ML-Based mitigation of UDP flooding attacks in IoT

A DISSERTATION

SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE

OF

MASTER OF TECHNOLOGY

In

INFORMATION SYSTEMS

Submitted By

Pragya Sahu

2K16/ISY/08

Under the supervision of

Ms. Anamika Chauhan

DEPARTMENT OF INFORMATION SYSTEM

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly DELHI COLLEGE OF ENGINEERING)

Bawana Road, Delhi-110042

MAY 2018

CERTIFICATE

This is to certify that the dissertation entitled "ML-Based mitigation of UDP flooding attacks in IoT" has been submitted by Pragya Sahu (Roll Number: 2K16/ISY/08), in partial fulfilment of the requirements for the award of Master of Technology degree in Information System at Delhi Technological University. This work is carried out by her under my supervision and has not been submitted earlier for the award of any degree or diploma in any university to the best of my knowledge.

Place: Delhi

Date**:**

Ms. Anamika Chauhan

SUPERVISER

Assistant Professor

Department of Information Technology

Delhi Technological University

Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I am very thankful to my major project guide Ms. Anamika Chauhan, Assistant Professor in Department of Information Technology at Delhi Technological University, Delhi for the valuable support and guidance she provided in formation of this major project. It is my pleasure to record my sincere thanks to my respected guide for her constructive criticism, interminable encouragement and valuable insight without which the project would not have shaped up as it has.

I would also like to express my gratitude to the university for providing us with the laboratories, infrastructure, testing facilities and environment which allowed us to work without any obstructions.

I would also like to appreciate the support provided by our lab assistants, seniors and our peer group who aided us with all the knowledge they had regarding various topics.

I would like to thank My Parents Ms. Rameshwar Prasad Sahu (Father) and Mrs. Usha Sahu (Mother), Mr. and Mrs. Anurag Sahu, and Mr. and Mrs. Riturag Sahu, friends, family, and everyone who helped me completing my Research work.

Place: Delhi PRAGYA SAHU

Date: 2K16/ISY/08

ABSTRACT

Internet of Things (IoT) is the concept that is directing a huge increase in the Internet and its capability to collect, investigate and distribute data which can be turned into information or knowledge. One category of devices such as the Low Power Lossy networks (LLNs) consists of numerous small sensors and low power devices as building block elements in IoT. The ROLL working group at Internet Engineering Task Force (IEFT) has designed the Routing Protocol for LLNs (RPL) which is the core of the IoT protocol stack used for communication between these low-power devices. IoT devices are resource constrained devices and hence it is very easy to exhaust them of their resources or deny availability. One of the most prominent attacks on the availability is the Denial of service (DoS) attack. Although, DoS is not a new Internet attack but in the recent times of 2017-18 even bigger DoS attacks took place. A few simulation tools exist which enable evaluation of RPL for a realistic deployment scenario. This paper focuses on understanding of the implementation of UDP flood attacks on RPL then recognizing the attacker nodes using statistical based approach for detecting Local Outlier. The implementation and analysis is carried out by the simulations done in the Contiki OS Cooja simulator with respect to the performance metrics such as radio duty cycle, energy consumption. Furthermore, in this paper, an ML based approach is proposed that could be used recognize the attacker to mitigate UDP flood attacks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

1.1. IOT

A ton of smart gadgets are interconnected to the Cyberspace today. There is an exponential development in the quantity of smart gadgets interconnected through the portable web. The gadgets are installed with knowledge because of the fast improvements in sensors and other handling equipment innovation. These brilliant gadgets are equipped for speaking with people and other savvy gadgets, too. This makes ready to the advancement of the expression "Cyberspace of Things" (IoT).

IoT is authored by Kevin Ashton with regards to store network administration in the year 1999 that portrays an innovation without bounds in view of the Cyberspace that includes sharing of data [1]. IoT or the IP-connected IoT is a heterogeneous network. It consists of the convention Cyberspace and networks of devices connected together using IP convention. 6LoWPAN networks are the networks of constrained devices in the IoT.

The Cyberspace of Things or entirely the IP-associated IoT is a heterogeneous system that comprises of the ordinary Cyberspace and systems of compelled gadgets associated together utilizing IP convention.

Things in the IoT are extraordinarily notable articles that sense the materialistic condition and the host gadgets and impart this data to the Cyberspace.

An IoT device (a thing) can be a light, an indoor regulator, a home apparatus, a stock thing, a cell phone, a PC, or any event. IPv6 with its conceivably boundless address space can associate trillions of gadgets with the IoT [2].

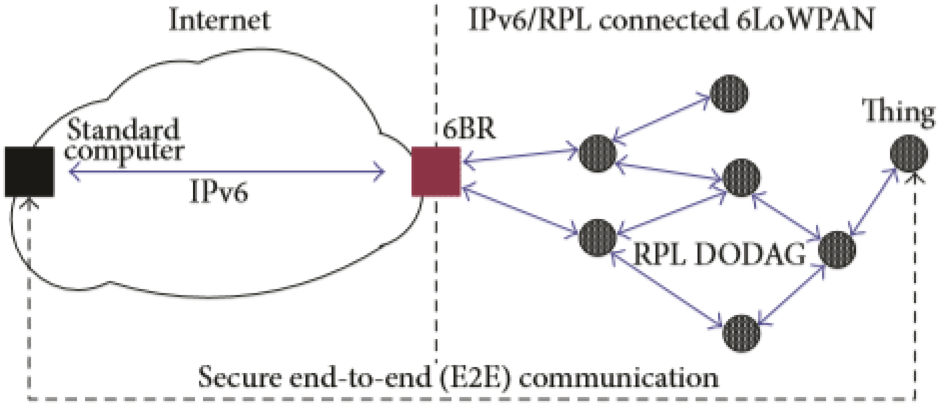

Internet
IPv6/RPL connected 6LoWPAN
Standard computer
IPv6
6BR
Thing
RPL DODAG
Secure end-to-end (E2E) communication

Figure 1. 1: An IoT scenario that shows an inter connection of Cyberspace and IPv6/RPL connected things in a 6LoWPAN.

The IoT is alluded as the interrelationship of materialistic items that have an IP address for web availability. As the quantity of gadgets associated are expanding, the correspondence between these gadgets is impossible just by the Cyberspace. Consequently plans in IoT empower selective nearby correspondence between different gadgets. The conventions and guidelines utilized as a part of IoT will be not the same as the benchmarks utilized as a part of the present Cyberspace.

The most recent innovative improvement in Cyberspace and gadgets, for example, tablets, workstations, advanced mobile phones and computerization machines are the principle apparatuses for the execution of IoT applications [3].

1.1.1 6LoWPAN

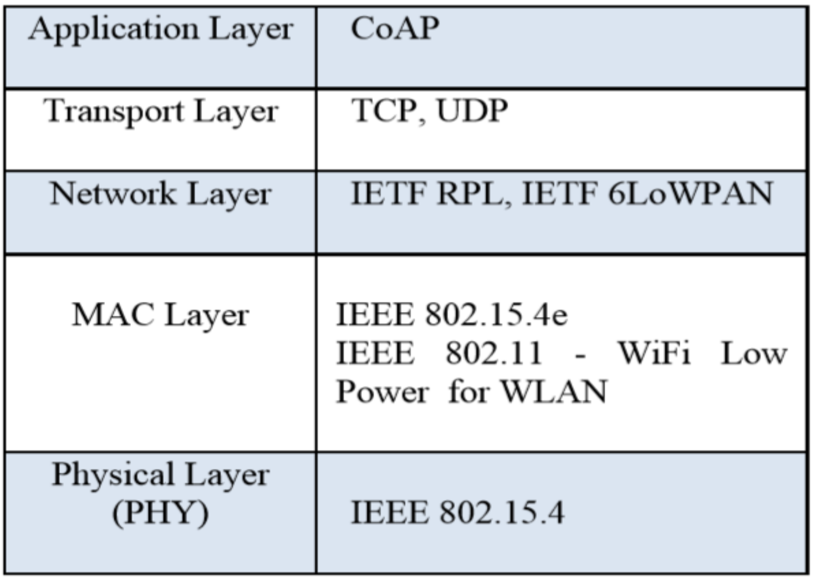
6LoWPAN is a minimal power and minimal cost arrangement which associates asset compelled remote gadgets, utilizing compacted Cyberspace Convention variant IPv6.6LoWPAN characterizes IPv6 header pressure and determines how packets are steered in wireless systems.

6LoWPAN systems bolster multi hop correspondence where hubs forward packets in the interest of different hubs. Power or Energy is one of the rare assets in 6LoWPAN systems, and normally the greater part of the energy is expended on listening idly; accordingly, 6LoWPAN systems are typically Duty cycled implying that the radio is off mostly and is turned on just for a brief timeframe for purpose of listening. Because of worldwide IP availability, 6LoWPAN systems are powerless against the majority of the invasions against IoT in addition to invasions starting from the Cyberspace. Due to the remote medium and generally unattended arrangements, it is less demanding to compromise 6LoWPAN gadgets than run on the Cyberspace. This escalates into new dangers against the Cyberspace [2].

1.1.2 ROUTING IN IOT

Routing or directing is an essential factor impacting interconnection amongst gadgets and execution of data trade. Routing is the essential procedure for IoT. The Routing convention and quality it possess in execution enhances the attainment of the LLN. The attainment measurements for assessing the convention incorporate Power consumption, PDR known as Packet Delivery Ratio and Latent period.

Low power remote gadgets are the principle part of IoT. IoT convention stacks is seeing an alternate course of development when contrasted with the general TCP/IP stack. The Convention Stack for Low Power Lossy Networks is portrayed in Figure 1.2.



Application Layer	CoAP
Transport Layer	TCP, UDP
Network Layer	IETF RPL, IETF 6LoWPAN
MAC Layer	IEEE 802.15.4e IEEE 802.11 - WiFi Low Power for WLAN
Physical Layer (PHY)	IEEE 802.15.4

Figure 1.2: Convention stack for Low Power Lossy Networks

1.2. INTRODUCTION TO RPL:

IETF turned out with new conventions for the IoT that is the Routing Convention for Low Power and Lossy Networks named as RPL [4].

RPL is an institutionalized routing convention for the IoT. RPL implements IoT to the real world. RPL is principally utilized as a part of a 6LoWPAN system. RPL makes a destination- oriented directed acyclic graph also known as DODAG between the hubs in a 6LoWPAN. It bolsters one way movement approaching the root and both way movement between 6LoWPAN gadgets and among gadgets and root. There occurs various worldwide RPL examples for solitary 6LoWPAN system, and a nearby RPL DODAG can be made among an arrangement of hubs inside a worldwide DODAG.

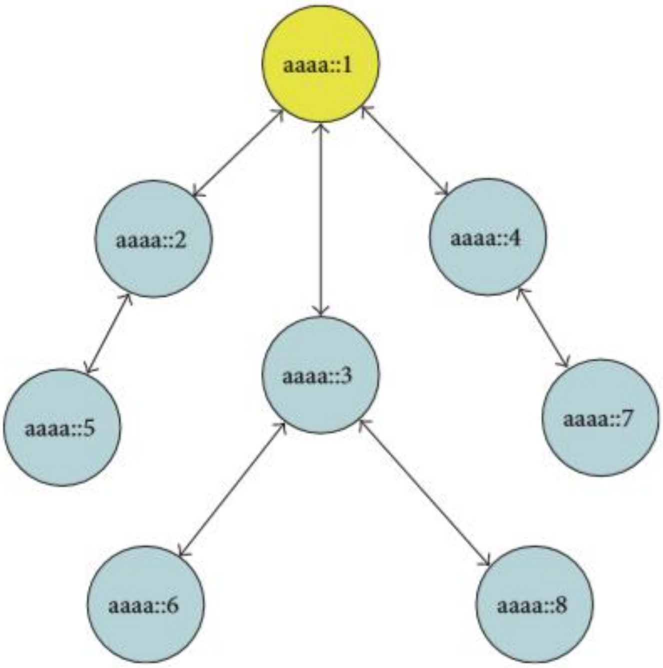


Figure 1. 3: A simple RPL DODAG

In Figure 1.3 a RPL DODAG is indicated where every hub has a hub ID, a rundown of neighbours, and a parent hub. Every hub in a DODAG has a rank that shows the location of a hubs in respect to different hubs and as for the root. Ranks entirely diminish in the up heading towards root and entirely increment from the DODAG root towards hubs. Source routing implies every packet contains the route, packet should take through the system. In storing Mode root keeps the data about every hub in the system. In a non-storing mode, all sending hubs must keep up in-organize directing tables so that they know where to forward packets.

1.3. DOS Invasions:

A DoS also known as denial of service invasion is a purposed attempt to prevent genuine users from reaching a specific network resource. This kind invasions have been known since the mid1980s [5].

A DoS invasion is a digitated invasion in which culprit looks to make a gadget or its asset unattainable to its expected person by shortly or indefinitely disrupting services of the host connected to the Cyberspace. Denial of service is proficient by flooding intended gadget or its asset with vain demands to burden system and prevent some or all legitimate requests from being fulfilled [6].

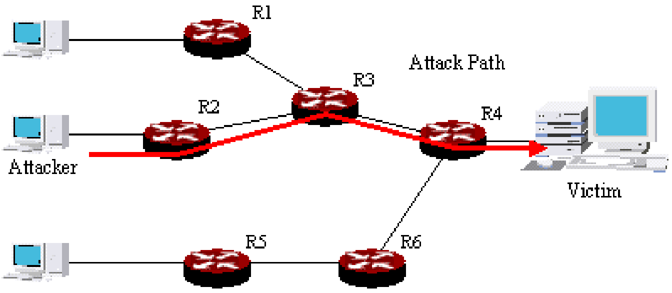


Figure1. 4: DOS Invasion

In 2017 GitHub, a popular website, confronted the world's most effective DDoS invasion. According to GitHub, the site was unattainable for around 5 minutes. It happened on February 28th.

Not long after the invasion, inside around 10 minutes, GitHub looked for assistance from Akamai Prolexic, which is a DDoS mitigation service. To hinder the noxious packets, Akamai steered all the activity through its scrubbing centres.

According to Akamai, the programmers could drive the invasion to around 126.9 million packets for every second. The invasion was more than double the extent of the September 2016 invasions that was a consequence of Mirai botnet.

This DDoS invasion came about because of "memcached servers," which are utilized to store data and decrease the load because of memory intensive services. Huge numbers of these servers are uncovered on the web, and anybody can look for them.

Different hubs which goes as intermediate hubs in directing procedure additionally endure their battery and assets likewise get traded off, for example, AODV, Bellman portage, DSR are purposed for perfect conditions yet they are helpless against these invasions so make them mindful of these invasion we require some security system which can analyse and keep these invasions utilizing remaining vitality of the hubs.

1.3.1. Types of DOS Invasions:

a) Volume Based Invasion: These invasions utilize enormous traffic saturating the bandwidth of the sufferer. Volumetric invasions are anything but difficult to create by utilizing employing simple strengthening techniques after that hubs won't have the capacity to perform to their abilities. With a specific end goal to make huge activity UDP flooding, TCP flooding or other caricature bundle flooding can be utilized and the sheer movement created by these invasion can hinder the entrance to the end client.

b) Convention Based Invasion: Convention based invasion essentially deserted an objective in-open by abusing a shortcoming in the Layer 3 and Layer 4 convention stack. It is otherwise called state weariness invasion, this is a condition.

1.3.2 Common Denial of Service Invasions:

a) Buffer Overflow:

An event where the data interchanged to a buffer oversteps the capacity limit of the support and a portion of the data. Into another buffer, one that the data was not expected to go into. Evil hackers launch buffer overflow invasions wherein data with guidelines to corrupt a system are intentionally built into a document in full learning that the data will flood a buffer and discharge the directions into the computer.

b) Ping of Death:

A sort of invasion where assailant forwards a ping demand which is bigger than 65,536 bytes, which is the greatest size that IP permits. TCP/IP enables a bundle to be divided, basically part the bundle into littler portions that are inevitably reassembled. Assailant exploited this imperfection by dividing packets that when gotten would add up to more than the permitted number of bytes and successfully purpose a support over-burden on the working computer.

c) Smurf Invasion:

A smurf assailant sends PING solicitations to an Cyberspace communicate address. These are exceptional tends to that communicate every single got message to the hosts associated with the subnet. Each broadcast address can bolster up to 255 hosts, so a solitary PING request can be increased 255 times. The arrival address of the demand itself is spoofed to be the address of the assailant's sufferer. Every one of the hosts accepting the PING ask for answer to this current assailant's address rather than the genuine sender's address.

1.4 Outlier Analysis:

Definition of Outlier?

Outlier, otherwise called anomaly, initially comes from statistics [7]. Two established meanings of outlier as described by Hawkins [8] and Barnett and Lewis [9] are: "An outlier is an observation, which deviates so much from other observations as to arouse suspicions that it was generated by a different procedure " and " an outlier is an observation (or subset of observations) which appears to be inconsistent with the remainder of that set of data."

B. Types of Outliers:

Based upon the data utilized for outlier detection, outlier might be either global or local.

1) **Local Outliers:** Local exceptions are distinguished at singular sensor hubs, methods for identifying local outlier spare correspondence burden and improve the adaptability.
2) **Global Outliers:** Global exceptions are recognized in a global potential of view. They are quite compelling since experts might want to have a superior comprehension of general data qualities in IoT. Contingent upon the system design, the ID of global exception can be acted at various levels in the system.

C. Identity of Outliers:

Mainly 3 causes of outliers happened in IoT: (1) Disturbances and faults, (2) incidents, and (3) evil invasions. The kind of anomalies due to evil invasions is worried about the issue of system security. For outliers came about because of various causes, anomaly identification systems are wanted to indicate the personality of these anomalies and arrangement promote with them.

D. Outlier Detection technique for IoT:

There are mainly four Techniques for Outlier detection:

A. Statistical-Based Approaches

B. Nearest Neighbour-Based Approaches

C. Clustering-Based Approaches

D. Classification-Based Approaches

1.4.1 Statistical-Based Approaches:

Statistical-based methodologies are the most punctual ways to handle the outlier detection. These systems are basically show based strategies. They accept or appraise a factual model (likelihood dispersion) demonstrate which catches the circulation of the data and assess data occurrences as for if they fit the model nicely or not. A data example is pronounced as an anomaly if the likelihood of the data occurrence to be produced by this model is down. They are helpful in light of the fact that the overhead with these methods is least.

Mainly there are 3 methods for detecting outliers:

- Mean and Std. Deviation Method
- Median and Median Absolute Deviation Method (MAD)
- Median and IQD also known as Interquartile Deviation Methods

For each situation, the distinction is computed between chronicled data focuses and values ascertained by the different gauging techniques. These distinctions are called residuals. They can be certain or negative contingent upon whether the verifiable esteem is more prominent than or not as much as the smoothed esteem. Different measurements are then figured on the residuals and these are utilized to recognize and screen exceptions.

Mean and Standard Deviation Method:

In this anomaly recognition technique, the mean and standard deviation of the residuals are figured and looked at. In the event that an esteem is a sure number of standard deviations from the mean, that data point is distinguished as an exception. The predetermined number of standard deviations is known as the edge. The default esteem is 2.68.

This strategy has a confinement in identifying exceptions in light of the fact that the anomalies expands the standard deviation. The more outrageous the exception, the more the standard deviation is influenced. Standard deviation and certainty interim both are utilized as a part of factual investigation.

Standard deviation:

$$6 = \frac{1}{N}\sum_{i=1}^{N}(x-\mu)^2$$

(Equation 1.1)

Where, N is population

μ is population mean

x_i is an element from population

MIN= Min. value from the Data set

MAX= Max. value from the Data set

RANGE= Difference between minimum and maximum

1.5 Simulation Tool:

The assessment of accomplishment of the RPL tradition is performed in Cooja test system. The convention is actualized in Java. The simulation was completed for various time interims and 32 RPL hub including one sink. The reproduction involves RPL root hub and receiver hubs which are imitated as Tmote sky receiver hubs that are produced from Cooja and uIPv6 module.

Cooja is proved to be one of the most suitable tool for the simulation of RPL convention of IoT, but it also has limitations in its use. This is especially apropos with respect to the absence of documentation accessible. The Contiki site might be a first port of bring concerning Cooja, and gives a picture of Instant Contiki which would then be able to be utilized with the virtualisation apparatus VMware [10].

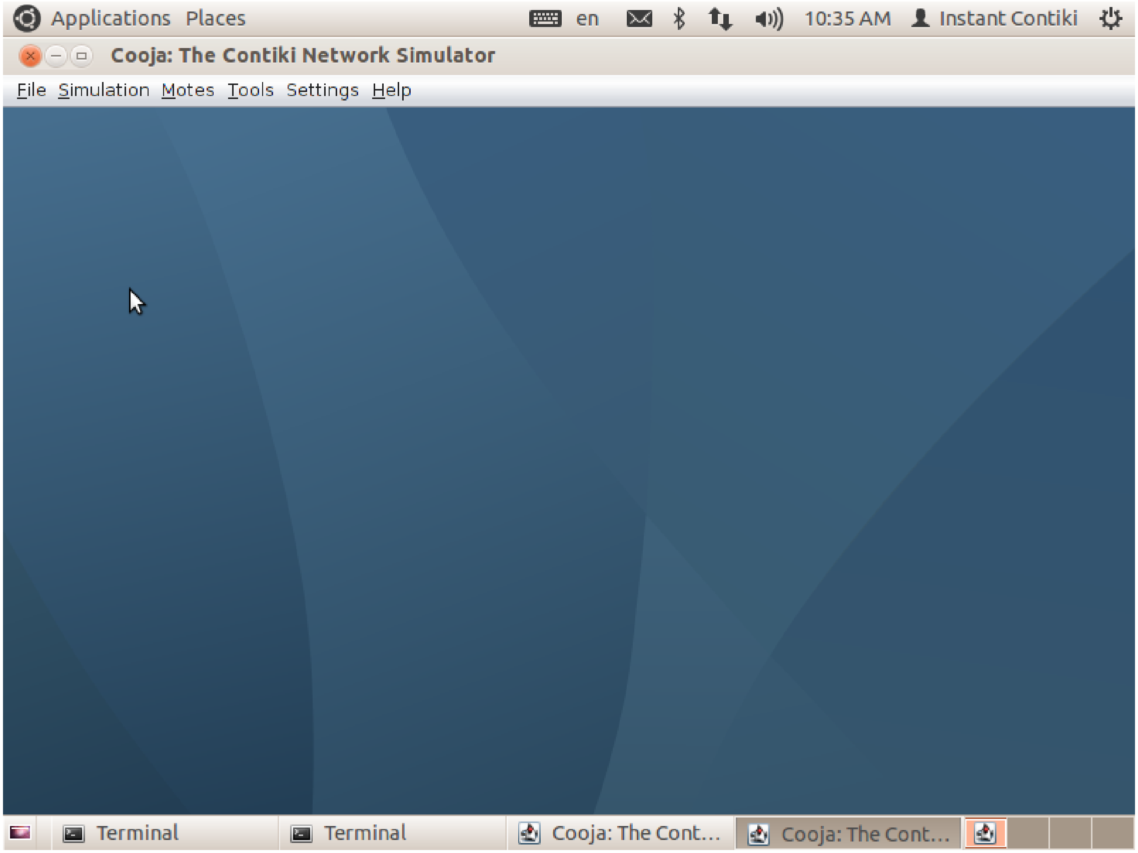


Figure 1. 5: Starting Page of Cooja simulator

CHAPTER 2

LITERATURE REVIEW

This chapter includes the researches and studies which helped to identify the problem existing in the Cyberspace of Things:

2.1 Review of Research Work

2.1.1. IOT, LLNs and RPL

As the associated gadgets are expanding in number, the correspondence between these gadgets is impossible just by the Cyberspace. In this manner outlines in IoT empower select nearby correspondence between different gadgets. The conventions and standards utilized as a part of IoT will be unique in relation to the principles utilized as a part of the present Cyberspace. Low power remote gadgets are the fundamental piece of IoT. IoT convention stacks is seeing an alternate course of advancement when contrasted with the normal TCP/IP stack. Web Engineering Task Force (IETF) has turned out with new conventions for the IoT.

LLN switches consistently work with prerequisites on control utilization, memory, and vitality (battery control). Their associated gadgets are depicted by high hardship rates, low data rates, and precariousness. LLNs are incorporated anything from two or three dozen to thousands of switches. Bolstered movement streams fuse point-to-point (between contraptions inside the LLN), point-to-multipoint (from a central control point to a subset of devices inside the LLN), and multipoint-to-point (from devices inside the LLN towards a central control point) [11].

The Trickle calculation is utilized to control when DIO messages are sent, and it depends on a clock whose term is multiplied each time it is fired, sending less messages per unit of time when the system is stable [13].

The Trickle estimation [12] is used to control when DIO messages are sent, and it relies upon a clock whose term is increased each time it is fired, sending less messages per unit of time when the framework is steady [13].

2.1.2. Security of RPL

As said by Kamaldeep et. Al [4], in a comparative Study on RPL Invasions IoT gadgets are asset obliged gadgets and subsequently it is anything but difficult to deplete them of their assets or deny accessibility. While the IoT innovation acquires the invasions of the customary Cyberspace, the rise of new conventions particularly for IoT increase to the quantity of conceivable invasions. DoS invasion is the invasions that has been a noteworthy worry in the traditional Cyberspace security and IoT. This invasions intends to devour the assets of a remote host or system, in this manner denying or corrupting administrations to real clients.

The Security dangers respect to Cyberspace of Things by Smitesh Mangelkar et. Al [14], in "A Comparative Study on RPL Invasions and Security" RPL accompanies worked in security modes, which are insufficient to relieve a wide range of invasions.

In RFC7416, Tsao et. al. [15], advised a security structure breaking security of RPL. They devised a game plan of security recommendations. Threat Causes and Classification of Threats and Invasions [15] are illuminated in detail as takes after:

2.1.2.1) Threat Causes

Threat cause is an enemy that purposely invasions the system, and in view of the invasion examples, place and capacity of assailant counterattack should be formulated. The assailant are characterized into two gatherings as:

- **Outsiders:** The assailant that are dwelling beyond the system on the web. They can sniff or satire data into the hubs. They are unapproved hubs in the system.

- **Insiders:** The assailant that are the legit hubs from the system. They are hazarded on account of bad configuration, a few flaws, or some materialistic altering of the gadget.

2.1.2.2) Classification of Threats and Invasions

They are mainly characterized into 3 types:

a) Inability to keep directing data secret (Invasions on Confidentiality): The data uncovered may impact the execution of the System, or this data can be used for various purposes. Hubs can imperilled by materialistic adjusting or intrusions did as a result of remote gadget get to be gadget particular. Hence this kind of assaults are out of scope to RPL security.

b) Inability to protect Integrity (Invasions on Integrity): conflicting data can prompt sub optimality or system can be divided into parts. Honesty any misuse which controls the directing data comes under risk space, for example, misrepresentation or replay steering data.

c) Accessibility of a hub can be debilitated in two different ways, (i) disturbance (ii) separation. Possible sufferers can be hubs with great movement. Different sorts of DoS invasions can be utilized to achieve hub or system of hubs unattainability. One way to accomplished it is over-burdening systems utilizing Hello Flooding Invasion.

2.1.3. DOS Invasion

As portrayed by Georgios Loukas et al. [5], the extraordinary assorted variety of DoS invasions has created likewise various assurance proposition from the system security look into network. As a rule an entire protection architecture ought to incorporate the accompanying components:

a) Discovery of nearness of an invasion: Discovery is on three bases (i) error based, (ii) signature-based, (iii) mixture of them. Framework perceives a deviation from the standard conduct of its customers in error based recognition. Signature-based endeavours to distinguish the attributes of known invasion composes.

b) Arranging the approaching bundles in legitimate and faulty: ordinary packets are legitimate and DoS ones are faulty. For identification, anomaly based or signature-based order strategies can be used.

c) Reaction: The security framework can do two things (i) dropping the invasion bundles in a convenient manner (ii) Let them safe via diverting them for a trap for assist assessment.

2.1.4. DOS invasion on RPL

UDP flooding Invasions are anything but difficult to provoke. They require almost no exertion with respect to the assailant. These invasions include overpowering the arbitrary ports of the sufferer machine with the help of directing lot of UDP packets. The sufferer reviews for the application related with packets, and revert with a "Goal Unreachable ICMP (Cyberspace Control Message Convention)"[15].

The way gadgets are all inclusive available makes IoT more vulnerable to this kind of invasions. Gadgets are vulnerable to DDoS invasions from the Cyberspace and additionally within the system. An assailant can over-burden a sufferer with a lot of activity. The absence of a handshake in UDP offers leads to various IP address caricaturing invasions [16].

The selective forwarding invasion, permits propelling DoS at the network layer in RPL. Malevolent hubs specifically direct packets with expectation to disturb directing ways. Names of of some other flooding invasions in RPL are Black hole invasion, Hello Flooding invasion, neighbour discovery etc. [17].

Thus, we conclude that DoS with IoT together is really a hazardous situation.

CHAPTER 3

PROPOSED SYSTEM

In the project, we advised an outlier based analysis to detect analyse and mitigate UDP flooding invasions in IoT internetworks. Our Outlier based procedure says if the Power consumption or Radio Duty cycle % of hub is excessively higher than that of the Average Hub's, it is highly possible that the hub is Disturber hub.

UDP Flooding invasion is implemented in Contiki. So it implements the UIP TCP/IP stack that gives intercommunication abilities to IoT gadgets that are bounded by assets. We exercised and changed the existing rpl-udp example in Contiki to implement UDP flooding invasion.

In this project work we deployed internally hazarded hubs as disturber hubs that have other properties same as the properties of network hubs so that they can't be easily detected and this will increase the impact of invasion as well. If we use external DOS with different assailant here to drain out the batteries of other hubs then easily because they possess different properties than the rest of the network.

The assumptions we made to implement this invasion are:

- The Disturber motee sends multicasting DIS messages to all neighbour RPL hubs. It uses the IPv6 address ff02::1a.
- We have to sniff the messages to or from the existing RPL motes, if the already existing network accepts a particular prefix for DIS message.
- The network created using RPL as routing convention doesn't uses authentication while connecting the network [18].

The measurements of Packet Transmission Ratio (TX), Packet Receiving Ratio (RX), CPU time, Transmission's Low Power Mode (LPM) and CPU were done using the Collect View Tool. Collect View is integrated in Contiki OS in power profiler. With the help of formulae Power Consumption and Radio Duty Cycle% is calculated and graphs are plotted. Then Outlier Analysis is done to identify the assailant and sufferer hubs.

CHAPTER 4

IMPLEMENTATION AND MODEL DESIGN

4.1 Overview

The purpose of this project is to detect DoS assailant hub, to do so we calculate Power consumption and Radio Duty Cycle %. As the DoS invasion doesn’t rely on the vulnerabilities of routing conventions, so it can be deployed in easily without making changes in the convention policies. In the project, we deploy and mitigate the D-dos invasion in Cyberspace of Things.

4.2 IMPLEMENTATION

The complete work has been implemented in 3 major sections, which are:

Section 4.2.1 Creation of IoT scenarios using RPL as routing convention for different topologies and calculation of the Power Consumption and Radio Duty Cycle% of different hubs.

Section 4.2.2 Deployment of flooding DoS Invasion by introducing Disturber hub in RPL Convention Simulation and calculation of the Power Consumption and Radio Duty Cycle% of different hubs.

Section 4.2.3 Detection of DoS assailant using the Statistical Outlier analysis.

4.2.1 Creation of IoT scenarios using RPL as routing convention for different topologies:

The IoT Gadgets are resource constrained, to provide intercommunication abilities to them, Contiki implements the UIP TCP/IP stack. This has been written in C language. The Cooja simulator is the simulation tool used. It is placed at the contiki-2.6/tools/cooja folder. For the front-end interface it uses it uses Java code and at the back end is uses platform specific emulators to carry out the simulations.

Following steps are followed in creating RPL scenario:

- Go to the terminal and Type cd contiki/tools/cooja ant run
- Select **File** menu, and then **New Simulation**, and save it as rpl-Example.
- Click on Motes menu then “Add Motes” then “create new mote type”, and then press the “Sky Mote” button.
- In the Create Mote type Window, type Uhub in the description and click on Browse.
- Locate simple-udp-rpl folder as shown below. /examples/ipv6/simple-udp-rpl
- From this folder select unicast-receiver.c and click on Compile then click Create.
- Select One Mote, 1st mote shows up, now again follow the same steps.
- Add 31 motes.

Below Figure shows the random arrangement of these motes. After Pressing the Start button in Simulation Control window, Traffic is started between these motes.

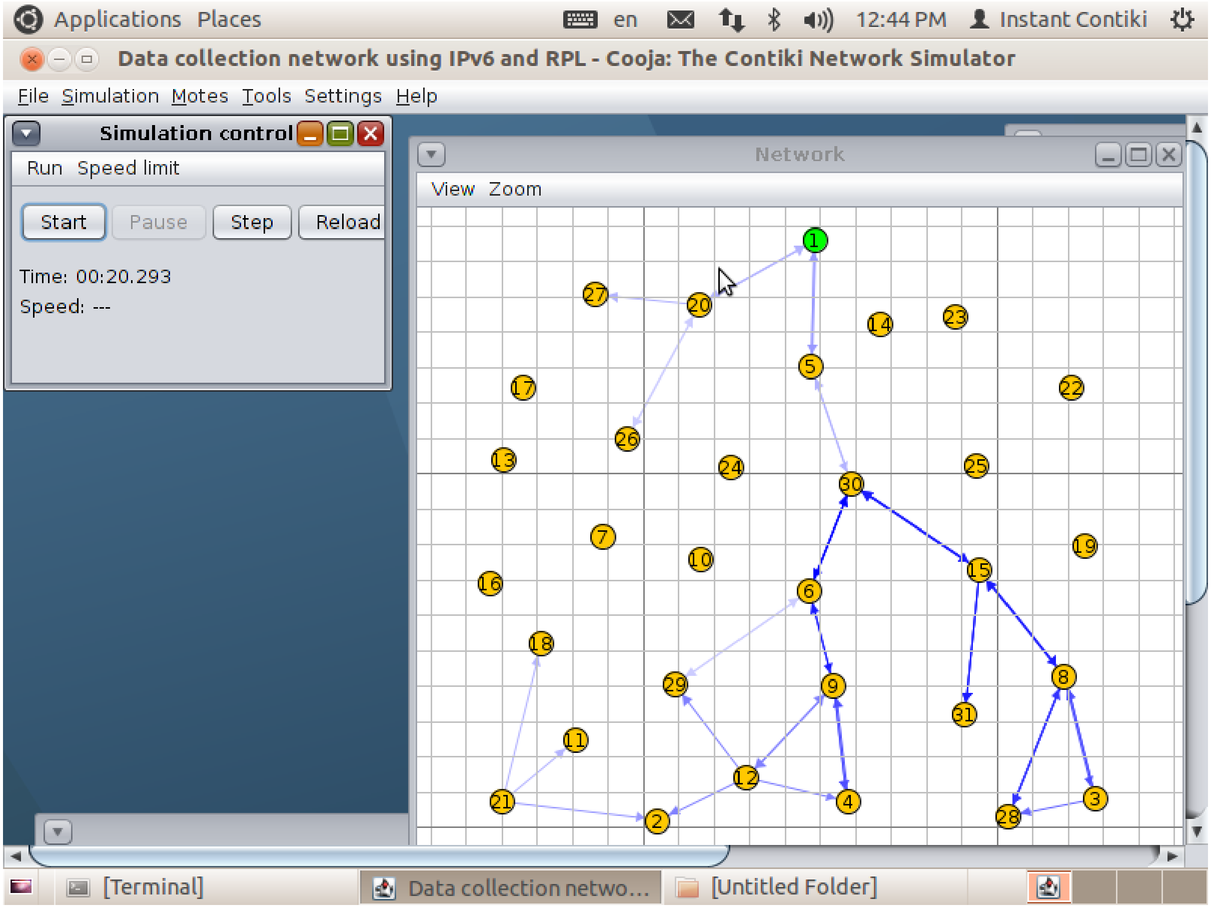


Figure 4.1: Implementation of RPL and creation of DODAG

Table 4.1: Hub Configuration

Parameter Name	Value
Internet Layer Simulated	ICMPv6, 6LoWPAN, IPv6, RPL
Transport Layer	UDP
Radio Model	UDGM(Unit Disk Graph Medium)
Bandwidth	250 kbps
Hub startup delay	1000 ms
Hub type	Tmote Sky

Calculation of Power Consumption:

A normal network has been considered for calculation of Energy consumption. First we started with 31 hubs and started sending packets randomly and after sending packets we have noted their TX, RX, LPM and CPU after every 10 seconds. After collecting these parameters we calculate the Power consumption and Radio Duty life Cycle for the network. In order to calculate formulas are given as:

Energy consumption (Power - mW) represented as **E** may be defined as given in equation 1

$$\boldsymbol{E = \frac{Energest\_Value \times Current \times Voltage}{RTIMER\_SECOND \times Runtime}}$$

(*Equation 4.1)*

Radio Duty Cycle% of hub represented as **R** is defined as

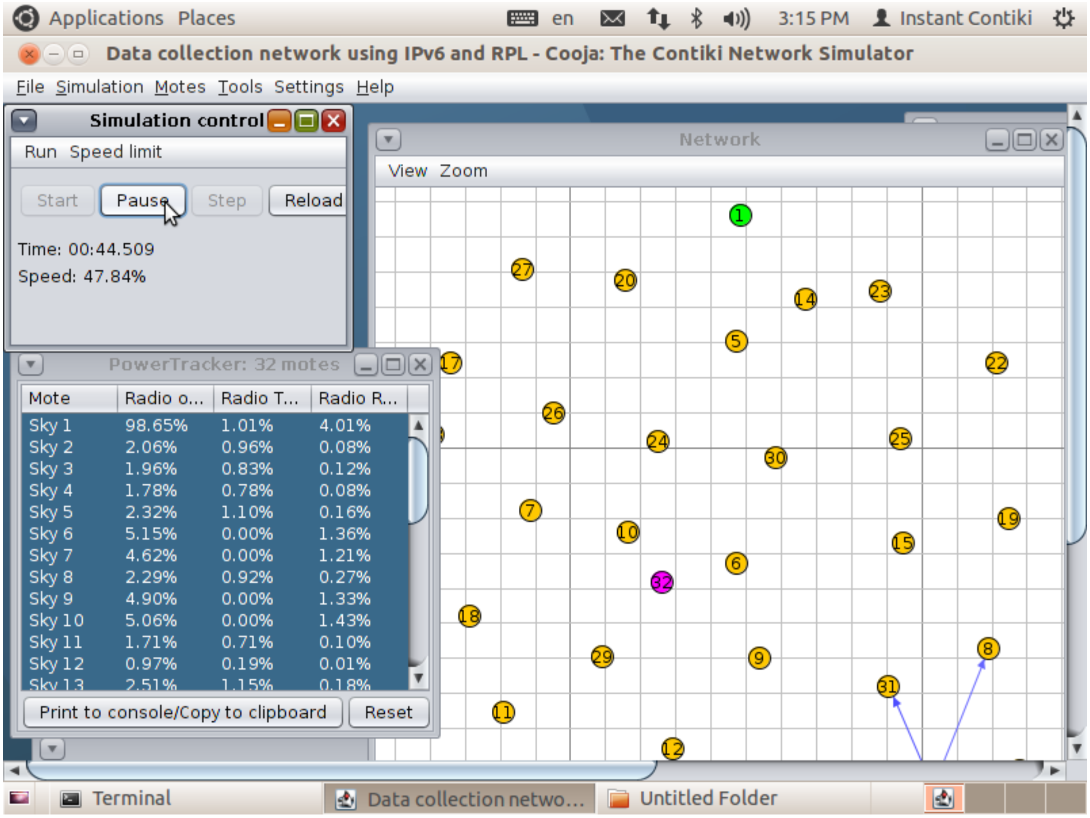
$$\boldsymbol{R = \frac{Energest\_TX + Energest\_RX}{Energest\_CPU + Energest\_LPM}}$$

(*Equation 4.2)*

After collecting the both the parameters it will look like a row way table. Data we have collected is almost uniform.

4.2.2 Deployment of Hello flooding DOS Invasion

We have developed a UDP flooding program. It is auto-programmed on different recipient motes. In this scenario Hub 32 is considered as an assailant hub which will going to flood the packets in the network. In the Figure below the simulation environment has 32 Hubs with 1 evil hub (in magenta) [4]


Applications Places
en 3:15 PM Instant Contiki
Data collection network using IPv6 and RPL - Cooja: The Contiki Network Simulator
File Simulation Motes Tools Settings Help
Simulation control
Run Speed limit
Start Pause Step Reload
Network
View Zoom
1 27 20 14 23 5 17 22 26 24 30 25 7 10 19 15 6 32 18 29 9 8 31 11 12
PowerTracker: 32 motes
Print to console/Copy to clipboard Reset
Terminal Data collection netwo... Untitled Folder

Time: 00:44.509
Speed: 47.84%

Mote	Radio o...	Radio T...	Radio R...
Sky 1	98.65%	1.01%	4.01%
Sky 2	2.06%	0.96%	0.08%
Sky 3	1.96%	0.83%	0.12%
Sky 4	1.78%	0.78%	0.08%
Sky 5	2.32%	1.10%	0.16%
Sky 6	5.15%	0.00%	1.36%
Sky 7	4.62%	0.00%	1.21%
Sky 8	2.29%	0.92%	0.27%
Sky 9	4.90%	0.00%	1.33%
Sky 10	5.06%	0.00%	1.43%
Sky 11	1.71%	0.71%	0.10%
Sky 12	0.97%	0.19%	0.01%
Sky 13	2.51%	1.15%	0.18%

Figure 4.2: RPL Assailant Simulation

To deploy D-dos invasion few changes has to make in the convention. We have followed steps as given in Ref [19]. This creates a JSON file. A JSON [20] file has been attached with the code of the hubs that changes the parameters of hub and make it a Disturber Mote. Next time the Disturber Mote can be directly accessed by Cooja interface. The main commands as described by procedure are as follows:

- **make**name[, n, ...]

  This creates a simulation named 'name' with specified parameters and also build all templates as root.c, sensor.c and disturber.c with the specified target mote type. This command also make the evil mote with an external library by providing its path.

- **make_all**simulation-campaign-json-file

  This will generate a campaign of simulations from a JSON file.

- **prepare**simulation-campaign-json-file

  From the template located at ./templates/experiments.json this command generates a campaign JSON file.

- **remake_all**simulation-campaign-json-file

  This re-generate evil motes for a campaign of simulations from the selected evil mote template.
- **run_all**simulation-campaign-json-file

  This will run the entire simulation campaign.

4.2.3 Detection and mitigation of D-dos Invasion

To detect and mitigate DoS invasion we made use of outlier Analysis which we have done on collected Power Consumption and Radio Duty cycle% Tables. Hubs detected as outlier in Statistical Outlier analysis will be considered as assailant or evil hub else it will be considered as suffererize hub. After deploying the DoS invasion an abrupt drain in the Energy of sufferer hub is observed, it can be easily concluded that Power Consumption and Radio Duty cycle% of sufferer hub is much less than that of assailant hub and these value comparison will be done after every 10 seconds. We can repeat this process unless we found all the assailant in the network.

4.3 Architectural Diagram of Advised System

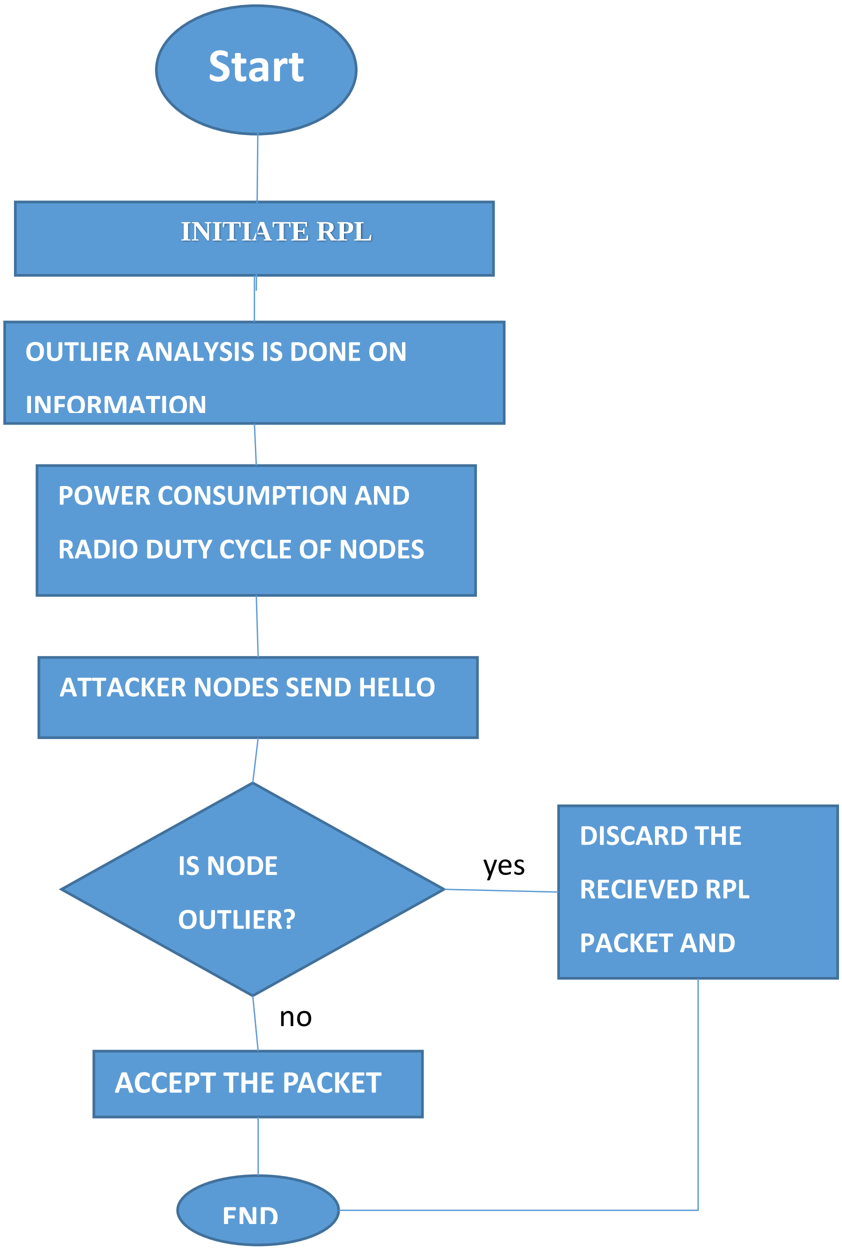

Start
INITIATE RPL
OUTLIER ANALYSIS IS DONE ON INFORMATION
POWER CONSUMPTION AND RADIO DUTY CYCLE OF NODES
ATTACKER NODES SEND HELLO
IS NODE OUTLIER?
yes
DISCARD THE RECIEVED RPL PACKET AND
no
ACCEPT THE PACKET
END

Figure 4.3: architectural diagram for advised method

CHAPTER 5

RESULTS AND ANALYSIS

5.1 Results

The advised system was implemented and simulated in Cooja simulator in Contiki Operating System. The used hub sky motes were placed randomly on a square Geographical Region.

Simulation work has been done in two phases:

i. First different scenario has been created without deploying invasion in order to calculate Power consumption and Radio Duty cycle. We started communicating with 31 hubs sending different amount of packets every time according to the RPL convention working. Transmission time, reiving time, CPU Time and LPM are observed and calculated.

Table 5.1: Data Gathered without creating Assailant hub

Time in Sec					Duty Cycle
10	AVG	ON	33002424 us	5.29%	0.21590069
	AVG	TX	7150901 us	1.15%	
	AVG	RX	2305760 us	0.37%	
	AVG	INT	780747 us	0.13%	
20	AVG	ON	12377884 us	4.02%	0.318115084
	AVG	TX	1967175 us	0.64%	
	AVG	RX	758188 us	0.25%	
	AVG	INT	245341 us	0.08%	
30	AVG	ON	33002424 us	5.29%	0.165906332
	AVG	TX	7150901 us	1.15%	

	AVG	RX	2305760 us	0.37%	
	AVG	INT	780747 us	0.13%	
40	AVG	ON	33002424 us	5.29%	0.235718518
	AVG	TX	7150901 us	1.15%	
	AVG	RX	2305760 us	0.37%	
	AVG	INT	780747 us	0.13%	
50	AVG	ON	69372071 us	5.59%	0.021528604
	AVG	TX	14184492 us	1.14%	
	AVG	RX	4568039 us	0.37%	
	AVG	INT	1710475 us	0.14%	

Table 5.2: Power consumption without Assailant

Time in Seconds	Average Power Consumption(in Mw)
10	77.7669321
20	87.9291546
30	91.66529279
40	80.82235162
50	83.17732983

With predefined value of hubs as current .33 amp, voltage 3 volt and run timer value as 32768.

ii. Then by introducing a Disturber hub an UDP flooding invasion is implemented. We started sending packets from root hub to destination in between there will be assailant hubs which will broadcast the upcoming packets and also they will flood hello packets to other hubs whosoever come in the range of these hubs. To check the versatility we have calculated data after every 10 seconds up to 50 seconds of each hub.

Table 5.3: Data Gathered with Assailant Hub

Time(sec)					Duty Cycle
10	AVG	ON	25740069 us	7.81 %	0.47792768
	AVG	TX	11858245 us	3.6%	
	AVG	RX	1415965 us	0.43%	
	AVG	INT	2034447 us	0.62%	
	Disturber_32	ON	10332600 us	100%	0.980392157
	Disturber_32	TX	10130000 us	98.04%	

	Disturber_32	RX	0 us	0%	
	Disturber_32	INT	0 us	0%	
20	AVG	ON	54843731 us	8.48%	0.453725645
	AVG	TX	24587577 us	3.8%	
	AVG	RX	3086110 us	0.48%	
	AVG	INT	4666873 us	0.72%	
	Disturber_32	ON	20257200 us	100%	0.980155132
	Disturber_32	TX	19860000 us	98.04%	
	Disturber_32	RX	0 us	0%	
	Disturber_32	INT	2400 us	0.01%	
30	AVG	ON	85184017 us	8.81%	0.443506184
	AVG	TX	37504279 us	3.88%	
	AVG	RX	4808924 us	0.5%	
	AVG	INT	7335182 us	0.76%	
	Disturber_32	ON	30263400 us	100%	0.980392157
	Disturber_32	TX	29670000 us	98.04%	
	Disturber_32	RX	0 us	0%	
	Disturber_32	INT	2400 us	0.01%	
40	AVG	ON	1.12E+08 us	8.69%	0.453352311
	AVG	TX	49196408 us	3.82%	
	AVG	RX	6418806 us	0.5%	
	AVG	INT	9925243 us	0.77%	
	Disturber_32	ON	40330800 us	100%	0.980158494
	Disturber_32	TX	39540000 us	98.04%	
	Disturber_32	RX	0 us	0%	

	Disturber_32	INT	4800 us	0.01%	
50	AVG	ON	1.39E+08 us	8.65%	0.433754888
	AVG	TX	60846011 us	3.77%	
	AVG	RX	7827043 us	0.49%	
	AVG	INT	12498578 us	0.78%	
	Disturber_32	ON	50479800 us	100%	0.980392157
	Disturber_32	TX	49490000 us	98.04%	
	Disturber_32	RX	0 us	0%	
	Disturber_32	INT	4800 us	0.01%	

Table 5.4: Power Consumption with Assailant Hub

Mote	Energy Consumption
Sky 1	291.0688522
Sky 2	11.41639069
Sky 3	6.441888428
Sky 4	10.75283569
Sky 5	5.890542297
Sky 6	15.9742337
Sky 7	15.98169617
Sky 8	7.081847534
Sky 9	15.83861023
Sky 10	15.94646851
Sky 11	7.447719727
Sky 12	2.767637329
Sky 13	4.882294006
Sky 14	5.893140564
Sky 15	7.851448059
Sky 16	5.01640686
Sky 17	2.765703735
Sky 18	5.080698853
Sky 19	6.99124054
Sky 20	3.957431946
Sky 21	3.720234375
Sky 22	3.400587158
Sky 23	4.032086792
Sky 24	16.16085571
Sky 25	7.45971405
Sky 26	4.935800171
Sky 27	2.144959717

Sky 28	7.182847595
Sky 29	15.9406073
Sky 30	2.868939514
Sky 31	8.253907471
Disturber 32	306.6256714

5.2 Result Analysis

Results were generated for two particular metrics:

1. The Radio duty life cycle of hubs.
2. The power consumption (Mw) of hubs.

The results were produced by measuring two different data sets one with simple RPL implemented and another one with introducing an assailant hub and then each one has been measured 5 times with an interval of 10 seconds different. The parameters taken into account are named as:

1. Packet Transmission Rate TX
2. Packet Receiving Rate RX
3. LPM (Low Power Mode)
4. CPU time

1. Based on The Radio duty life cycle of hubs

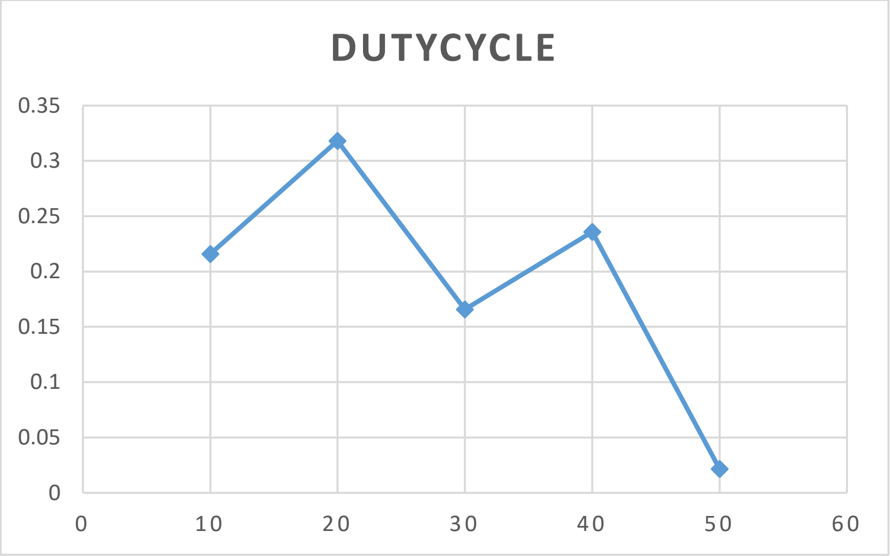


Figure 5.1: Result graph for Duty cycle of hubs without assailant hub

The above figure is a graph depicting results for test cases where duty cycle of hubs is plotted against time intervals of 10 seconds.

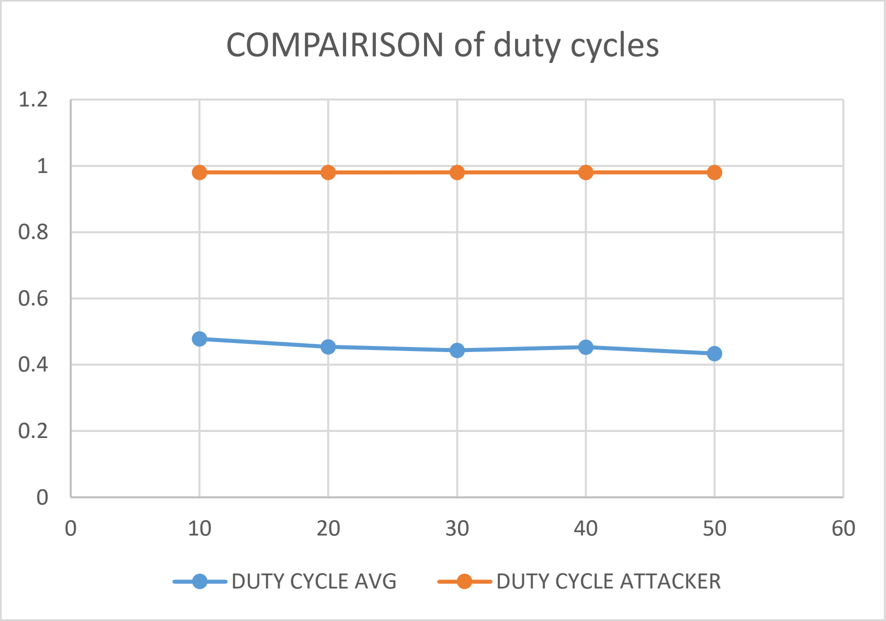


Figure 5.2: Result graph for comparison of hub's Duty Cycle with the assailant hub's Duty Cycle

The above figure is a The above figure is a graph depicting results for test cases where duty cycle of hubs is plotted against time intervals of 10 seconds with introducing an assailant hub.

2. Based on Power Consumption

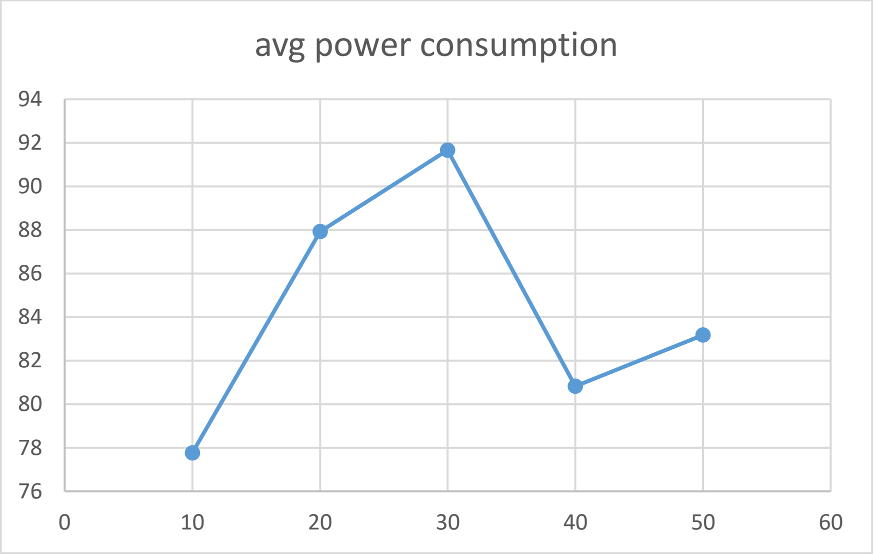


Figure 5.3: Result graph for average power consumption of hubs with assailant hub

The above figure is a The above figure is a graph depicting results for test cases where power consumption of hubs is plotted against time intervals of 10 seconds with introducing an assailant hub.

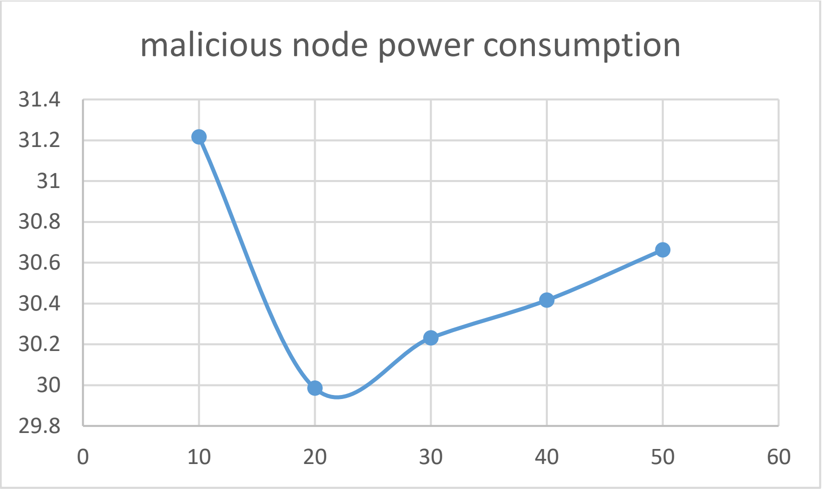


Figure 5.4: Result graph for average power consumption of assailant hub

The above figure is a The above figure is a graph depicting results for test cases where power consumption of hubs is plotted against time intervals of 10 seconds with introducing an assailant hub.

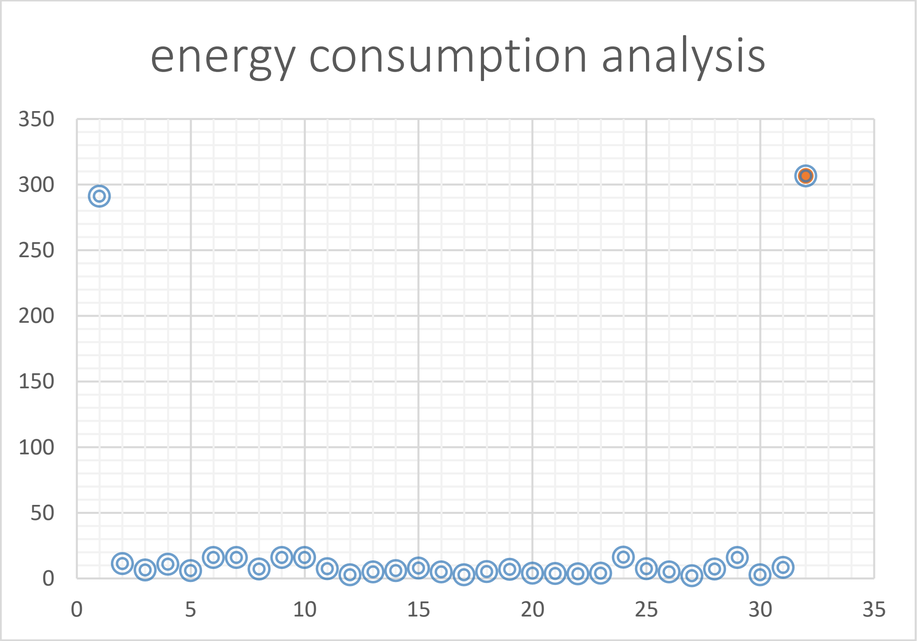


Figure 5.5 - Result graph for power consumption of each hubs and assailant hub

The above figure is a The above figure is a graph depicting results for test cases where power consumption of each hubs is plotted along with the assailant hub (shown in red colour) after 50 seconds of simulation.

5.3 Analysis and Outlier Analysis

The results depict the theorized difference between simple hub average parameters and parameters after applying flooding based assailant hub As is evident from the first graph, the radio duty cycle %."The ***duty cycle*** is the fraction of one period in which a signal or system is active and is commonly expressed as a percentage or a ratio" [21].

But form the second graph when an assailant hub is introduced its duty cycle % is approximately 1 that is near to 100% which means it is active all the time simulation runs which is far away from other hub's duty cycle are falling in a certain range .It is Firstly affecting the hubs in its range then all other hubs deployed in the region. This is because of the high number of data packets generated in the flooding based invasion.

It can be concluded from the above graph 5.5 that the power consumption (measured in Mw) is also very much higher than the other hubs excluding server. So the battery draining rate of this hub is very much and it is increasing draining rate of battery of other hubs too.

Outlier Analysis:

a) Based on Power Consumption

Using the Formulas, standard Deviation= 50.25387162

Minimum= 2.144959717

Mean= 16.94024603

Maximum= 291.0688522

Range= 288.9239

As described according to outlier detection technique .First we have calculated Z score. After calculating the Z score, if the Z score of energy consumption of any hub is between -2.68 to 2.68 it is a normal hub. Any hub having another Z score can be seen as outlier.

So, other than the hub 1 that is the server hub, only hub 32 is having a Z score of 3.973912863 and therefore, is detected as outlier. Further mitigation techniques can be applied to block the outlier.

a) Based on Radio Duty Cycle

Using the Formulas, standard Deviation= 0.223136728

Minimum= 0

Mean= 0.331103216

Maximum= 0.980392157

Range= 0.980392157

As described according to outlier detection technique. First we have calculated Z score. After calculating the Z score, if the Z score of Radio Duty Cycle of any hub is between -2.68 to 2.68 it is a normal hub. Any hub having another Z score can be seen as outlier.

So, other than the hub 1 that is the server hub, only hub 32 is having a Z score of 2.909826467 and therefore, is detected as outlier. Further mitigation techniques can be applied to block the outlier.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The project is based on Flooding DoS Invasion which is a battery draining invasion and in broad perspective it is a resource depletion invasion in IoT. In the complete procedure of analysis we have observed that battery power and radio duty cycle both are the most vital and important component of IoT hubs. Our methodology have successfully detected the assailant hub in early phase of simulation as the energy and Radio Duty Cycle% of hub with the help of Outlier Analysis. If we talk about Radio Duty Cycle % of Disturber hub it is reaching 1 and LPM is tending towards 0, i.e. the disturber hub is almost every time active so it's draining rate is very high also. We are successfully able to detect flooding DoS Disturber hub from the network

6.2 Future Scope

We have used some statistical data analysis methods like Quartile analysis and Outlier Analysis methods to detect the Disturber Hub from the networks systems using Residual energies of hubs and Radio Duty Cycle%. We can make use of other or same statistical methods to analyse other network parameters to detect and mitigate other network threats like Black Hole invasion, Grey Hole invasion etc.

REFERENCES:

[1]. Ashton, Kevin. "That ‘cyberspace of things’ thing." RFID journal 22, no. 7 (2009): 97-114.

[2]. Linus Wallgren, Shahid Raza, and Thiemo Voigt, “Routing Invasions and Counterattack in the RPL-Based Cyberspace of Things,” Int. J. Distributed Sensor Networks, vol. 2013, 794326, 2013.

[3]. Umamaheswari, S., and Atul Negi. "Cyberspace of Things and RPL routing convention: A study and evaluation." In Computer Communication and Informatics (ICCCI), 2017 International Conference on, pp. 1-7. IEEE, 2017.

[4]. Malik, Manisha, and Maitreyee Dutta. "Contiki-based mitigation of UDP flooding invasions in the Cyberspace of things." In Computing, Communication and Automation (ICCCA), 2017 International Conference on, pp. 1296-1300. IEEE, 2017.

[5]. Loukas, Georgios, and Gülay Öke. "Protection against denial of service invasions: A survey." The Computer Journal 53, no. 7 (2009): 1020-1037.

[6]. https://en.wikipedia.org/wiki/DOS

[7]. V. Hodge and J. Austin, A Survey of Outlier Detection Methodologies, Artificial Intelligence Review, Vol. 22, pp. 85-126, 2003.

[8]. D.M. Hawkins, Identification of Outliers, London: Chapman and Hall, 1980.

[9]. V. Barnett and T. Lewis, Outliers in Statistical Data, New York: John Wiley Sons, 1994.

[10]. VMware, “VMware Virtualization for Desktop & Server, Application, Public & Hybrid Clouds,” 2016. [Online]. Available: http://www.vmware.com/uk. [Accessed: 19-Feb-2016].

[11]. P. Thubert et al., "RPL: IPv6 Routing Convention for Low-Power and Lossy Networks," RFC 6550, 2012.

[12]. P. Levis, N. Patel, D. Culler, and S. Shenker, “Trickle: A Self-regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks,” in Proceedings of the 1st Conference on Symposium on Networked Systems Design and

Implementation - Volume 1, ser. NSDI’04. Berkeley, CA, USA: USENIX Association, 2004, p. 2. [Online]. Available: http://portal.acm.org/citation.cfm?id=1251177.

[13]. Schandy, Javier, Leonardo Steinfeld, and Fernando Silveira. "Average power consumption breakdown of Wireless Sensor Network hubs using IPv6 over LLNs." In Distributed Computing in Sensor Systems (DCOSS), 2015 International Conference on, pp. 242-247. IEEE, 2015.

[14]. Mangelkar, Smitesh, Sudhir N. Dhage, and Anant V. Nimkar. "A comparative study on RPL invasions and security solutions." In Intelligent Computing and Control (I2C2), 2017 International Conference on, pp. 1-6. IEEE, 2017.

[15]. Tsao, T., et al. “A Security Threat Analysis for the Routing Convention for Low-Power and Lossy Networks (RPLs)”. No. RFC 7416, 2015.

[16]. Gaddour, Olfa, and Anis Koubâa. "RPL in a nutshell: A survey." Computer Networks 56.14 : 3163-3178, 2012.

[17]. Jiang, Nan, et al. "Routing invasions prevention procedure for RPL based on micropayment scheme." Wireless Communications, Signal Processing and Networking (WiSPNET), International Conference on. IEEE, 2016.

[18]. https://blog.imaginea.com/simulation-of-rpl-dos-invasion-in-cooja/

[19]. https://github.com/dhondta/rpl-invasions

[20] https://www.json.org/

[21] https://en.wikipedia.org/wiki/Duty_cycle