] Burhan, M., Rehman, R., Khan, B. and Kim, B. (2018) IOT Elements, Layered Architectures and Security Issues: A Comprehensive Survey. Sensors, 18, 2796-2812.

[16] Marques, G., Garcia, N. and Pombo, N. (2017) A Survey on IOT: Architectures, Elements, Applications, QoS, Platforms and Security Concepts. In: Advances in Mobile Cloud Computing and Big Data in the 5G Era, Springer, Berlin, 2-24.

[17] Scully P., (2018). The Top 10 IoT Segments in 2018 – based on 1,600 real IoT projects. Retrieved on January 13, 2019 from https://iot-a nalytics.com/top-10-iot-segments-2018-real-iot-projects.

[18] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. across-the-mobile-ecosystem.pdf.

[19] X. Sun and N. Ansari, "Dynamic resource caching in the IoT application layer for smart cities," IEEE Internet Things J., vol. 5, no. 2, pp. 606–613.

[20] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Resource provisioning for IoT application services in smart cities," in 2017 13th International Conference on Network and Service Management (CNSM), 2017, pp.

[21] G. Rathee, S. Garg, G. Kaddoum, and B. J. Choi, "A decision- making model for securing IoT devices in smart industries," IEEE Trans. Ind. Inform.

[22] M. Elhoseny, A. Abdelaziz, A. S. Salama, A. M. Riad, K. Muhammad, and A. K. Sangaiah, "A hybrid model of internet of things and cloud computing to manage big data in health services applications," Future Gener. Comput. Syst., vol. 86, pp. 1383– 1394.

[23] J. Ruan et al., "A life cycle framework of green IoT-based agriculture and its finance, operation, and management issues," IEEE Commun. Mag., vol. 57, no. 3, pp. 90–96.

[24] H. O. Al-Sakran, "Intelligent traffic information system based on integration of Internet of Things and Agent technology," Int. J. Adv. Comput. Sci. Appl. IJACSA, vol. 6, no. 2, pp. 37–43.

[25] A.-R. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using IoT and big data analytics approach," IEEE Trans. Consum. Electron., vol. 63, no. 4, pp. 426–434.

[26] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. McCann, and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: Standards, challenges, and opportunities," IEEE Wirel. Commun., vol. 20, no. 6, pp. 91–98.

[27] J.-P. Vasseur and A. Dunkels, "Ip for smart objects," White Pap., vol. 1.

[28] J. P. Vasseur et al., "A survey of several low power link layers for IP smart objects," Internet Protoc. Smart Objects IPSO Alliance.

[29] J. Vasseur, N. Agarwal, J. Hui, Z. Shelby, P. Bertrand, and C. Chauvenet, "RPL: The IP routing protocol designed for low power and lossy networks," Internet Protoc. Smart Objects IPSO Alliance, vol. 36.

[30] Ee, G., Ng, C., Noordin, N. and Ali, B. (2010) A Review of 6LoWPAN Routing Protocols. Proceedings of the Asia-Pacific Advanced Network, 30, 71-81

[31] Polgavande, A. and Kulkarni, A. (2017) Internet of Things (IOT): A Literature Review. International Journal of Research in Advent Technology.

[32] Abdul, M.M. and Islam, N. (2012) Overview of Wireless Sensor Network. Intech, London.

[33] Gupta, C. and Kumar, A. (2013) Wireless Sensor Networks: A Review. International Journal of Sensors, Wireless Communications and Control, 3, 25-36.

[34] https://www.intechopen.com/chapters/66938.

[35] Ian F Akyildiz, Xudong Wang, and Weilin Wang. Wireless mesh networks: a survey. Computer Networks, (47).

[36] Internet Engineering Task Force. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. https://tools.ietf.org/pdf/rfc6550.pdf.

[37] Conta, Alex, and Mukesh Gupta. "Internet control message protocol (icmpv6) for the internet protocol version 6 (ipv6) specification.".

[38] Internet Engineering Task Force. RPL: IPv6 Routing Protocol for Low-Power and

Lossy Networks. https://tools.ietf.org/pdf/rfc6550.pdf .

[39] Winter T, Thubert P (2010) RFC 6550: RPL: IPv6 routing protocol for Low-Power and Lossy Networks, Internet Engineering Task Force (IETF) Request For Comments.

[40] https://link.springer.com/content/pdf/10.1007/s12243-018-0645-4.pdf

[41] Gaddour, O., & Koubâa, A. (2012). RPL in a nutshell: A survey. Computer Networks, 56(14), 3163–3178.

[42] https://www.mdpi.com/1424-8220/19/9/2144/htm.

[43]P. O. Kamgueu et al., "Energy-based routing metric for RPL," [Research Report] RR-8208, INRIA, pp. 14.

[44] H. Lamaazi and N. Benamar, "RPL enhancement using a new objective function based on combined metrics," in Pr oceedings of the 13th International Wireless Communications and Mobile Computing (IWCMC), Valence, Sapain, 2017.

[45] P. Thubert, "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)," RFC 6552, pp. 1-14.

[46] O. Gnawali and P. Levis, "The Minimum Rank with Hysteresis Objective Function," RFC 6719, pp. 1-13.

[47] P. Levis, T. Clausen, J. Hui, O. Gnawali, and J. Ko, "The Trickle algorithm," Internet Requests for Comments, RFC 6206.

[48] Winter T, Thubert P, Brandt A, Hui J, Kelsey R, Levis P, Pister K, Struik R, Vasseur J, Alexander R. RPL: IPv6 routing protocol for low-power and lossy networks. RFC 6550, Internet Engineering Task Force.

[49]Perazzo, P.; Vallati, C.; Anastasi, G.; Dini, G. DIO suppression attack against routing in the internet of things. IEEE Commun. Lett. 2017, 21, 2524–2527.
[50]https://link.springer.com/article/10.1007/s00607-020-00862-1.

[51]L. Wallgren, S. Raza, and T. Voigt, "Routing attacks and countermeasures in the RPL-based internet of things," International Journal of Distributed Sensor Networks, vol. 2013, pp. 1-11, 2013.

[52]https://hal.inria.fr/hal-01207859/document

[53]https://www.researchgate.net/publication/303009441_Study_on_Sinkhole_Attacks_in_Wireless_Ad_hoc_Networks.

[54] D. Airehrour, J. A. Gutierrez, and S. K. Ray, “A trust-aware RPL routing protocol to detect blackhole and selective forwarding attacks,”.

[55] https://onlinelibrary.wiley.com/doi/epdf/10.1002/sec.1684