An Overview of RPL Networks from the Viewpoint of Cybersecurity

Cosmina STALIDI, **Eduard-Cristian POPOVICI, George SUCIU**
Telecommunications Department & Research & Development Department, Faculty of Electronics, Telecommunications and Information Technology & Beia Consult International, Bucharest, Romania
cosmina.stalidi@beia.ro, eduard.popovici@upb.ro, george@beia.ro

Abstract

In the past decade, the Internet of Things (IoT) has had a significant impact on a global scale. The Internet of Things (IoT) has facilitated the interconnection of a vast number of devices in contemporary times. The proliferation of Internet of Things (IoT) devices underscores the importance of ensuring robust security measures to safeguard against potential threats. The RPL protocol has been specifically designed for routing purposes within the context of IoT devices, operating at the network layer. The exploitation of the RPL protocol poses a threat to IoT networks and has the potential to substantially affect network performance. This article introduces the STACK project, which aims to improve IoT transmission capabilities, identify and mitigate attacks using performance and interference monitoring, and use methods tightly integrated with an intelligent edge.

Index terms: RPL, Contiki, COOJA, Security challenges, IoT

1. Introduction

The Internet of Things, also known as IoT, is an expansive field of technology and study, a portion of which is made up of Low-power and Lossy Networks, or LLNs for short [1]. Due to the fact that the nodes that make up such networks are vulnerable to a variety of restrictions and problems, the currently used routing protocols are inadequate [2].

The maturity of RPL has been demonstrated in its ability to establish connectivity between IPv6 devices, while exhibiting an acceptable amount of control overhead, even under demanding circumstances such as lossy links, heterogeneous and constrained devices and new security risks [1]. The RPL routing standard has been developed with an elevated level of adaptability, necessitating its customization to meet the particular demands of various applications. Current research endeavors pertaining to RPL are focused on enhancing its energy efficiency through various techniques. Numerous objective functions and metrics have been proposed in previous scholarly works, as a result of this.The parent nodes that are selected are experiencing an excessive burden as a consequence of multiple child nodes being linked with each parent node, which consequently leads to a compromising of these particular nodes [3].

The purpose of this article is to introduce the STACK project (Smart and Resilient IoT Networks against Attacks), which aims to enhance the transmission capabilities of IoT, identify and mitigate attacks through the use of performance and interference monitoring, and employ algorithms that are tightly integrated with an intelligent edge. This manuscript outlines the procedures involved in conducting a series of simulations using the COOJA and Contiki simulator. Additionally, it provides an analysis of the network topology parameters and potential cybersecurity obstacles that may arise.

2. Literature review

Contemporary Internet of Things (IoT) gadgets are typically equipped with low power and lossy networks, rendering conventional routing protocols such as RIP, DSR, and OSPF inapplicable. The RPL protocol is utilized for routing method in intelligent gadgets that are subject to constraints such as restricted memory and energy, as well as reduced processing power [4]. The development of destination-oriented DAG (DODAG) was aimed at achieving a loop-free system and merging towards one destination [5]. Each node in the DODAG graph is assigned a rank number that indicates its location. The rank number additionally serves to calculate the proximity between a node and adjacent ones as well as its location from the root node [6].

The RPL protocol (Fig.1) allows for the categorization of nodes through three distinct methods. The first category of nodes demonstrates host-like behavior; these are referred to as end devices or leaf nodes. The subsequent classification is denoted as router and bears the responsibility of executing the tasks of traffic generation and message transmission. The aforementioned group of nodes may be regarded as a border router, commonly referred to as a downstream node or DODAG root.

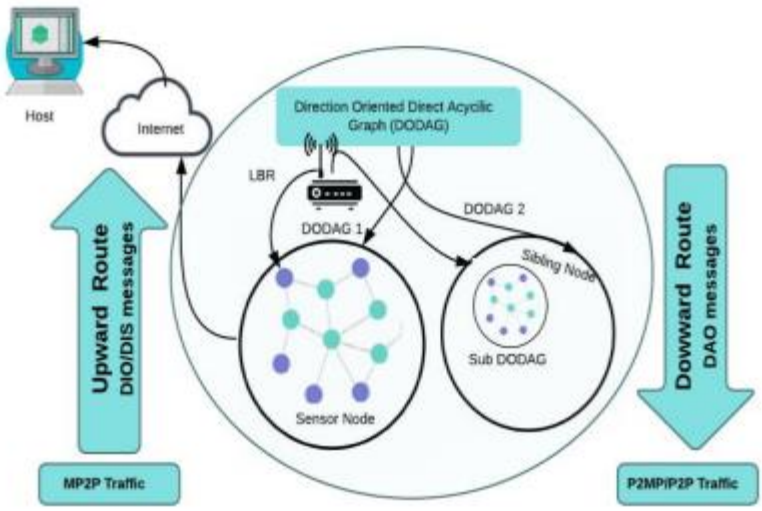

Fig. 1. Concept of RPL Protocol (source: [7])

Numerous attacks on networks have been observed on the RPL protocol, including but not limited to connection failures, processing power limitations, accessibility, network switching, and network structure. The classification of the network layer attacks primarily consists of external attacks and internal attacks [8].

a. Clone ID Attack

It is possible for an assailant to replicate the characteristics of additional nodes, which are commonly referred to as hooked nodes. Consequently, the attacker is able to generate several copies of packets by obtaining the encryption secret and ID of the node, leading to the misrouting of said packets. In general, nodes that act as attackers gather information such as rank ID and other relevant data pertaining to the nodes they target [9].

b. Selective Forwarding Attack

The present assault is initiated through the discriminatory transmission of packets, resulting in significant disturbances within the routing trajectory. This attack has the potential to facilitate the activation of a DDoS attack. The perpetrators discard all network traffic except for control communications, according to the statement [10].

c. Blackhole Attack

The blockhole attack (Fig.2) is characterized by the intentional dropping or blocking of data packets that carry legitimate messages by a malicious node. Instead, the attacker intentionally forwards messages containing misleading data, resulting in increased control overhead and packet delay. A node that seeks to reach its final location may be misled into following the shortest path towards said destination, thereby falling prey to deception. Upon receipt of data packets of data, a loss of service may occur, resulting in failure to deliver the packet to its intended destination. This may also lead to location exploitation. Consequently, there is a deficiency of interaction between the authentic source and destination nodes. The blockhole node is not visually discernible within the network, thus necessitating meticulous monitoring of network traffic. The occurrence of a blackhole attack results in a decline in network performance, manifesting as decreased throughput and routing complications [11].

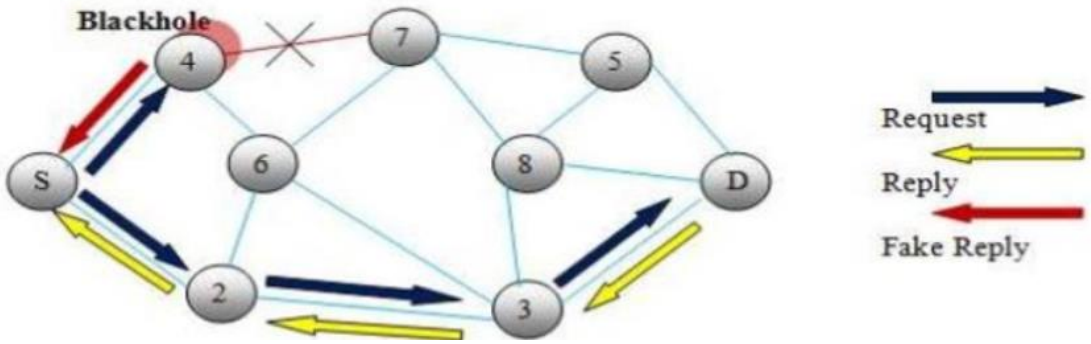

Fig. 2. Black Hole Attack (source: [12])

d. Wormhole Attack

The RPL protocol may be adversely affected by wormhole attacks, which can cause disturbances in both traffic patterns and the structure of networks. The present assault involves the rerouting of all information packets and traffic through a tunnel that has been established by two assailants. There exist two distinct mechanisms that determine how a wormhole may manifest. The encapsulation process involves the reception of a packet in a defined form by the connected or neighboring node, whereby the packet is detached from the payload. Packet relay is a method by which a malevolent node transmits packets to remote nodes that are considered as neighboring nodes [13].

3. Experimentation in Contiki

a. Performance metrics of the RPL network

The Packet Delivery Ratio (PDR) is a metric that quantifies the proportion of messages that are successfully transmitted from the origin node to be received at the root node of the network. It is calculated by dividing the quantity of acquired messages at the root node by the total amount of packets of data delivered through the origin node. The mean Packet Delivery Ratio (PDR) is calculated by aggregating all received and processed transmissions at the base node in the network.

The power consumption of a node is the aggregate amount of power utilized for transmission, acceptance, low power/sleep mode, and data analyzing by the microcontroller for processing.

Latency refers to the amount of duration required for a packet of data to obtain the root network node from its origin node.

b. Contiki and Cooja simulation environment

Contiki is an operating system designed for the Internet of Things (IoT) that is tailored to cater to the requirements of tiny IoT gadgets that have limited memory, power, bandwidth, and processing capabilities.

Contiki facilitates both conventional and contemporary enabling protocols for the Internet of Things (IoT). The uIP protocol is designed for use with IPv4. The present implementation of TCP/IP has the capability to provide support for microcontrollers of both 8-bit and 16-bit. The uIPv6 extension to uIP is a fully compliant implementation of IPv6. The Rime stack serves as a viable option in situations where the utilization of IPv4 or IPv6 is not feasible. The system provides a collection of fundamental components suitable for energy-efficient devices. The acronym 6LoWPAN denotes the implementation of Internet Protocol version 6 over wireless personal area networks with low power consumption. The technology offers compression capabilities to facilitate the utilization of low data rate wireless communication, which is essential for devices that possess restricted resources.

The Routing Protocol for Low-Power and Lossy Networks (RPL) is an IPv6 distance vector protocol that enables the identification of the optimal path in a network of devices with diverse capabilities, particularly those operating in low-power and lossy networks (LLNs). The Constrained Application Protocol (CoAP) facilitates communication for low-power and resource-constrained devices, typically those necessitating extensive remote monitoring [14].

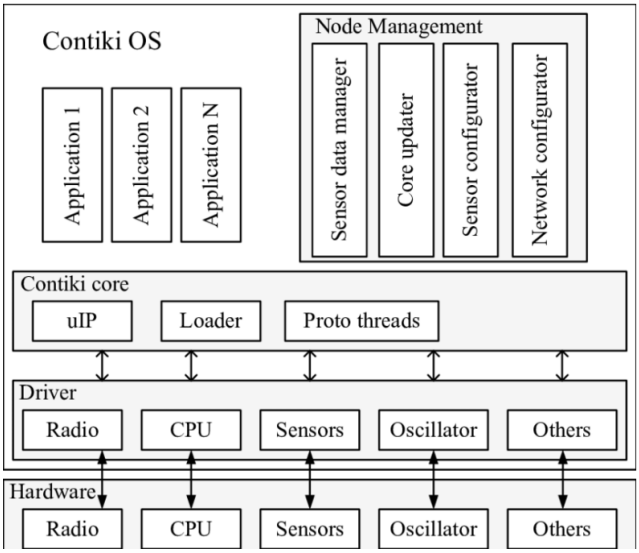

Fig. 3. Architecture of the OS Contiki (source: [15])

As previously discussed, Contiki is a minimalistic operating system that has been primarily designed for wireless nodes. Contiki's developed algorithms provide numerous benefits. The software platform known as Contiki offers a simulator named Cooja, which is based on the Java programming language and is utilized for the purpose of simulating wireless sensors. The Cooja simulator exhibits a higher degree of flexibility, as numerous components of the simulator are amenable to replacement and extension. The replaceability of certain components within the simulator, such as the simulated node hardware, extensions and broadcasting circumstances, represents a notable feature. Cooja is characterized by its scalability, efficiency, extensibility, and flexibility. The Contiki Cooja Wireless Sensor Network Simulator is primarily utilized for simulating numerous wireless scenarios [16].

c. The implementation and setup of RPL

Upon completion of the software technology installation procedure, the user inputs a command `cd contiki/tools/cooja/ant run` into the terminal, which triggers the emergence of an initial window, thereby enabling the commencement of the simulation process in Cooja.

```
robert@Ubuntu:~/contiki/tools/mspsim$ ant run
Buildfile: /home/robert/contiki/tools/mspsim/build.xml

init:
    [mkdir] Created dir: /home/robert/contiki/tools/mspsim/build

compile:
    [javac] Compiling 242 source files to /home/robert/contiki/tools/mspsim/build
    [javac] warning: [options] bootstrap class path not set in conjunction with -so
urce 7
    [javac] warning: [options] source value 7 is obsolete and will be removed in a
future release
    [javac] warning: [options] target value 7 is obsolete and will be removed in a
future release
    [javac] warning: [options] To suppress warnings about obsolete options, use -Xl
int:-options.
    [javac] /home/robert/contiki/tools/mspsim/se/sics/mspsim/Main.java:52: warning:
 [deprecation] newInstance() in Class has been deprecated
    [javac]       return nodeClass.newInstance();
    [javac]                       ^
```

Fig. 4. The command prompt following the installation of Contiki

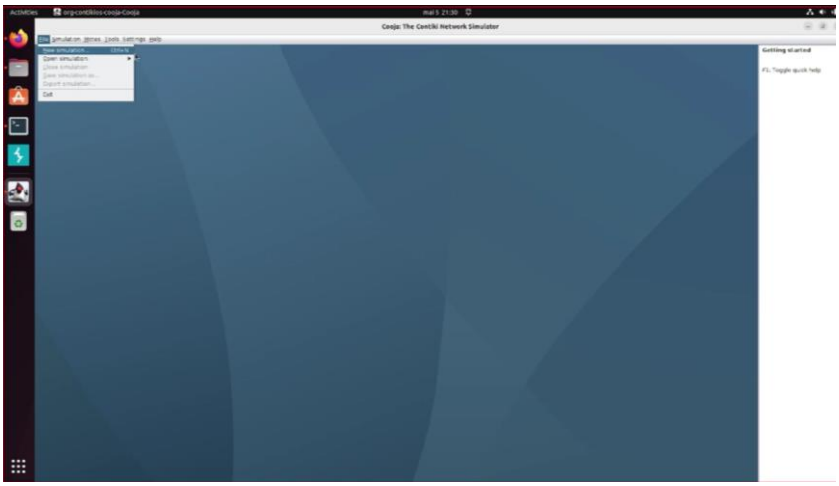

Fig. 5. A new Cooja simulation window

To initiate a new simulation, please proceed to the STACK simulation and select the option to create it.

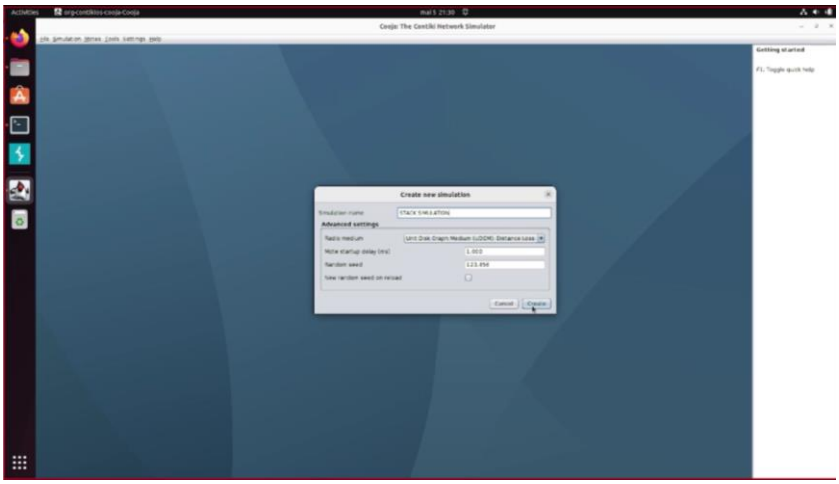

Fig. 6. The initial step in commencing a Cooja simulation

In the event that the Network Window lacks Grids, one may navigate to the view menu and select the 10m background grid from the options presented. Now to add a Sky mote, access the Motes menu, select the option to Add motes, proceed to click on Create new mote type, and ultimately opt for the Sky mote alternative. The Sky Mote option was selected for the purpose of emulating Tmote Sky motes.

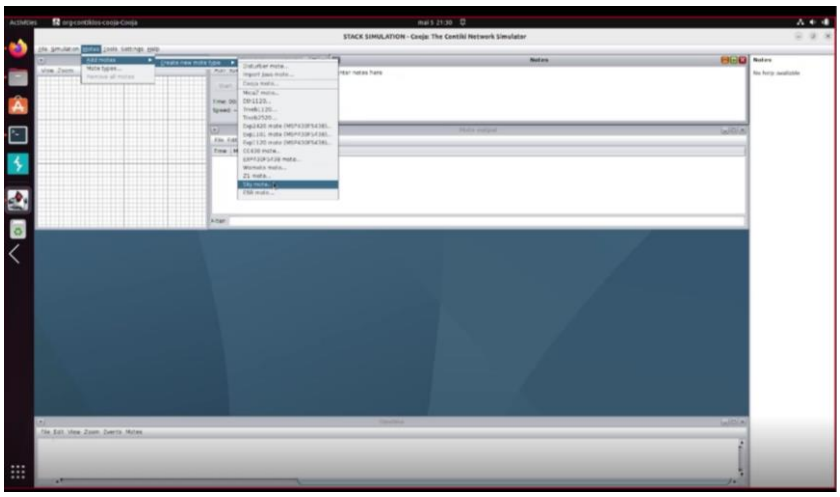

Fig. 7. Configuring parameters

To initiate the creation of a Mote type window, please select the option to browse. After this step, navigate to the directory *path /home/user/contiki/examples/ipv6/simple-udp-rpl*.

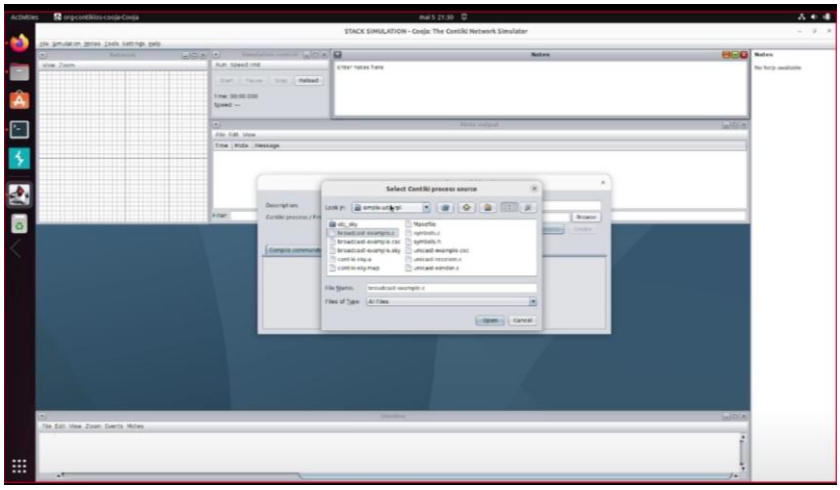

Fig. 8. The process of preparing for compilation

Select the file named *broadcast-example.c* and click on the Open button. To proceed, navigate back to the previously accessed Menu, select the option to compile, allow for a period of time to elapse, and ultimately initiate the process by selecting the Create button.

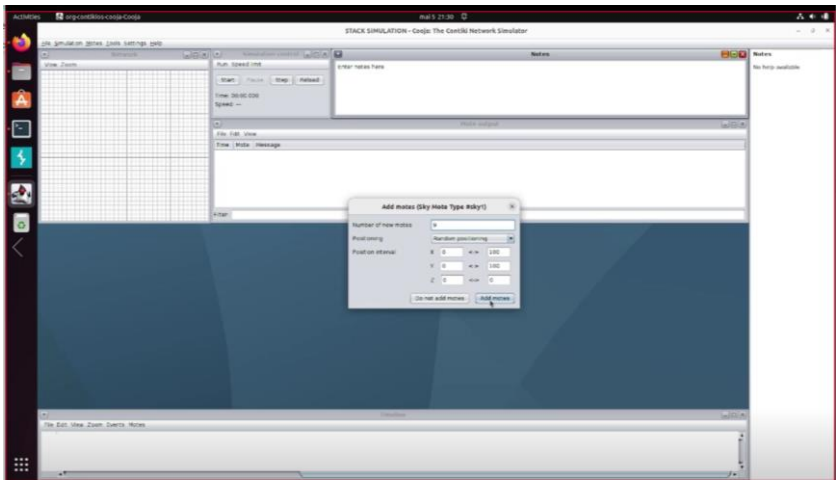

Fig. 9. Determining the quantity of nodes

A prompt will be displayed to add notes in a new window. The next step is to choose multiple options and click on the "Add Notes" button. To distinguish among these motes, access the View Menu located in the Network Window and select the Mote IDs option. Each mote will be assigned a numerical value.

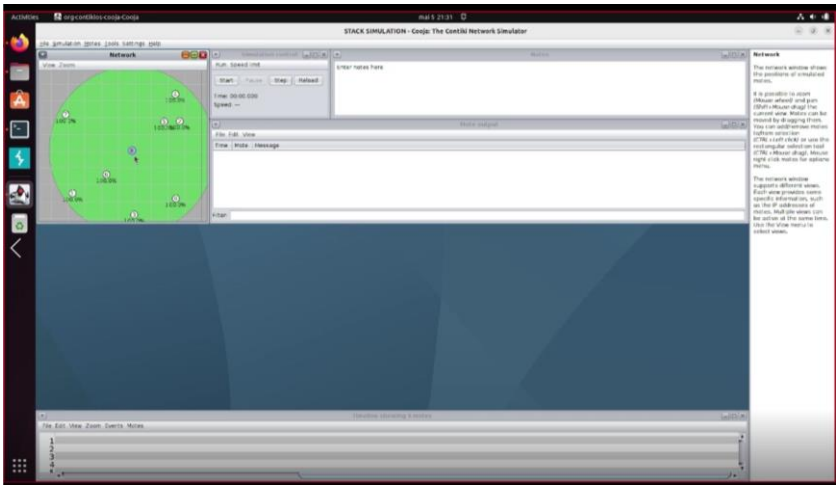

Fig. 10. The practice of monitoring network radio traffic

To monitor the radio traffic, navigate to the View Menu and select the option for Radio Traffic. Depress the Start button to initiate the simulation. By pressing the "Pause" button, the user can observe the interactions within the network through the Network Monitoring Window. The Mote results display will display a printout of the simulated motes. The conclusion of our simulation has been reached.

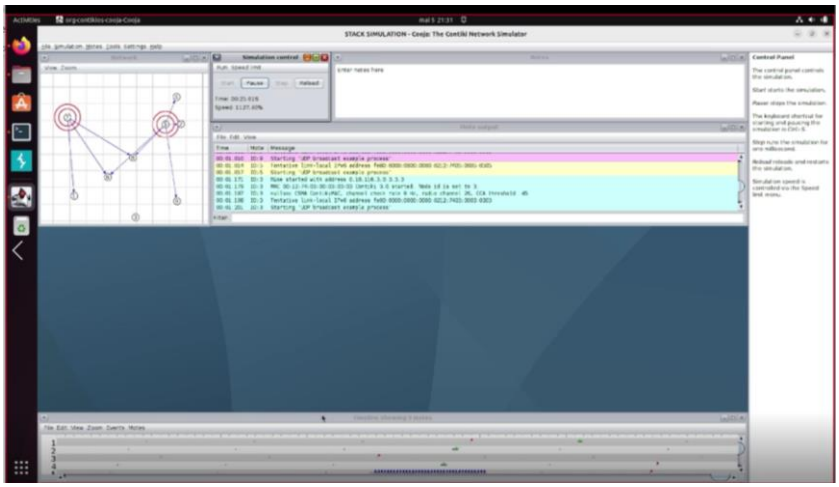

Fig. 11.

4. STACK Project

The STACK project, part of the ITEA initiative, aims to facilitate the provision of high Quality of Service (QoS) for Internet of Things (IoT) applications, even in situations that are not benign, by making them resistant to attacks. The objectives encompass enhancing the transmission capabilities of IoT, detecting and mitigating attacks through accomplishment and interference tracking, and employing algorithms that leverage a closely integrated smart edge [17].

4.1. Approach methodology for addressing the challenge

The absence of assured reliability, delay, and privacy in numerous IoT devices is a significant apprehension, particularly in light of the escalating incidence of security breaches. The vulnerability of IoT mesh networks comprising devices that are embedded is primarily attributed to their wireless communication and relatively low output power. Despite these limitations, their impact on critical domains such as autonomous vehicles and healthcare is increasingly significant, thereby posing a significant threat to our security and livelihood [18]. The issue in ensuring the performance of Internet of Things (IoT) networks during challenging circumstances, such as incidents and cross-technology interference, lies in the limitations imposed by resource constraints that avoid the implementation of advanced defenses on machines.

4.2. Possible project outcomes and impact

STACK provides a variety of innovative solutions to address this difficulty. Given the need of both recognition of attacks and prevention in non-benign circumstances, the initiative aims to leverage the computational capabilities of the smart edge.

The acquisition of requisite training data and the development of novel defenses will be facilitated through the utilization of deployments and testbeds. The data can be utilized to develop compressed models that can be deployed on IoT devices or gateways [18]. The proposed approach for mitigating new attacks involves leveraging frequency, data rate, and protocol variety. This strategy aims to ensure quality of service (QoS) levels and communication prioritization in the event of an

5. Conclusion

The current investigation introduces the routing protocol for low power consumption and lossy networks (RPL) specifically developed for wireless sensor networks. This study aims to provide a comprehensive understanding of the operational mechanisms of RPL, that lacks a predetermined standard for its security operations, necessitating the standardization of security operation protocols by researchers. It outlines the step-by-step configuration of RPL in the Cooja simulation environment, while also discussing potential cybersecurity-related obstacles that could impact the routing protocol for low-power wireless networks that are prone to packet loss. lacks a predetermined standard for its security operations, necessitating the standardization of security operation protocols by researchers.

References

[1]. George Simoglou, George Violettas, Sophia Petridou, Lefteris Mamatas, Intrusion detection systems for RPL security: A comparative analysis, Computers & Security, Volume 104, 2021,102219, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2021.102219.

[2]. Olfa Gaddour, Anis Koubâa, Mohamed Abid,Quality-of-service aware routing for static and mobile IPv6-based low-power and lossy sensor networks using RPL, Ad Hoc Networks, Volume 33, 2015, Pages 233-256, ISSN 1570-8705, https://doi.org/10.1016/j.adhoc.2015.05.009.

[3]. Qasem M, Al-Dubai A, Romdhani I, Ghaleb B, Gharibi W, "A new efficient objective function for routing in internet of things paradigm", in Standards for Communications and Networking (CSCN), 2016 IEEE Conference on 2016 Oct 31 (pp. 1-6). IEEE.

[4]. S. Y. Hashemi and F. Shams Aliee, "Dynamic and comprehensive trust model for IoT and its integration into RPL," J. Supercomput., vol. 75, no. 7, pp. 3555–3584, 2019.

[5]. A. E. Hassani, A. Sahel, A. Badri, and E. M. Ilham, “A hybrid objective function with empirical stability aware to improve RPL for IoT applications,” Int. J. Electr. Comput. Eng., vol. 11, no. 3, pp. 2350– 2359, 2021.

[6]. Z. A. Almusaylim, N. Z. Jhanjhi, and A. Alhumam, “Detection and mitigation of RPL rank and version number attacks in the internet of things: SRPL-RP,” Sensors (Switzerland), vol. 20, no. 21, pp. 1–25, 2020.

[7]. A. Jahangeer, S. U. Bazai, S. Aslam, S. Marjan, M. Anas and S. H. Hashemi, "A Review on the Security of IoT Networks: From Network Layer’s Perspective," in IEEE Access, doi: 10.1109/ACCESS.2023.3246180.

[8]. W. Fang, W. Zhang, W. Chen, T. Pan, Y. Ni, and Y. Yang, “TrustBased Attack and Defense in Wireless Sensor Networks: A Survey,” Wirel. Commun. Mob. Comput., vol. 2020, 2020.

[9]. S. R. Taghanaki, S. B. Arzandeh, and A. Bohlooli, “A Decentralized Method for Detecting Clone ID Attacks on the Internet of Things,” Proc. 2021 5th Int. Conf. Internet Things Appl. IoT 2021, 2021.

[10]. Q. Zhang and W. Zhang, “Accurate detection of selective forwarding attack in wireless sensor networks,” Int. J. Distrib. Sens. Networks, vol. 15, no. 1, 2019.

[11]. Muneer Bani Yassein, I. Hmeidi, Y. Khamayseh, M. Al-Rousan, and D. Arrabi, “Black Hole Attack Security Issues, Challenges & Solution in Manet,” no. April 2019, pp. 199–207, 2018.

[12]. Gulzar, C.M., Kurnool, & Kashyap, R. (2015). Prevention of Black Hole Attack in MANET.

[13]. O. R. Ahutu and H. El-Ocla, “Centralized Routing Protocol for Detecting Wormhole Attacks in Wireless Sensor Networks,” IEEE Access, vol. 8, pp. 63270–63282, 2020.

[14]. https://www.tutorialspoint.com/internet_of_things/internet_of_things_contiki.htm accessed at March 20th, 2023.

[15]. Derogarian, Fardin. (2015). Design of a Body Sensor Network Embedded in Textiles for Biomedical Applications. 10.13140/RG.2.1.1192.3920.

[16]. https://phdinfo.org/contiki%20cooja%20wsn%20simulator.html accessed at April 3rd, 2023.

[17]. File STACK%20Project%20profile%20leaflet.pdf accessed at April 23rd, 2023.

[18]. https://agile.ro/stack/about/ accessed at April 23rd, 2023.