COMPARISON OF HYBRID TECHNIQUE OF INTRUSION DETECTION SYSTEM IN 6LOWPAN

SITI AISHAH BINTI RAZALI

UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITY TEKNIKAL MALAYSIA MELAKA

COMPARISON OF HYBRID TECHNIQUE OF INTRUSION DETECTION SYSTEM IN 6LOWPAN

SITI AISHAH BINTI RAZALI

This report is submitted in partial fulfilment of the requirements for the Bachelor of [Computer Science (Computer Networking)] with Honours.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2021

DECLARATION

I hereby declare that this project report entitled

COMPARISON OF HYBRID TECHNIQUE OF INTRUSION DETECTION SYSTEM IN 6LOWPAN

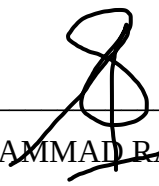
is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : _____ Date : 10th September 2021

SITI AISHAH BINTI RAZALI

اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of [Computer Science (Computer Networking)] with Honours.



SUPERVISOR : _____ Date : 10th September 2021

EN. MOHAMMAD RADZI MOTSIDI

DEDICATION

In honour of my supporting and wonderful friends, siblings and parents, who have always supported, led the way, and pushed me on my scholastic travels, I dedicate my final year project 1. Many thanks to my helpful instructor for his contribution to encourage me from the start to the end of my research.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ACKNOWLEDGEMENTS

All glory be to Allah, I shall succeed in the end, and I have completed my final year assignment 1 (FYP 1). I would like to recognize my project's supervisor, En. Mohammad Radzi Motsidi, for his important position, dedication, and endless patience during the project's development and evaluation of me. Without his assistance, the project report could not be finished. I would also want to credit my wonderful family, which includes all of my siblings, for their aid and encouragement throughout my challenging struggle to accomplish my project. Additionally, I would want to thank everyone who helped make this report an accomplishment, especially my housemate. My comparison hybrid technique IDS in 6LoWPAN, project research and experiment are seemed to be completed. This research will be utilised as a guide for Final Year Project 2's experience and completion in the final semester.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

The term "Internet of Things" refers to the connectivity of physical items, such as smart objects, that exchange data and provide services through the internet. The network between IoT nodes can be protected by avoiding attacks with conventional mechanisms such as encryption and authentication, however these methods will not detect all potential attacks. The resource-constrained sensors node in an IoT environment are causing untrusted connection were made since the connection are made through internet using IPv6 and because communication protocol is IPv6 over Low -Power Wireless Personal Area Networks (6LowPAN). 6LowPAN is a communication protocol that were used for resource-constrained applications. Since the attack are more likely to happen, Intrusion Detection System are necessary to detect the attack that occur in the system or network by analysing the activity in the system or in the network and get the IDS log information about it and get the alarm report. In this project research will be focus on hybrid detection method using Network Intrusion Detection System (NIDS). The project experiment is a comparison between two extended version of existing computer simulation of hybrid detection method IDS, SVELTE. The Cooja network simulator will be used as the Routing Protocol for Low-Power and Lossy Networks (RPL) simulator in a 6LowPAN environment to create nodes of IoT and the attacker such as sink node using set of source code. The simulation produced network traffic and the data, metrics, and graph of RPL that can be collect and analyse.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRAK

Istilah "Internet of Things" merujuk kepada penyambungan item fizikal, seperti objek pintar, yang saling bertukar data dan menyediakan perkhidmatan melalui internet. Jaringan antara nod IoT dapat dilindungi dengan menghindari serangan dengan mekanisme keselamatan seperti enkripsi dan pengesahan, namun kaedah ini sukar untuk mengesan semua serangan. terhadap nod sensor yang terhad sumber dalam persekitaran IoT ini. Ini merisikokan rangkaian tersebut kerana IoT digunakan melalui internet dengan menggunakan IPv6 dan protokol komunikasi dalam IoT adalah IPv6 melalui Rangkaian Kawasan Peribadi Tanpa Wayar Rendah-Kuasa (6LowPAN). 6LowPAN adalah protokol komunikasi yang digunakan untuk peranti yang kekurangan sumber. Oleh kerana serangan lebih cenderung berlaku, Sistem Pengesanan Pencerobohan (IDS) diperlukan untuk mengesan serangan yang berlaku di sistem. Dalam projek ini penyelidikan akan difokuskan pada kaedah pengesanan hibrid menggunakan Sistem Pengesanan Pencerobohan Rangkaian (NIDS). Eksperimen projek ini adalah perbandingan antara dua versi simulasi komputer yang telah dinaik taraf daripada IDS sedia ada, iaitu IDS kaedah pengesanan hibrid, SVELTE. Simulator rangkaian Cooja akan digunakan sebagai simulator *Routing Protocol for Low-Power and Lossy Networks* (RPL) di persekitaran 6LowPAN untuk membuat nod IoT dan penyerang seperti *sink node* menggunakan set kod sumber. Simulasi dihasilkan melalui graf rangkaian dan data, metrik, dan graf RPL untuk dikumpulkan dan dianalisis untuk hasil perbandingan.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

TABLE OF CONTENTS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF TABLES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF ABBREVIATION

6LowPAN	-	**IPv6 over Low -Power Wireless Personal Area Networks**
RPL	-	**Routing Protocol for Low-Power**
IDS	-	**Intrusion Detection System**
DODAG	-	**Destination-Oriented Directed Acyclic Graph**
IoT	-	**Internet Of Things**
LLN	-	**Low Power and Lossy Network**
DIS	-	**DODAG Information Solicitation**
WSN	-	**Wireless Sensor Network**
6BR	-	**IPV6 Border Router**
OS	-	**Operating System**
DIO	-	**DODAG Information Object**
OF	-	**Objective Function**
ETX	-	**Expected Transmission Count**

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 1 : INTRODUCTION

1.1 Introduction

Chapter 1 will focus on the planning of the project where the problem statement (PS), project question (PQ), project objective (PO), project scope and project contribution (PC) will be discussed. The project is about research of the analysis of hybrid technique approach in the intrusion detection system. The Cooja network simulator will be used as the RPL simulator in a 6LowPAN environment to create nodes of Internet Of Things (IoT) and the attacker such as sink node using set of source code. RPL stands for IPv6 Routing Protocol for Low-Power and Lossy Networks. Source code of Intrusion Detection System (IDS) implementation will be used inside the simulation. The simulation produced network traffic and the data, metrics, and graph of RPL that can be collect and analyse (tuz-Zahra et al., 2020).

There would be the comparison of hybrids IDS technique in order to compare the IDS evaluation. The result of false positive, false negative, number of nodes, time before an attack are collected in order to count the average percentage of the detection rate IDS (Nygaard, 2017). The attack from dataset csc file extension that would be evaluate is sinkhole attack and selective forwarding attack. Sinkhole attack enables intruders to interrupt and alter network traffic which that if occurred together with selective forwarding attack will effect larger part of the network to be controlled by the intruders (Raza et al., 2013). Hence, it is important to detect these two attacks.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.2 Problem Statement

Network Intrusion Detection System has been introduced for few years back on, and so are the hybrid technique as one of the techniques commonly used in Intrusion Detection System. This project would be identified to differentiate between the existing computer simulation hybrid technique IDS to identify which hybrid

technique IDS suitable according to environment. As most IDS technique require different routing schemes that are not based on standardized mechanisms (Raza et al., 2013).

Table 1.1 Problem Statement (PS)

PS	Problem Statement
PS_1	To identify the most stable network between existing computer simulation hybrid technique Intrusion Detection System in IoT environment.

1.3 Project Question

Problem statement element to be measure are power consumption, and the Expected Transmission Count (ETX) value of the network. These components were used to determine the quality of the IDS implemented causing to the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) environment. Since the IDS are implemented around network layers it is important to know how the performance being monitored. In Internet of Things (IoT), the Low Power and Lossy Network (LLN) is a sector that includes constrained devices such as sensors and RFIDs. The routing protocol for Low Power and Lossy Networks (LLN) is called Routing Protocol for Low-Power (RPL) that is an open routing protocol that enables nodes to interact in a mesh topology of 6LowPAN (Bhattacharyya and Pushpalatha, 2018).

RPL routing protocol in 6LowPAN are vulnerable to attacks. The link in the RPL can brings various of attacks through Destination-Oriented Directed Acyclic Graph (DODAG) Information Solicitation (DIS) when transmitting nodes to join a network (Wallgren et al., 2013). This issue arises a question on how can Network IDS in hybrid detection method help to overcome the attacks? Apart from that, there are many hybrid techniques in computer simulation to implement IDS, how would these hybrid techniques be any different from each other? This question arises and would be discover within this project.

Table 1.2 Project Question (PQ)

PS	PQ	Project Question
PS_1	PQ_1	How the simulation performance of IDS in IoT environment monitored?
	PQ_2	How a hybrid detection method IDS in IoT environment help to overcome the RPL attacks?
	PQ_3	What is the difference between the existing

		hybrid detection method IDS(s) in computer simulation of IoT environment?

1.4 Project Objective

The objective of the project are to investigate the implementation of hybrid detection method Intrusion Detection System in simulation IoT, 6LoWPAN network environment. Next, objective to compare the effectiveness between the hybrid detection method of IDS(s) in simulated IoT environment based on the experiment analysis and lastly to make recommendations of best simulation environment settings based on the experiment analysis of hybrid detection method IDS in simulated IoT, 6LoWPAN network environment.

Table 1.3 Project Objective (PO)

PS	PO	Project Objective
PS_1	PO_1	To investigate the implementation of hybrid detection method Intrusion Detection System in simulated IoT, 6LoWPAN network environment.
	PO_2	To compare the effectiveness between the hybrid detection method of IDS(s) in simulated IoT environment based on the experiment analysis.
	PO_3	To make recommendations of best simulation environment settings based on the experiment analysis of hybrid detection method IDS in simulated IoT, 6LoWPAN network environment

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.5 Project Scope

There were several detection methods of Intrusion Detection System (IDS), one of it was hybrid detection method. Hybrid detection method IDS are the mixture of both signature and anomaly detection method which are more efficient to detect more types of attacks than signature and anomaly detection method. The project scope is focusing on comparison hybrid detection method of IDS.

The experiment of the comparison will be implemented on simulation software which is Cooja network simulator that will be use inside a Contiki Operating System (OS) since Cooja has been demonstrated to be an excellent tool for simulating IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) in Wireless Sensor Network (WSNs).

The comparison are involving the extension version of existing computer simulation, SVELTE by Raza et al. (2013), that is IDS by Matsunaga et al. (2015) and IDS by Shreenivas et al. (2017). The source code that will be use in this project would be the existing computer simulation SVELTE by Raza et al. (2013), where the dataset of IPV6 Border Router (6BR) inside the IPv6 over Low -Power Wireless Personal Area Networks (6LowPAN) are to specified the rank metrics of the IoT nodes in, next the IDS client and also the malicious node. The source code will be manipulated according to algorithm of Matsunaga et al. (2015) and Shreenivas et al. (2017) proposed. The evaluation of the experiment will be the dataset of experiment, power consumption and ETX average value.

Table 1.4 Project Scope

Project Scope	**Details**
Experiment of	IDS Computer Simulation
Intrusion Detection System Type	Hybrid IDS
Platform	Virtual Machine (VMWare Workstation)
Operating System	Contiki
Source Code	IDS by SVELTE, Matsunaga et.al (2015) and Shreenivas et. al (2017)
Network Simulator	Cooja
Dataset	Data from Cooja 'Collect-view' features
Evaluation component	1. Power Consumption 2. ETX average value

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

1.6 Project Contribution

The project analyzes the comparison between the hybrid detection method Intrusion Detection System by power consumption and ETX value. These comparison benefits to user who uses IOT environment network as part of their life as the implementation of IDS in hybrid technique works differently in each environment. Hybrid technique also was chosen for this project also as it the better method than signature and anomaly detection method because hybrid detection method overcome the weaknesses of signature and anomaly technique (Napiah et al., 2018).

Table 1.5 Project Contribution (PC)

PS	**PQ**	**PO**	**PC**	**Project Contribution**
PS_1	PQ_1	PO_1	PC_1	Proposed a technical process of how the

				hybrid detection method IDS working in simulation environment.
		PO_2	PC_2	Proposed a comparison result of detection rate and accuracy rate of existing hybrid detection method IDS in computer simulation IoT environment.
		PO_3	PC_3	Proposed analysis report based on best simulation environment settings based on the experiment analysis of hybrid detection method IDS in simulated IoT, 6LoWPAN network environment.

1.7 Report Organisation

Chapter 1: Introduction

This chapter discuss about the analysis planning towards hybrid technique of network Intrusion Detection System. The planning is as important as to figure the main component to be analyzed and the component to be evaluate in implementation phase. Problem statement is to identify the most stable network between existing computer simulation hybrid technique Intrusion Detection System in IoT environment.

There are three project questions developed from the problem statement, which is PQ_1, How the simulation performance of IDS in IoT environment monitored? next PQ_2, how a hybrid detection method IDS in IoT environment help to overcome the RPL attacks? and PQ_3, what is the difference between the existing hybrid detection method IDS(s) in computer simulation of IoT environment? Project scope for this project as per discussed, the IDS detection method, the simulation software, the dataset and the component to be evaluated.

Lastly project contribution is to propose a technical process of how the hybrid detection method IDS working in simulation environment., proposed a comparison result of detection rate and accuracy rate of existing hybrid detection method IDS in computer simulation IoT environment and proposed analysis report based on best simulation environment settings based on the experiment analysis of hybrid detection method IDS in simulated IoT, 6LoWPAN network environment.

Chapter 2: Literature Review

The project's literature analysis will be the subject of Chapter 2. These would aid in the development of a theoretical framework and technique for the topic study field, as well as a picture of material similarities in each subject. This chapter will describe chosen content relevant to Intrusion Detection System (IDS) for 6LoWPAN network hybrid technology. In this chapter, the proposed strategy for each item will be justified. Hybrid technique is one of the most often used techniques in Intrusion Detection Systems (IDS). This paper will analyses and evaluate major components of hybrid technique requirements using selected academic sources.

Chapter 3: Project Methodology

The project's methodology will be identified and detailed in Chapter 3. This project's technique would be to implement the project using a model System Development Life Cycle (SDLC).The phases are planning phase, analysis phase, design phase, implementation phase, testing phase, integration and analysis phase. Each phase's aspects will be identified and developed. This chapter will also cover the project's major milestones. The project strategies may be used to specify how procedures, components, and processes are organised, as well as architecture, division, and perspective

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Chapter 4: Design

In this chapter, the planning of project will be inspected to make sure the project can be done realistically. This chapter will emphasize the project prerequisite in terms of software requirement, programming specifications and the project limitation. The simulation design and data validation will be elaborated in detail in this chapter to emphasize the component involved in this project experiment.

Chapter 5: Implementation

The previous chapter discussed over project design, and this chapter will go over project implementation. This chapter will enlighten on how the simulation environment will be configured based on the source of main Intrusion Detection System (SVELTE) according to IDS by Matsunaga et. al. (2015) and Shreenivas et. al. (2017).

Chapter 6: Testing and Analysis

The project's testing procedure is described in the sixth chapter. Furthermore, testing is essential to confirm that the finished product and system fulfil the requirements and function properly. In addition, this step will strengthen the project's involvement in achieving the project's goal. The texting will be involving the experiments that will be done in this project. There would be eight experiments in total. All components and modules will be verified to guarantee that the experiment is done according to the project requirement and precisely.

Chapter 7: Project Conclusion

This chapter will summarize the project's conclusion and progress. It will discuss the overall project progress and achievement of this project's contribution, as well as the project's capabilities, weaknesses, and future improvements. Furthermore, every project specific will be clarified and comprehensible by giving the project overview. This chapter will also go through the changes that will be made and how they will be implemented for the next phase of the experiment.

1.8 Conclusion

This chapter discuss about the analysis planning towards hybrid technique of network Intrusion Detection System. The planning is as important as to figure the main component to be analyzed and the component to be evaluate in implementation phase. In this chapter, problem statement (PS), project question (PQ), project objective (PO), project scope and project contribution (PC) were discussed. Next chapter would literature review of the project.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 2 : LITERATURE REVIEW

2.1 Introduction

Chapter 2 will focus on the literature analysis of the project. Literature review is a survey of scholarly sources on a particular field of research. It refers to the collection of published materials on certain field of research to be analyzed, synthesized, and evaluated. These would help to see the picture of similarity of the materials on the specific field and helps to develop theoretical framework and methodology of the topic research field.

This chapter would summarize the selected material related to Intrusion Detection System (IDS) for hybrid technique in 6LoWPAN network. The suggested approach of each material would be justified in this chapter. Hybrid technique has been one of the most practiced technique in Intrusion Detection System (IDS), this may help to analyses and evaluate key components of requirement in hybrid technique based on the selected scholar sources.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.2 Internet Of Things

Internet of Things refers to the connection between physical devices such as smart objects that exhanging data and offer services using internet (Ahmed et al., 2017). The connection between the IoT nodes may be secured by preventing attacks using standard mechanism like cryptography and authentication process but these way will not detect all possible attacks (Bostani and Sheikhan, 2017). However, because IPv4 has a limited address space, items in the Internet of Things (IoT) employ IPv6 to expand their address space.

The Internet of Things (IoT) is a hybrid network of small devices, usually WSNs, and the traditional Internet. A wireless sensor network (WSN) is a collection of

sensor nodes that detect, record, and transmit environmental data to a sink node. The sink node then processes the data it has received and corresponds with the router nodes. The sensor nodes are limited-resource devices with limited computation power, that is all basically overall of the concept of IOT.

2.3 Security In IoT

Because of the rapid expansion of the Internet of Things, its security has become one of the most complex concerns in such a connected and mutual framework (Hajiheidari et al., 2019). Now, the number of smart things connected to the Internet is growing as a rising number of developing WSNs employ IP. Over trillion smart objects will be controlled and connected via the Internet and a variety of applications. As the number of devices connected to the Internet grows, security becomes a more pressing concern.

The resources-constrained like sensors node in IoT environment are causing untrusted connection were made since the connection are made through internet using IPv6 and also because communication protocol is 6LowPAN (Raza et al., 2013) (Bostani and Sheikhan, 2017). Since the attack are more likely to happen, Intrusion Detection System are necessary in order to detect the attack that occur in the system or network by analysing the activity in the system or in the network and get the IDS log information about it and get the alarm report (Raza et al., 2013).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Despite the fact that IoT networks can only be accessed by authorised users, they are vulnerable to a variety of assaults. These attacks try to interrupt network connection or capture personal information. Denial-of-service (DoS) attacks, for example, degrade communication at the network layer very quickly (Oh et al., 2014).

Sinkhole attack are one of the RPL attacks that often occurs. Sinkhole attack is when a malicious node propagates its rank with a high value, which is usually anything like the sink's rank. Consequently, its neighbours will choose it as their preferable parent and traffic will be directed to it. To drop all the traffic attracted, the sinkhole attack is frequently paired with the selective forwarding attack.

2.4 6LowPAN Network

6LowPAN is a communication protocol that were used for resource-constrained applications (Raza et al., 2013). A 6LoWPAN network is a multi-hop wireless

network with lossy communication links and resource-constrained devices that are frequently powered by batteries. As a result, in 6LoWPAN networks, the connectionless

User Datagram Protocol (UDP) is a basic OSI transport layer protocol based on Internet Protocol for client/server network applications (IP). UDP is commonly utilised in application to run in real-time. For example, for this project will utilised UDP due to the real-time data collecting. 6LoWPAN is an IPv6 stub network network design for low-power wireless area networks. It compresses or decompresses IPv6 datagrams and fragments or assembles them. It may be used on any device and is not limited to constrained devices alone (Napiah et al., 2018).

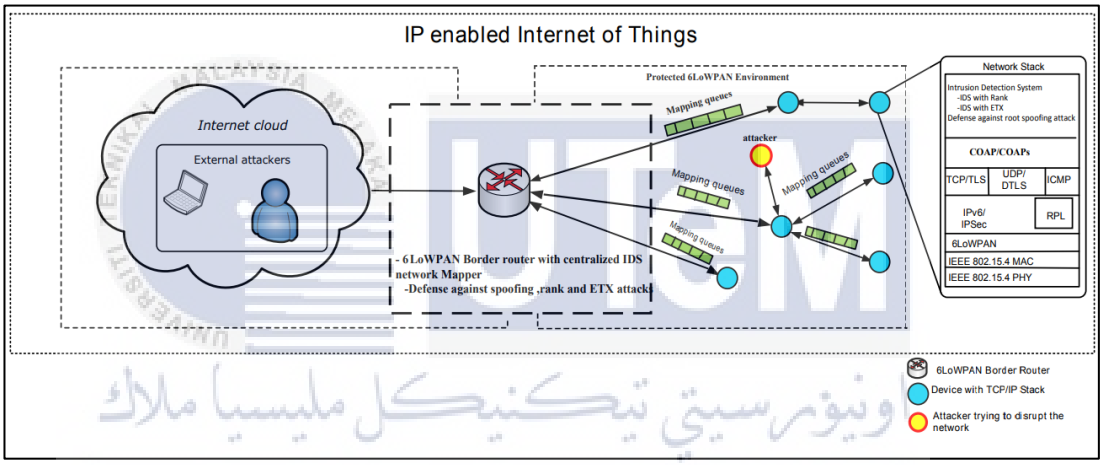


Figure 2-1 Architecture of Network in 6LowPAN

In Figure 2-1 shows that the 6LoWPAN Border Router (6BR) is used to link all the device nodes to the Internet, which are also the Destination-Oriented Directed Acyclic Graph (DODAG) that was constructed by RPL. RPL stands for IPv6 Routing Protocol for Low-Power and Lossy Networks, and it is a standardized routing protocol for IP-connected IoT.

2.5 RPL

RPL is a flexible protocol that allows communication between many-to-one, many-to-many, and one-to-one. It constructs a Destination-Oriented Directed Acyclic Graph (DODAG) and supports two modes of operation: unidirectional traffic

to a DODAG root (usually the 6BR) and bidirectional traffic to a DODAG root (usually the 6BR) (Raza et al., 2013).

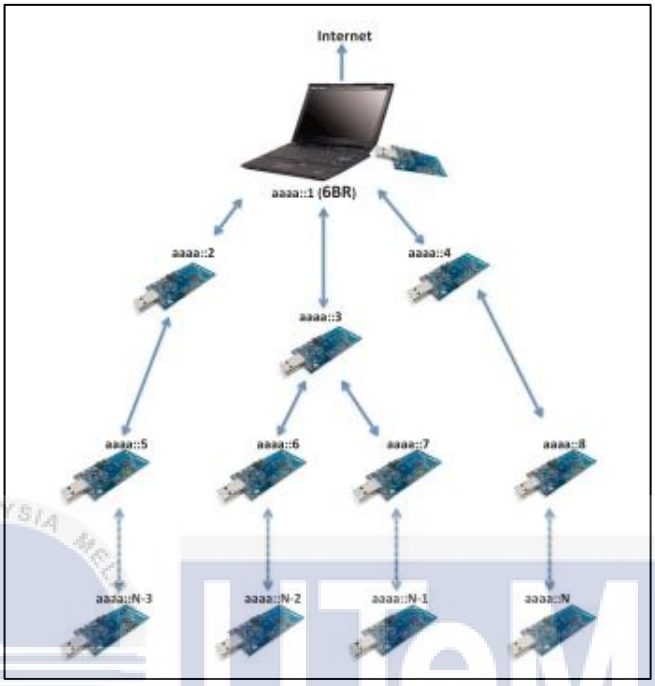


Figure 2-2 An example of RPL DODAG with N nodes and IPv6 addresses ranging from aaaa::1 to aaaa::N (Raza et. al., 2013)

DODAG (Destination Oriented Directed Acyclic Graph) is an IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) that takes the shape of a tree with one root known as a sink node. RPL create DODAG by using Objective Function (OF) to minimize energy and latency of network (Mardini et al., 2018). In RPL, the rank selection is calculated by an objective function (OF) through the link quality of the rank calculation, which is the Expected Transmission Count (ETX).

The ETX is used to calculate rank and small average value of ETX helps to ensure the network stability. It approximates the link quality and also the maximum number of retransmissions required to transmit a single packet to its intended destination. RPL performance may effect when the number of nodes was increased because of the increasing of received packets, number of hops, Rtmetric value, ETX, and power consumption (Mardini et al., 2018).

The transmission of DODAG Information Object (DIO) messages is required for the establishment of the topology root node. RPL is a new distance vector routing protocol that enables nodes to interact in a mesh topology in limited 6LoWPAN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

networks. Unfortunately, RPL routing protocol in 6LowPAN are vulnerable to attacks.

2.6 Intrusion Detection System

The IDS can be categorized as type, Network IDS (NIDS), Host IDS (HIDS), Hybrid IDS, Network Behaviour Analysis (NBA), and Wireless-based (Othman et al., 2018a) (Liao et al., 2013). There are few method detection of IDS, signature based, anomaly based, specification based, hybrid approach and knowledge driven approach (Nandhini and Mehtre, 2019) (Zarpelão et al., 2017) (Hajiheidari et al., 2019).

IDS also can be categorised as placement architecture of the IDS in the network, centralized, distributed and hybrid of the two centralized and distributed (Midi et al., 2017). In this project research will be focus on hybrid detection method using Network Intrusion Detection System (NIDS). There were also few platform to simulate the IDS in IoT environment, Cooja network simulator, C/C++, SENSE, Raspberry Pi, TineyOS TOSSIM and many more.

اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

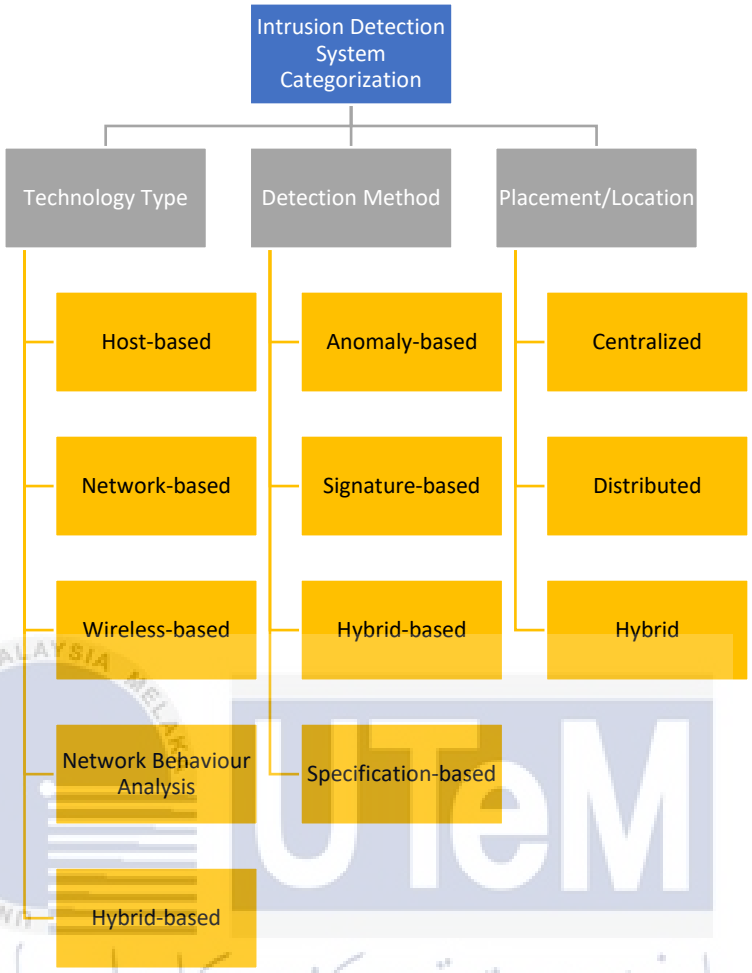


Figure 2-3 Intrusion Detection System Categorization

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

i) Technology Type

a) Host-IDS: Technology type

HIDS was the first intrusion detection system to be created. HIDS monitors and analyses the tasks of a single host's inbuilt computing system, such as software behaviour, wireless network traffic specifically for that host only, system logs, network interface system configuration, audit log, running user or application processes (Othman et al., 2018b). Because it analyses log files, HIDS is more accurate and fewer false positive than network-based IDS. As a consequence, it can identify whether an attack was successful or not. OSSEC and Tripwire are two examples of existing systems that use the host-ID system type. Nevertheless, analysing a huge volume of data in order

to distinguish between known and unknown activities takes a long time and a lot of resources.

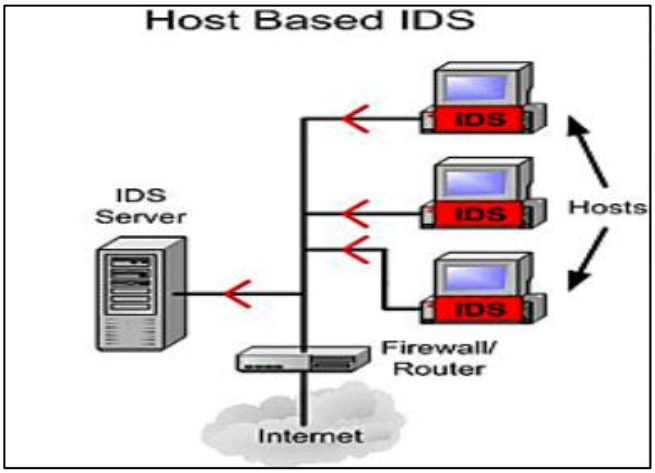


Figure 2-4 Host-IDS (Technology types)

b) Network-IDS: Technology type

The NIDS is used to keep track of the network's traffic. It is also used to analyse traffic in order to defend the system from network-based attacks. NIDS will examine all of the packets coming into the network and look for unusual patterns. If threats are recognised or found, the system can take action based on their severity, such as blocking the source's IP address. NIDS focuses on the misuse of vulnerabilities, whereas HIDS focuses on the abuse of power (Othman et al., 2018b). The difficulty with NIDS is that it only has limited access inside the host computer, and there is no reliable mechanism to examine protected network data in order to detect attacks. Snort and NetSTAT, which is a tool focused on real-time NIDS, are one of the network intrusion detection.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

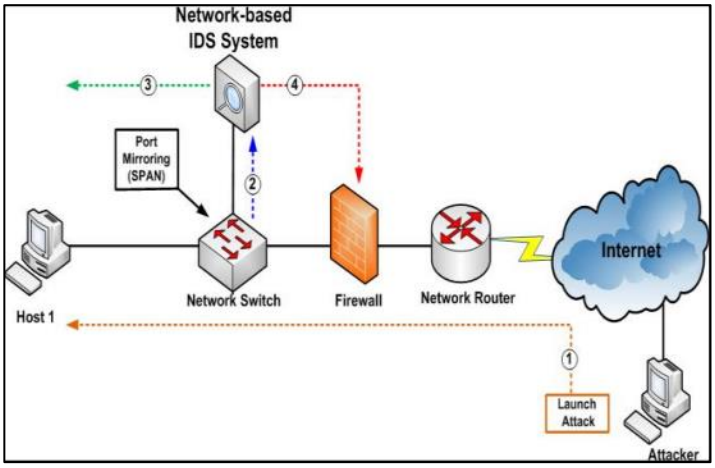


Figure 2-5 Network IDS (Technology Type)

c) Hybrid: Technology type

Hybrid IDS as example, Double Guard IDS, which employs host IDS and network IDS, combines two or more forms of IDS to accomplish the benefits of IDS and complete an accurate detection. However, Hybrid IDS takes a lengthy time in evaluating data. Hybrid-based IDS is depicted in Figure 2-6.

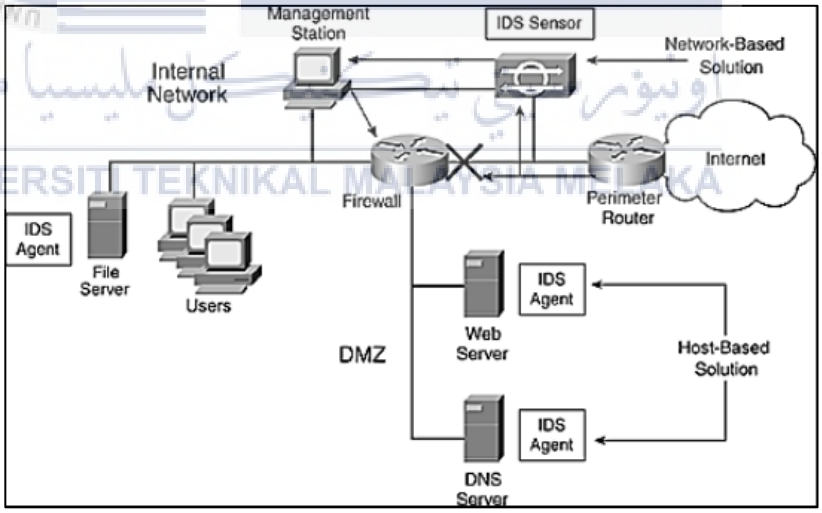


Figure 2-6 Hybrid IDS(Technology types)

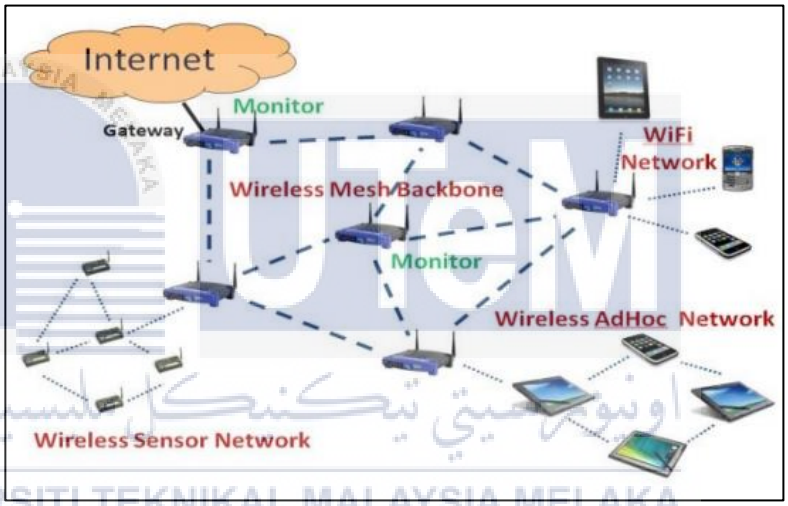
d) Network Behaviour Analysis: Technology type

Network Behaviour Analysis (NBA) monitors and analyses network traffic to identify the risks such as DDOS attacks, viruses, and policy breaches that result in unusual traffic patterns. The NBA system examines network traffic in order to detect assaults with unusual traffic flows. In

particular, the behaviour is compared to a standard to detect any unusual activity in a network. NBA has the benefit of focusing on the whole behaviour of network devices, allowing it to deal with unidentified or particular threats.

e) Wireless-IDS: Technology type

Wireless IDS (WIDS) analyses and monitors wireless network in order to identify any threats. Ad-hoc networks, wireless mesh networks, and wireless sensor networks are all examples of wireless traffic. Sinkhole attack, spoof altered routing attack, Flood attack, and Sybil attack are all examples of wireless network attacks.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2-7 Wireless-IDS (Technology Types)

ii) Detection method

a) Anomaly-based

The task of a system will be evaluated to the typical behaviour pattern at any time in anomaly-based IDS. The alert is created if there is any deviation from usual behaviour and also if the value surpasses the limit value. It is also capable of detecting new types of attack. This method can also be used to detect attacks involving resource waste.

b) Signature-based

Signature-based detection is where the attack's signature or pattern is saved in the IDS's internal database. The alert will be issued if any

network or system activity matches the preset pattern. These intrusion detection systems (IDS) are effective at identifying known threats.

c) Hybrid-based

Hybrid detection methods are the upgradation on signature and anomaly where both detection method were applied within the same IDS (Raza et al., 2013) (Liao et al., 2013). The list of hybrid detection method from past research were in Table 2-1.

d) Specification-based

The specification in this system refers to the collection of rules or limits that are used to describe the behaviour of network components. Routing tables, protocols, and nodes are examples of networking devices. The warning is triggered if there is any difference from the specified standard. This is related to the anomaly-based method, which involves finding deviations from the norm.

iii) Placement architecture

a) Centralized

In comparison to a distributed system, the cost of maintenance and administration is reduced where the central console is in charge of all IDS actions. The limitation of centralized are it incapable of detecting harmful events occurring in many locations at the same time.

b) Distributed

IDS that are distributed have the capacity to link alerts from several sensors. The coordination unit connects these intrusion alarms, then generates reports, and lastly confirms the true state of the intrusions. This has the ability to make the IDS self-contained, capable of self-adjusting, parallel, structured, and efficient.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.6.1 SVELTE

SVELTE by Raza et al. (2013) is a Network IDS for Internet of Things. Raza et al. (2013) designed SVELTE integrated with mini firewall as security measures and also IPSec for end-to-end message security. SVELTE placement design is including hybrid of centralized and distributed architecture. There were three main modules placed inside the border router, one of it was the IDS module. SVELTE one of the IDS in IoT that is only available in Contiki OS.

The first module was 6LoWPAN Mapper (6Mapper) that are works for collecting information about the RPL network which also works to reconstructs border router's network. The second module works where the border router is the intrusion detection component. This component evaluates the 6Mapper's mapped data and identifies the attack. The mini firewall is the last module, and it cuts out undesirable traffic before it affects the nodes with limited resources. The firewall protects the external harmful host provided by a node within an RPL network in real time. There are two centralised modules on each of these nodes where the one of it gives mapping information and the other that interacts with the firewall.

SVELTE takes into account two types of assaults: selective forwards attacks and sinkhole attacks. With sinkhole attacks, SVELTE has strong results (90 percent true positive rate) for small networks, but this drops to 70-80 percent in a bigger network with 32 nodes. When it comes to selective forwarding attacks, SVELTE has respectable true positive rates, with virtually a 100 percent detection rate.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

According to Matsunaga et al. (2015), SVELTE has a high false detection rate. False positives occur when the border router sends a request between rank updates and each node broadcasts their DIO message, based on the research. Another cause SVELTE has a high false detection rate is that due to packet loss, neighbouring nodes might not always receive DIO messages. For these temporal irregularities, SVELTE has a significant false positive rate. The extended version of SVELTE has been developed by two different research, extended version of SVELTE by Matsunaga et al. (2015) and by Shreenivas et al. (2017).

2.7 Critical Review

2.7.1 Introduction

Security in IOT has been recognized to network environment. Number of IOT users are getting higher by the time. Implementation of IDS are becoming critical at the moment as the part of the basic security measurement. Critical review in this section will discussed few methodologies that was used in past research that related to Hybrid technique IDS implementation. The difference will be value based on the thread addressed, advantages, limitation of each methodology.

2.7.2 Previous Existing Product

The past research showing that proposed Hybrid IDS are different in each environment. The algorithm developed were improvised every time has passed and that research are still being continued until these days. The comparison scholar papers side by side that are comparing by IDS categories are as Table 2.1.

Table 2.1 Past research of related project

Author	Advantages	Disadvantages
Hajiheidari et al. (2019)	• Large data research • Research sources that cover main four technique of categorization of IDS. • Research sources from variety of research platform • Comparison between techniques is producing IDS metrics of over 40 paper research.	• No experiment implementation because of IOT is resource-constrained. • Direct comparison with results in the literature.
Nandhini and Mehtre (2019)	• Small data research. • Research sources that cover more than four techniques of categorization of IDS. • Literature review on related IDS technique and component.	• No experiment implementation because of IOT is resource-constrained. • Direct comparison with results in the literature.
Bouziani et al. (2019)	• Comparison between	• No technique was

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	software od IDS (SNORT, SURICATA and BRO) • Experiment implementation on the ability to detect different type of intrusion of the software. • Evaluation comparison between rules, RAM usage, drop packages, compatibility of OS etc.	specified.
Othman et al. (2018b)	• Small data research. • Research sources that cover more than four techniques of categorization of IDS. • Literature review on related IDS technique and component.	• No experiment implementation because of IOT is resource-constrained.
Perdisci et al. (2006)	• Comparison of classification module training, clustering algorithm threshold tweaking, and performance testing	• Does not have standard performance evaluation strategy for alarm correlation systems. • Does not have dataset explicitly designed for testing alarm clustering algorithms that is publicly available. • Direct comparison with results in the literature.
Jow et al. (2017)	• Identified detection latency and attack categories by using reliability and risk analysis approaches	• In perspective of computing, smart grid devices have limited resources. • In the published

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

		literature, there are no investigations on IDS reliability and risk analysis in the smart grid.

The past research made that were compilation of comparison of IDS categorization are as table Table 2.1 were the research of comparison of IDS in general. Based on past research above, the data evaluated in past research of Hybrid technique in IDS are selected as Table 2.2. The value of measurement including the true positive rate, false positive rate, accuracy detection rate, consumption of power, memory, and resources.

Table 2.2 Comparison of Hybrid technique IDS

Reference	Thread Addresses	Simulation platform	Advantages	Limitation
Raza et al. (2013)	Routing attack	Contiki/Cooja	• Real-time IDS • Flexible environment • High true positive rate • Extendable	• Detection rate is low • DoS may affect the IDS system • High false positive rate
Shreenivas et al. (2017)	Routing attack	Contiki/Cooja	• Real-time IDS • High accuracy of detection • High true positive rate • Power consumption that is low • Resource consumption that is low	• High false positive rate • High computation overhead
Sedjelmaci et al. (2016)	Wormhole, sinkhole, blackhole attacks	TOSSIM simulator	• High accuracy of detection • Low false positive rate • Latency low • Resource consumption	• High computation overhead

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

			that is low • Lightweight	
Amin et al. (2009)	DDoS attacks	SENSE simulator	• Low false positive rate • Memory consumption that is low • Lightweight	• High delay • High computation overhead
Cervantes et al. (2015)	Routing attacks	Contiki/Cooja	• Low false positive rate • Low true posive rate	• High resource consumption • Low power device usage
Matsunaga et al. (2015)	Routing attacks	Contiki/Cooja	• Low false positive rate	• Timestamp of IDS increases

2.8 Conclusion

Based on the related work and critical review, conclusion can be made past research conclude that IOT are resource-constrained which made it hard to implement IDS in the network. Related project past research also has no specific dataset, or the training effected by the constant introduction and removal. In conclusion, chapter 2 summarize the comparison of past related project and also the hybrid comparison of IDS from the related past project.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 3 : PROJECT METHODOLOGY

3.1 Introduction

In chapter 3, the methology of project will be identified and specified. The methodology of this project would use model System Development Life Cycle (SDLC) to implement the project. Every acpect in each phase will be specified and elaborated. This chapter will also highlight the milestone of the project.

In this chapter would highlight the information about the IDS that need to be compared in an experiment in order to be evaluated. The simulation software, Operating System (OS) requirement and algorithm flow will be elaborate in this chapter.

اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.2 Methodology

Project methodology is crucial in the construction process to effectively execute the project with the appropriate procedures and processes. In this topic, the best strategy is presented and discussed. To summarize this project's scope, methodology focuses on the notion of SDLC, which stands for Model System Development Life Cycle, and particularly on planning phase, analysis phase, design phase, implementation phase, testing phase, integration and analysis phase.

3.2.1 Planning Phase

Planning phase would be focus on the structure of the project to implement the comparison as output of the project. Initiating the component to be evaluated for the project, as for IDS the rate of power consumption and ETX value.

Planning phase including constructing objective based on problem statement of the project, identifying project scope and project distribution that will be gained and lastly organization report. In this phase is important to identify the limitation of project and expectation for the project.

In this project, comparison results are based on an experiment. The experiment will be designed and demonstrated in Virtual Machine (VM). The network simulation for this project would be chosen based on the source code from past research that need to be compared.

3.2.2 Analysis Phase

In analysis phase will be focus on the end-needs, and project expectations for the system and how it will function, should be identified, and recorded. The project will also undergo a feasibility analysis to see if it is organizationally, socially, and technologically feasible. This phase is based on the related work and past research evaluation.

Analysis phase would highlight the implementation and methodologies of past research and related work, for this case the comparison between past research on comparative studies of IDS implementation for general outcome. Hybrid detection method of IDS were highlighted in Table 2.2. The source code of existing computer simulation IDS that would be use to conduct the experiment are the extended version of existing hybrid detection method IDS, SVELTE, which is IDS by Matsunaga et al. (2015) and Shreenivas et al. (2017).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.2.3 Design Phase

Design phase will be focus on specification of experiment implementation.

Figure 3-1 Project software requirement (VMware Workstation Pro 16)

Figure 3-2 Project Software requirement (Contiki OS)

As for this project is about the analysis of hybrid detection method for the IDS, the Cooja network simulator will be used as the RPL simulator in a 6LowPAN environment to create nodes of IoT and the attacker such as sink node using set of source code. Cooja has been demonstrated to be an excellent tool for simulating RPL in WSNs. Cooja network simulator will be installed in an operating software Contiki. Contiki is an Ubuntu based operating system with Cooja already built in and ready to use. Contiki OS will be setup inside a virtual machine, as for this project, it is VMware Workstation Pro 16. The virtual machine was installed in a Windows OS.

The comparison between two extended version from an existing computer simulation hybrid method detection IDS by Raza et al. (2013), SVELTE, are project by Shreenivas et al. (2017) and Matsunaga et al. (2015). The IoT environment simulation specification are as below.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

i) IDS by Matsunaga et al. (2015).

Matsunaga et al. (2015) proposed two new proposals consist of upgradation of rank metrics in SVELTE in order to lowering false positive rate and timestamp of IDS in SVELTE.

Table 3.1 Simulation Parameter IDS by Matsuna et. al (2015)

Name	Simulation Parameters
Operating System	Contiki version 2.6
Simulator	Cooja
Simulation area	400x400m
Number of all nodes	32
Number of sink node	1
Number of attackers	1-4

Radio model	Unit Disk Graph Medium: Distance Loss
Mote type	Tmote Sky

i) IDS by Shreenivas et al. (2017).

Shreenivas et al. (2017) proposed two new proposals consist of upgradation of rank and ETX metrics in SVELTE in order to lowering false positive rate and geographic hints of IDS in SVELTE to improve the inconsistencies in the rank and ETX metrics.

Table 3.2 Simulation Parameter IDS by Shreenivas et. al (2017)

Name	Simulation Parameters
Operating System	Contiki version 2.6
Simulator	Cooja
Simulation area	400x400m
Number of all nodes	28
Number of sink node	1
Number of attackers	1-4
Radio model	Unit Disk Graph Medium: Distance Loss
Mote type	Tmote Sky

This project requirement is based on the specification of simulation environment from Table 3.1 and Table 3.2. As for this project, Table 3.3 and Table 3.4 below are the project simulation parameter on two Contiki OS for project experiment. The project simulation will be real-time comparison, the simulation will be on two Contiki OS.

Table 3.3 Project Simulation Parameters on Contiki A

Name	Simulation Parameters
Operating System	InstantContiki version 2.7
Contiki	A
Simulator	Cooja
Number of all nodes	26
Number of sink node	1
Number of attackers	2
Radio model	Unit Disk Graph Medium: Distance Loss
IDS by	Matsunaga et al. (2015)
Proposal 1	Rank metrics upgradation
Proposal 2	Timestamp of IDS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 3.4 Project Simulation Parameters on Contiki B

Name	Simulation Parameters
Operating System	InstantContiki version 2.7
Contiki	B
Simulator	Cooja
Number of all nodes	10, 20
Number of sink node	1
Number of attackers	2
Radio model	Unit Disk Graph Medium: Distance Loss
IDS by	Shreenivas et al. (2017)
Proposal 1	Rank and ETX metrics upgradation
Proposal 2	Geographic hints of IDS

3.2.4 Implementation Phase

Implementation phase where the stage occurs after a thorough study of the system's needs and specifications. It is the actual building procedure that follows the completion of the required system's comprehensive and illustrated design.

Source code of IDS implementation will be used inside the simulation. The simulation produced network traffic and the data, metrics, and graph of RPL that can be collect and analyze (tuz-Zahra, Zaman, & Jhanjhi, 2020). There would be the comparison of IDS by Shreenivas et al. (2017) and Matsunaga et al. (2015).

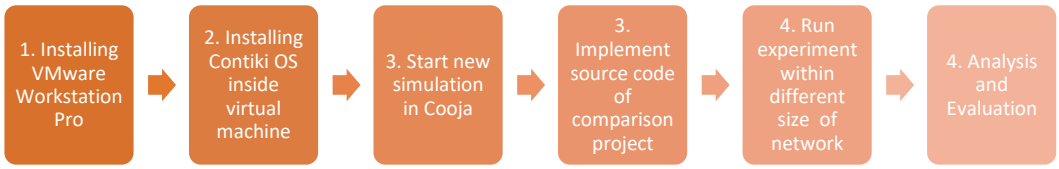


Figure 3-3 Diagram of Experiment of IDS in Cooja flow

3.2.5 Testing Phase

In testing phase, would be putting together various components and subsystems to form an integrated system, and then exposing the system to various inputs to

acquire and evaluate its outputs, behavior, and operation. The project will be examined for flaws and shortcomings at this point. The project will focus on resolving such difficulties until the result satisfies the initial requirements. In other words, the project will be checked to see if the code satisfies the requirements.

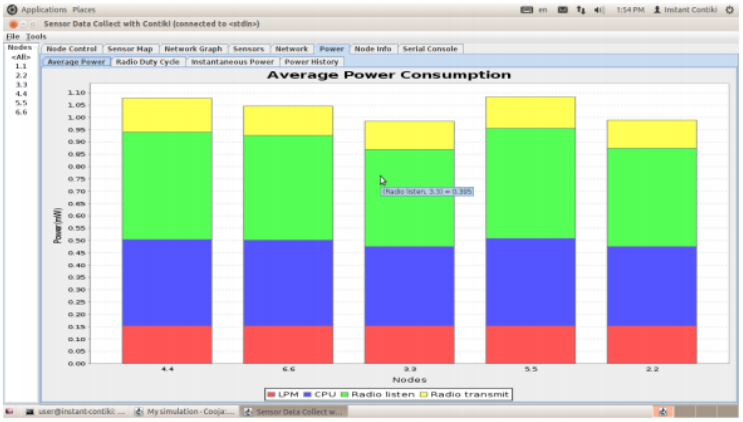


Figure 3-4 Example of average power consumption graph in Cooja

The result of false positive, false negative, number of nodes, time before an attack is collected to count the average percentage of the detection rate IDS (Nygaard, 2017). The attack from C file extension that would be evaluate is wormhole attack. There would also be data of ETX average value, power consumption graph that would be used to evaluate.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.2.6 Integration And Analysis Phase

In this phase would discuss on the data validation and integration. The result of the experiment will be computed and analysed in this phase. The data validation would be consist of the computation of ETX value and power consumption between the two IDS from Shreenivas et al. (2017) and Matsunaga et al. (2015). The power consumption can be calculated by the formula below. The final computer simulation of extended version SVELTE will be concluded.

$$\text{Power (mW)} = Energy(ml) \div Time\ (s)$$

For ETX average value also will be taken as part of the evaluation component. The forward and reverse delivery ratios of a connection are used to determine its ETX. (De Couto et al., 2003). The forward delivery ratio, *df*, is the calculated chance

that a data packet will arrive at its destination successfully; the reverse delivery ratio, *dr*, is the calculated probability that the ACK packet will arrive successfully. The expected probability that a transmission is successfully received and acknowledged is $df \times dr$. A sender will retransmit a packet that is not successfully acknowledged. Because each attempt to transmit a packet can be a success or a failure, the expected number of transmissions is as below.

$$ETX = \frac{1}{df \times dr}$$

3.3 Project Milestone

Milestone tasks can be utilized as a point of view for tracking the project's performance and verifying the venture's core operations. A project accomplishment will be described as stable and resilient in order to ensure that all activities inside a project may be maintained through the course of events of the project in order to ensure the project's smooth operation.

Figure 3-5 Milestone of project

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Week	Phase/Activities	Deliverable
1-3	1. Planning	Project Proposal
	1.1 Identifying problem statement	Report: Chapter 1
	1.2 Identifying project question	
	1.3 Identifying project objectives	
	1.4 Identifying project scope	
	1.5 Identifying project contribution/project expected outcome	
	1.6 Preparing progress report	
	1.7 Proposal presentation	
4-7	2. Analysis: Literature Review	Report: Chapter 2
	2.1 Review previous	

	research	
	2.2 Review related task	
	2.3 Preparing progress report	
8-11	3. Design: Methodology	Report: Chapter 3
	3.1 SDLC planning	
	3.2 Phase detailing	
	3.3 Preparing progress report	
	3.4 Progress presentation	
8-14	4. Design: Experiment simulation	Report: Chapter 4
	4.1 Analysis on Project Problem, Requirement, Dataset	PSM 1 presentation
	4.2 Design simulation design, data validation	
	4.3 Preparing progress report	
	4.4 Progress presentation	
	4.5 PSM 1 project presentation	
	4.6 Submission of report and logbook	
16-23	5. Implementation	Report: Chapter 5
	5.1 Source code analysis	
	5.2 Simulation setup	
	5.3 Experiment implementation	
	5.4 Preparing progress report	
16-23	6. Testing	Report: Chapter 6
	6.1 Testing experiment simulation	
	6.2 Progress report	
19-23	7. Integration and Analysis	Report: Chapter 6
	7.1 Experiment analysis on power	Report: Chapter 7

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	consumption	
	7.2 Experiment analysis on ETX value	PSM 2 presentation
	7.3 Summary of analysis	
	7.4 Defining Project Conclusion	
	7.5 Defining Project Contribution	
	7.6 Collect errors and drawbacks during experiments into project limitations	
	7.7 Extimation of extended experiment parameter for future work	
	7.8 Preparing report progress	
	7.9 Progress presentation	
	7.10 PSM 2 project presentation	
	7.11 Submission of report and logbook	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.4 Conclusion

Finally, strategies may be used to specify how procedures, components, and processes are organised, as well as architecture, division, and perspective. There are several types and techniques of unique attempt to do this. Despite the fact that the focused project has certain flaws, there are certain difficulties that can be addressed and might be replied with a bad reputation.

CHAPTER 4 : DESIGN

4.1 Introduction

In this chapter, the planning of project will be inspected to make sure the project can be done realistically. This chapter will emphasize the project prerequisite in terms of project requirement, programming specifications and the project limitation.

4.2 Problem Analysis

The Internet of Things (IoT) is the next step in the growth of the internet. Physical items may now be brought into the digital space thanks to advancements in technology. Despite the growing popularity of IoT, many individuals are unaware of its capabilities. For this project, the problem statement of the project (PS_1) as stated in chapter 1, to identify the most stable network between existing computer simulation hybrid technique Intrusion Detection System in IoT environment.

PS_1 was determined in order to proof which computer simulation existing hybrid detection method IDS that are more efficient and realistic due to different working environment. The experiment will be involving two extended version of existing computer simulation hybrid detection method IDS, SVELTE. The two extended version of SVELTE are IDS by Matsunaga et al. (2015) and Shreenivas et al. (2017).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3 Requirement Analysis

Requirement of analysis of this project are including project requirement and dataset of project where the experiment details will be elaborated.

4.3.1 Project Requirement

As stated in Chapter 3 in design phase where the software requirements are first steps to setting up the experiment environment. The specification of original IDS project by Matsunaga et al. (2015) and Shreenivas et al. (2017) have state in Chapter 3. For this project would be similar based on it. The experiment will be done in total 8 experiment and evaluation. The experiment coordinates as below:

Table 4.1 Experiment parameters

Experiment	Index	Parameter	Details	Notes
1	(a)	Number of attacker	0	• 10 node sized network • no attacker • no IDS
		Type of attack	None	
		Time when attacks occured	None	
		Number of nodes in the network	10	
		Experiment time	10 minutes	
		IDS	None	
		UDP Sender ID	1	
		UDP Client ID	3-8	
	(b)	Number of attacker	0	• 20 node sized network • no attacker • no IDS
		Type of attack	None	
		Time when attacks occured	5th Minute	
		Number of nodes in the network	20	
		Experiment time	20 minutes	
		IDS	None	
		UDP Sender ID	1	
		UDP Client ID	3-20	
2	(a)	Number of attacker	2	• 10 node sized network • 2 attacker • no IDS
		Type of attack	Malware Attack	
		Time when attacks occured	5th Minute	
		Number of nodes in the network	10	
		Experiment time	10 minutes	
		IDS	None	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

		UDP Sender ID	1	
		UDP Client ID	2-10	
		Attacker ID	11-12	
	(b)	Number of attacker	2	• 20 node sized network • 2 attacker • no IDS
		Type of attack	Malware Attack	
		Time when attacks occured	10th Minute	
		Number of nodes in the network	20	
		Experiment time	20 minutes	
		IDS	None	
		UDP Sender ID	1	
		UDP Client ID	3-20	
		Attacker ID	21-22	
3	(a)(i)	Contiki	A	• 10 node sized network • 2 attackers • 1 IDS (each experiment)
		Number of attacker	2	
		Type of attack	Malware Attack	
		Time when attacks occured	5th Minute	
		Number of nodes in the network	10	
		Experiment time	10 minutes	
		IDS	Matsunaga et. al.(2015)	
		UDP Sender ID	2	
		UDP Client ID	3-10	
		Attacker ID	11-12	
		IDS ID	1	
	(a)(ii)	Contiki	B	
		Number of attacker	2	
		Type of attack	Malware Attack	
		Time when attacks occured	5th Minute	
		Number of nodes in the network	10	
		Experiment time	10 minutes	
		IDS	Shreenivas et. al. (2017)	
		UDP Sender ID	2	
		UDP Client ID	3-10	
		Attacker ID	11-12	
		IDS ID	1	
	(b)(i)	Contiki	A	• 20 node sized network
		Number of attacker	2	
		Type of attack	Malware	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

			Attack	• 2 attackers • 1 IDS (each experiment)
		Time when attacks occured	10th Minute	
		Number of nodes in the network	20	
		Experiment time	20 minutes	
		IDS	Matsunaga et. al.(2015)	
		UDP Sender ID	1	
		UDP Client ID	3-20	
		Attacker ID	21-22	
		IDS ID	23	
	(b)(ii)	Contiki	B	
		Number of attacker	2	
		Type of attack	Malware Attack	
		Time when attacks occured	10th Minute	
		Number of nodes in the network	20	
		Experiment time	20 minutes	
		IDS	Shreenivas et. al.(2017)	
		UDP Sender ID	1	
		UDP Client ID	3-20	
		Attacker ID	21-22	
		IDS ID	23	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The Operating System Contiki already have the cooja simulator installed, the experiment will be using SVELTE source code as base since both of the extended version project IDS are based on SVELTE. The source code of SVELTE were provided on Raza et al. (2013) research paper.

4.3.2 Dataset

The source code that will be use in this project would be the existing computer simulation SVELTE by Raza et al. (2013), where the source code of IPV6 Border Router (6BR) inside the IPv6 over Low -Power Wireless Personal Area Networks (6LowPAN) are to specified the rank metrics of the IoT nodes in, next the IDS client and also the malicious node. The source code is in programming C class file.

The dataset of the project is the component of evaluation of the experiment, as below:

1. Power Consumption of experiment
2. ETX average value of experiment

The dataset was collected by the sink node of each experiment. The dataset was developed in each different experiment environment. The dataset is collected in Cooja feature of 'Collect-view'. The data are summarized in a table (Node Info). Based on the data set of SVELTE, the source code will be manipulated as algorithm of both extended version of SVELTE to be use as the experiment source code.

4.3.2.1 IPV6 Border Router (6BR)

Source code for the 6BR are located in file *mapper.c* where the proposals of both Matsunaga et al. (2015) and Shreenivas et al. (2017) proposed the rank metrics and other metrics.

i) Matsunaga et. al (2015)

- ***Rank Metrics***

  In order to reduce false alarm caused by the timing inconsistency. This algorithm working when the rank mismatch that was caused by the rank mismatch, the rank will be reported to neighbour nodes instead of its current rank.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

- ***Timestamp of IDS***

  In order to account for time inconsistencies, each node adds a timestamp to the sink node. The threshold for this algorithm is k_1 and k_2 for the timestamp of the sink node.

ii) Shreenivas et. al. (2017)

- **Rank and ETX metrics**

  In this file were implemented as 6BR in the IoT environment simulation. This file contains the rank metrics that were specified to work as IDS in the simulation. The algorithm for this file as Figure where it was made to verify the propagation of ETX inside the 6LowPAN network. This file is to identify intruders that publish fake ETX numbers in order to increase

their strength or execute DoS attacks We additionally protect against root spoofing attacks by comparing the rank values to the ETX value.

```
Require: N - Set of nodes
Require: P - Parent set of the node
Require: Neighbors - Neighbor set of Node N
  for Node in N do
    for all Neighbor in Node.Neighbors do
      if Node.etx == 0 then
        if Parent.etx == 0 then
          if Node.Rank == Root.Rank then
            Node.fault = Node.fault + 1 {The node is trying to
            advertise a root etx value}
          end if
        end if
      end if
      if Node.etx > 0 then
        if Node.etx < Parent.etx then
          if Node.Rank > Parent.Rank +
          MinHopRankIncrease then
            Node.fault = Node.fault + 1 {The node is trying to
            advertise an invalid etx value}
          end if
        end if
      end if
    end for
  end for
  for Node in N do
    if Node.fault > FaultThreshold then
      A new parent is chosen
    end if
  end for
```

Figure 4-1 Algorithm proposal 2 Shreenivas et. al. (2017)

- **Geographical Information inside 6LowPAN**

  In this file were made to differentiate the inconsistency of rank in the simulation to overcome rank and ETX attacks. The algorithm for attempting to group the nodes in the simulation with transmission power restriction in order to determine their closest neighbours. The objective of this approach is to identify intruders who pose as legitimate authority in order to execute numerous attacks in an IoT environment. The algorithm compute the transmission limits for each node in the network and keep a neighbour database that lists the identities of the nodes within their transmission range.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Require: N - Set of nodes participating in the 6LoWPAN networks
Require: Tx - Nodes within the receiving vicinity
Require: NT- Neighbor table listing a collection of nodes
Require: $Neighbor$ - Neighbor of the Node N
  for all $Node$ in N **do**
    $NT = nodeswithin|Node.Tx|$
    if $NodeinNT$ **then**
      $Node.ETX < Node.Neighbor.ETX$
    end if
  end for

Figure 4-2 Algorithm proposal 2 Shreenivas et. al. (2017)

4.4 Project Design

For project design, the experiment are based on extension of existing computer simulation IDS named SVELTE by Raza et al. (2013). As stated in Chapter 3, simulation structure of project by Shreenivas et al. (2017) are the same seed as SVELTE (Shreenivas et al., 2017) while project by Matsunaga et al. (2015) using more nodes and number of attackers. However, in this project will be in same simulation environment with different rank metrics.

The component that needs to be compared between the two extensions of SVELTE are the rank and ETX updates as both project proposed different extension metrics. The difference of the two projects will be evaluated in an experiment of sinkhole attack. In order to compare between the two IDS simulation, the simulation of IDS by Shreenivas et al. (2017) and Matsunaga et al. (2015) will be on different virtual machine (Contiki) but with the same seed and attack simulation.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.4.1 Simulation Design

The simulation of the project would be created in Cooja. The steps require to create a Cooja new simulation are first to create name of the simulation, then for the advance settings need to choose radio medium, and also set the mote startup delay and random seed. The general steps of this project simulation will be discussed in this part.

Table 4.2 Creating simulation example

Name	Details
Simulation Name	PSM – Experiment 1(a)
Radio Medium	1000000

Random seed	123456

The simulation of IDS has the project saved as *csc* file project. The IDS motes were different based on each algorithm project. Below is the simulation of non-IDS sinkhole attack in Cooja network simulator. The motes were positioned in rank trees. There were three Tmote Sky inside the network simulator with the mote source file as nodes.

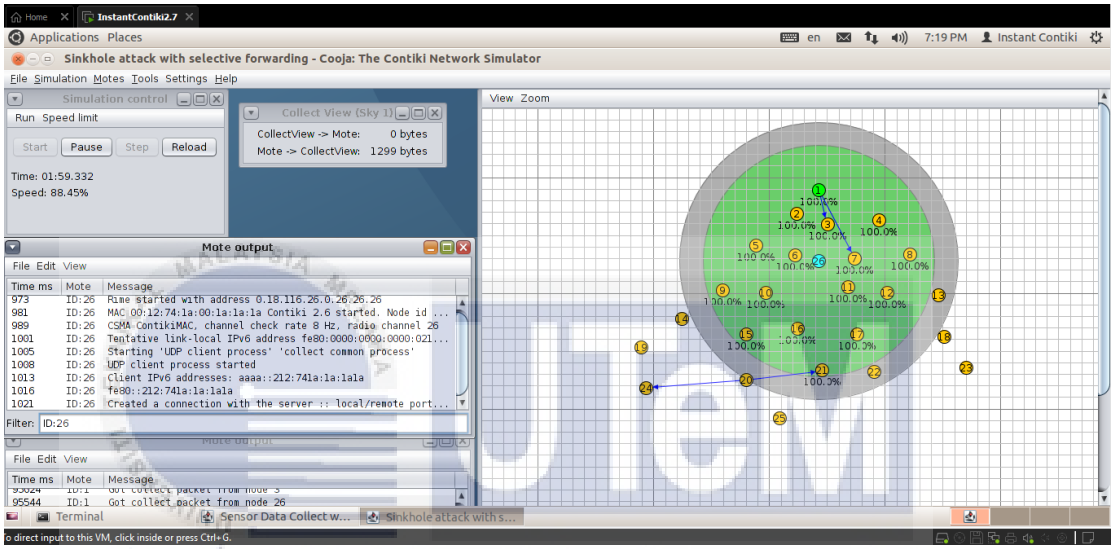


Figure 4-3 attack_sinkhole_ids_demo.csc in Cooja

Mote ID 1 (#sk1) compiled and creating mote by the command ‘make network-mapper.sky TARGET=sky’, mote ID 2 to 25 (#sky2) compiled and creating mote by the command ‘make network-mapper-client.sky TARGET=sky’ and mote ID 26 (attacker) compiled and creating mote by the command ‘make udp-sender.sky TARGET=sky’.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

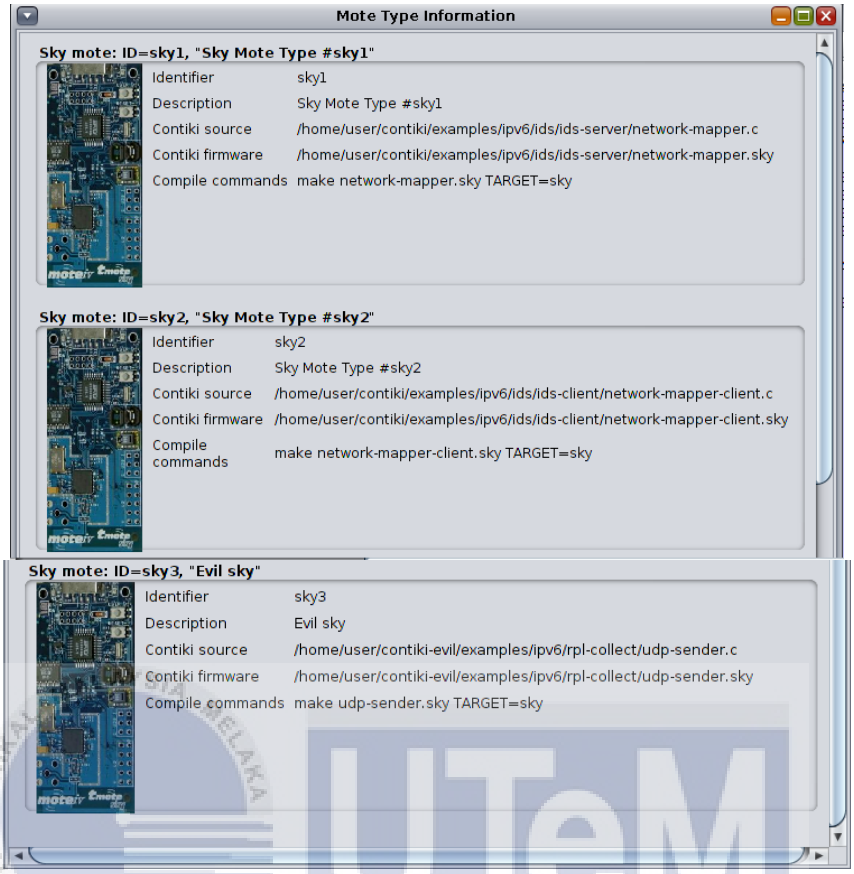


Figure 4-4 Mote Type Information

Mote ID 1 (green) are the 6BR (Border Router) where the module of rank and ETX metrics were implemented (Raza et al., 2013). The structure of the motes are as Figure 4-2 below, where the 6BR (green) are located on top of the motes and followed by the attacker node (purple) and random IDS client notes(yellow) located below it.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

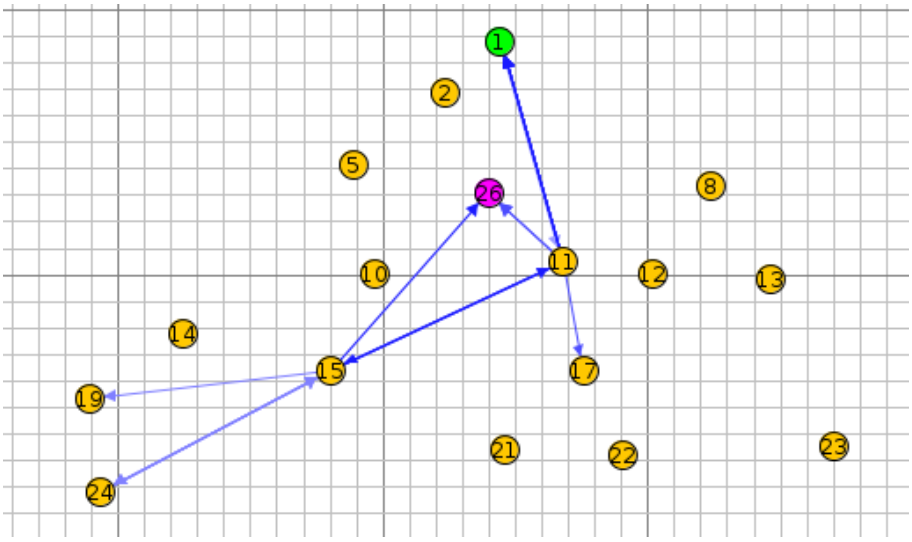


Figure 4-5 Motes in network windows in Cooja (Raza et. al.)

4.4.2 Data Validation

The results of experiment for this project will be validated based on the specifications according to the theoretical claims and also the research paper. The result of project that claimed by Shreenivas et al. (2017) are based on the power consumption. For the result of project claimed by Matsunaga et al. (2015) will compared with this project based on the power consumption as well.

Power consumption data validation can be evaluated within the graph that Cooja will produce when the simulation is working. The percentage of this project and claimed by product evaluated will be compared. As mentioned in paper, Shreenivas et al. (2017) claimed that the power consumption 0.02% increment of power consumption of project. For this project the power consumption graph can be found at collecting view of node's information.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

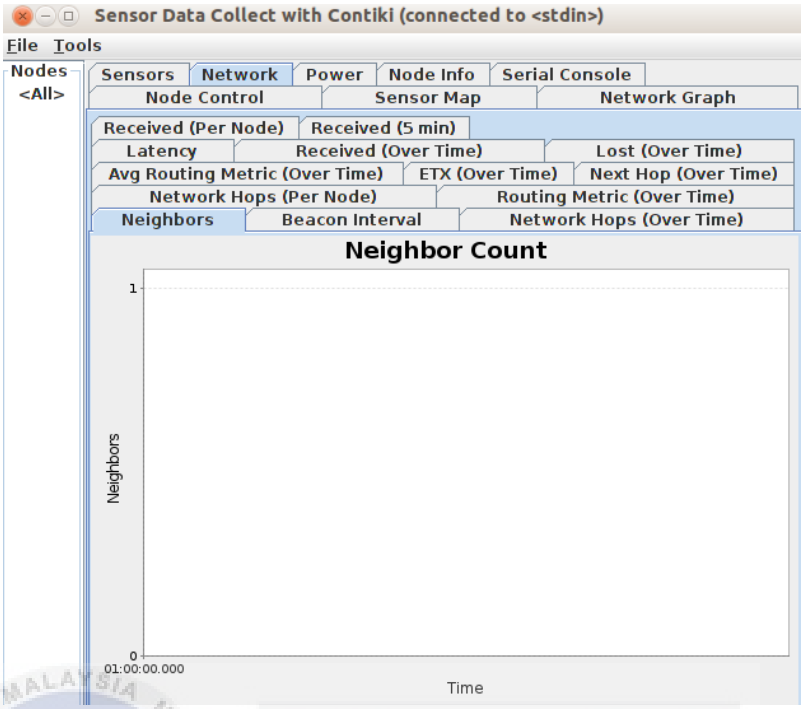


Figure 4-6 Collect View of Sky mote

False positive rate is considered in IDS when there are slight changes in the rank of two nodes that supposed to be non-malicious. It also can be when non-malicious nodes use the malicious nodes' rank to determine their own rank. After computing the inaccurate rank, a node with the right rank observes the node, which can be results in a negative observation. However the execution may be flawed.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.5 Conclusion

After emphasizing the project prerequisite in terms of software requirement, programming specifications and the project limitation. The project can be done realistically based on the software requirement, simulation design and data validation. Although there would be limitation on how the experiment would work as the data in the experiment may be vary from the validation comparison.

CHAPTER 5 : IMPLEMENTATION

5.1 Introduction

The previous chapter discussed over project design, and this chapter will go over project implementation. This chapter will enlighten on how the simulation environment will be configured based on the source of main Intrusion Detection System (SVELTE) according to IDS by Matsunaga et. al. (2015) and Shreenivas et. al. (2017).

5.2 Source Code Of IDS

The source code IDS are based on the SVELTE that was improvised by Matsunaga et. al.(2015) and Shreenivas et. al. (2017). The IDS contained IDS rules that were applied when the IDS are working in the simulation.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

```
void
detect_inconsistencies()
{
  detect_correct_rank_inconsistencies();
  check_child_parent_relation();
  missing_ids_info();
}
```

Figure 5-1 Code Snippet of IDS rules

Figure 5-1 above are the snippet of IDS rules code that are being implemented in SVELTE. The function *detect_correct_rank_inconsistencies* are the function to produce the true positive rate of the IDS and promote the inconsistencies if nodes are persistent over several consecutive mapping intervals. The function *missing_ids_info* is where the false information or if some node has as of yet not sent information about its neighbors, the IDS consider it a fault and alert the user. The function

check_child_parent_relation are the function to check that all information provided by nodes correspond with the information provided by their parent which are the rank metrics of the IDS.

In Matsunaga et. al.(2015) IDS, the IDS are improvised SVELTE's rank metrics by the constant *K* value that holds new update rank value for the metrics. While Shreenivas et. al. (2017) using replace rank metrics by SVELTE and introduce new metrics, ETX metrics where it indicates the communication quality of the neighbours.

5.3 Project Simulation Setup

Cooja project simulation are initiated by the running the file *home/user/contiki/tools/cooja/build.xml* on terminal of the Contiki, the experiment proceeds with creating a network simulation environment by the menu file>create new simulation.

Inside the Cooja new simulation created, there are few tabs displayed, network windows, simulation control, mote output, notes and timeline. The experiment setup firstly by adding mote in Tmote Sky and the source file. Below are the source code of file that has been used for all the 8 experiments:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 5.1 Project Simulation File Path

Motes	Experiment	File Path Name
UDP Server (sink node)	1 (a)	Contiki-IDS/examples/ ipv6/rpl-collect/udp-sink.c
	1 (b)	
	2 (a)	
	2 (b)	
	3 (a)(i)	
	3 (a)(ii)	
	3 (b)(i)	
	3 (b)(ii)	
UDP Sender	1 (a)	Contiki-IDS/examples/ ipv6/rpl-collect/udp-sender.c
	1 (b)	
	2 (a)	
	2 (b)	
	3 (a)(i)	
	3 (a)(ii)	
	3 (b)(i)	
	3 (b)(ii)	
Attacker	2 (a)	ab/examples/Untitled

	2 (b)	Folder/2.c
	3 (a)(i)	
	3 (a)(ii)	
	3 (b)(i)	
	3 (b)(ii)	
IDS	3 (a)(i)	Contiki-IDS/examples/ ipv6/ids/ids-server/network-mapper.c
	3 (a)(ii)	
	3 (b)(i)	
	3 (b)(ii)	

5.3.1 Experiment 1: No Attacker And No IDS Setup

There would be two times experiment 1 will be executed. The first time was simulated with total 10 nodes, the second time are simulated with total of 20 nodes. The number of nodes in this experiment influenced the accuracy and efficiency of the simulation in different size of network. This method to collect the data on how the RPL network is working on normal environment without any attacks or Intrusion Detection System. The experiment are involved with a UDP server and UDP client.

UDP Server or Sink node used in this experiment are named *udp-sink.c* where the source file contained the function to collect data from other nodes as well it worked as UDP server. The sink node can prove that a node may not inform the accurate number of packets for each time frame. The decision step helps to determine malicious nodes while minimizing the rate of false positives and false negatives. UDP Client or UDP sender in this experiment are named *udp-sender.c* where the source file contained the function to communicate and share information with UDP server by using the CoAP routing protocol.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.3.1.1 10 Nodes: RPL Network With No Attacker And No IDS

In this experiment the nodes total are set as 10 where the position of each nodes are sequenced according as a RPL network as in Figure 5-2.

Table 5.2 Experiment 1(a) parameters

Number of attacker	0
Type of attack	None
Time when attacks occured	None
Number of nodes in the network	10
Experiment time	10 minutes
IDS	None

UDP Sender ID	1
UDP Client ID	2-8

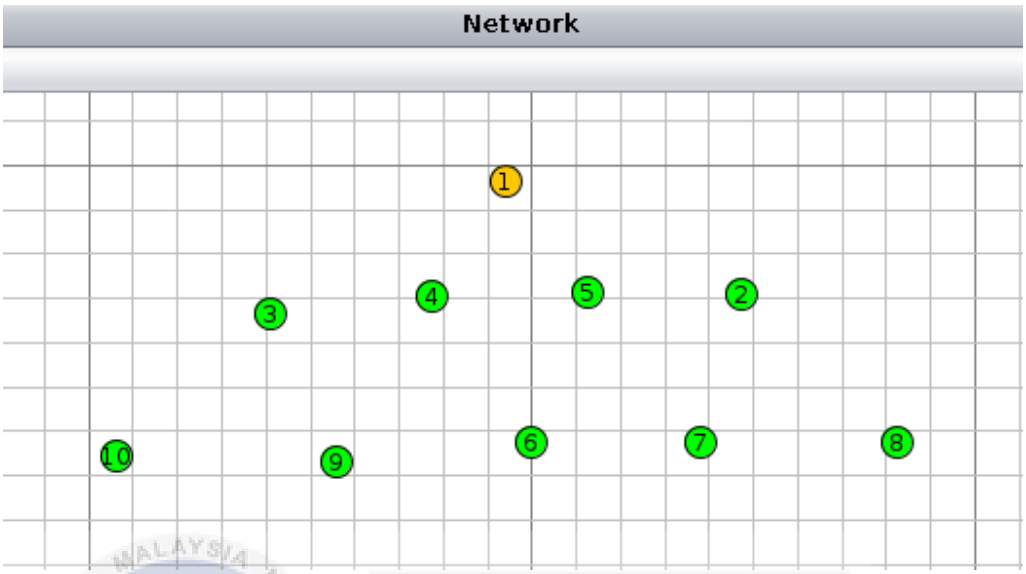


Figure 5-2 Experiment 1(a) Network Structure

5.3.1.2 20 NODES: RPL NETWORK WITH NO ATTACKER AND NO IDS

In this experiment the nodes total are set as 20 where the position of each nodes are sequenced according as a RPL network as in Figure 5-3.

Table 5.3 Experiment 1(b) parameters

Number of attacker	0
Type of attack	None
Time when attacks occured	None
Number of nodes in the network	20
Experiment time	10 minutes
IDS	None
UDP Sender ID	1
UDP Client ID	2-20

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

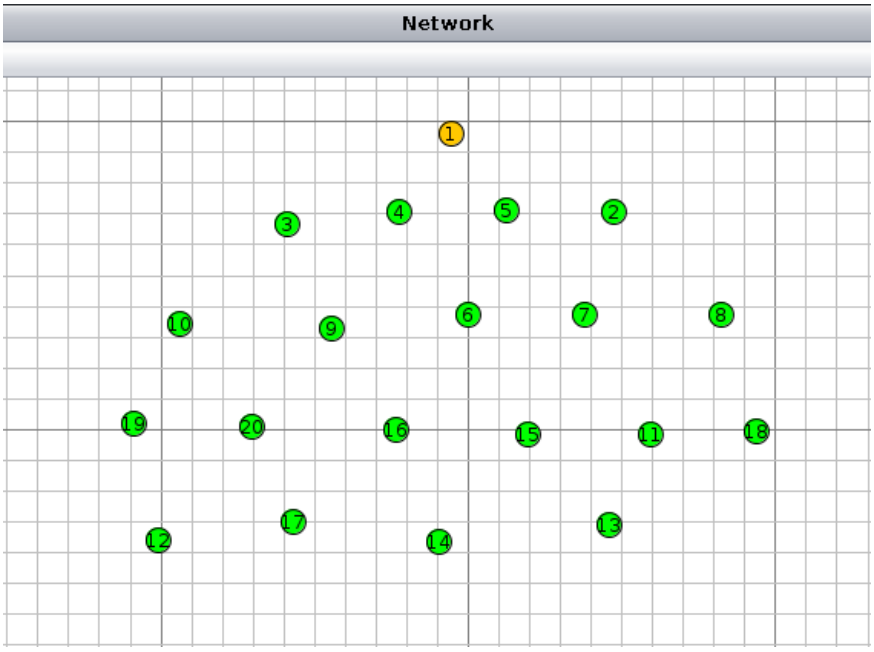


Figure 5-3 Experiment 1(b) Network Structure

5.3.2 Experiment 2: Two Attacker And No IDS Setup

There would be two times experiment 2 will be executed. The first time was simulated with total 10 nodes, the second time are simulated with total of 20 nodes. The number of nodes in this experiment influenced the accuracy and efficiency of the simulation in different size of network. This method to collect the data on how the RPL network is working on normal environment without any attacks or Intrusion Detection System. The experiment are involved with a UDP server, UDP client and two attackers.

The number of attackers are doubled in order to get data that is visible to the energy consumption chart and network traffic so that the comparison easily made. The attack that has been made are malware attack, source code by (Pongle and Chavan, 2015).

5.3.2.1 10 Nodes: RPL Network With Two Attackers And No IDS

In this experiment the nodes total are set as 10 where the position of each nodes are sequenced according as a RPL network as in Figure 5-4.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 5.4 Experiment 2(a) parameters

Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	5th Minute
Number of nodes in the network	10
Experiment time	10 minutes
IDS	None
UDP Sender ID	1
UDP Client ID	2-10
Attacker ID	11-12

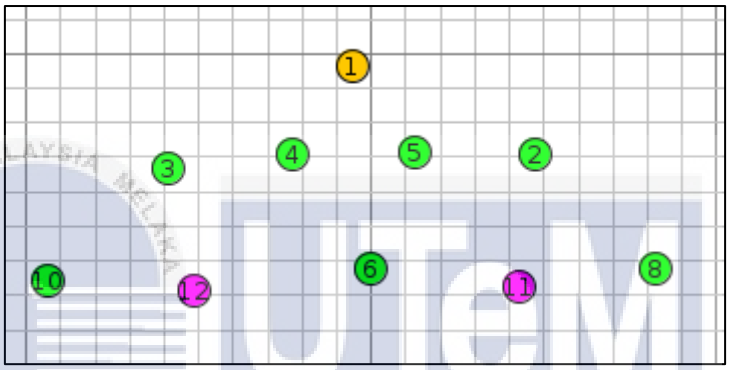


Figure 5-4 Experiment 2(a) Network Structure

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.3.2.2 20 NODES: RPL NETWORK WITH TWO ATTACKERS AND NO IDS

In this experiment the nodes total are set as 20 where the position of each nodes are sequenced according as a RPL network as in Figure 5-5.

Table 5.5 Experiment 2(b) parameters

Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	10th Minute
Number of nodes in the network	20
Experiment time	20 minutes
IDS	None
UDP Sender ID	1
UDP Client ID	3-20
Attacker ID	21-22

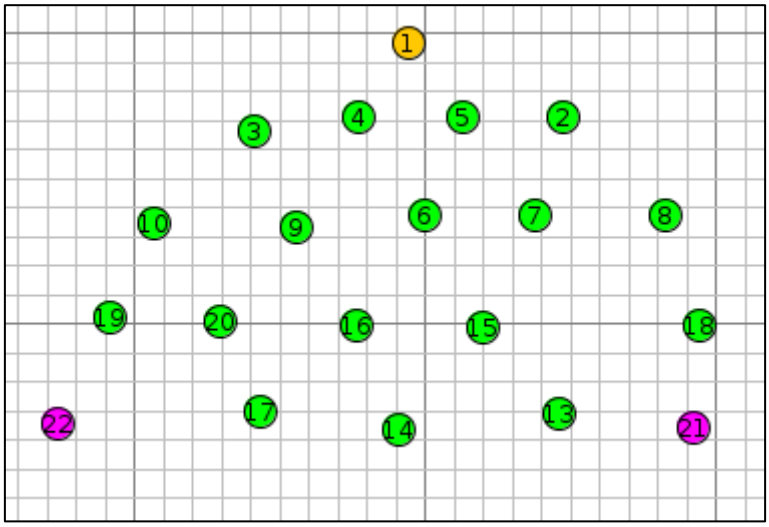


Figure 5-5 Experiment 2(b) Network Structure

5.3.3 Experiment 3: Two Attackers With IDS Setup

There would be two times experiment 3 will be executed. The first time was simulated with total 10 nodes, the second time are simulated with total of 20 nodes. The number of nodes in this experiment influenced the accuracy and efficiency of the simulation in different size of network. This method to collect the data on how the RPL network is working on normal environment without any attacks or Intrusion Detection System. The experiment are involved with a UDP server, UDP client, two attackers and an IDS. In this experiment will be involved two different IDS.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.3.3.1 10 Nodes: RPL Network With Two Attackers With IDS

In this experiment the nodes total are set as 10 where the position of each nodes are sequenced according as a RPL network as in Figure 5-6. Both IDS by Matsunaga et. al.(2015) and Shreenivas et. al.(2017) are using the same network structure.

Table 5.6 Experiment 3(a) parameters - Contiki A

Contiki OS	A
Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	5th Minute
Number of nodes in the network	10
Experiment time	10 minutes
IDS by	Matsunaga et. al. (2015)
UDP Sender ID (Sink node)	2

UDP Client ID	3-10
Attacker ID	11-12
IDS	1

Table 5.7 Experiment 3(a) parameters - Contiki B

Contiki OS	B
Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	5th Minute
Number of nodes in the network	10
Experiment time	10 minutes
IDS by	Shreenivas et. al. (2017)
UDP Sender ID (Sink node)	2
UDP Client ID	3-10
Attacker ID	11-12
IDS	1

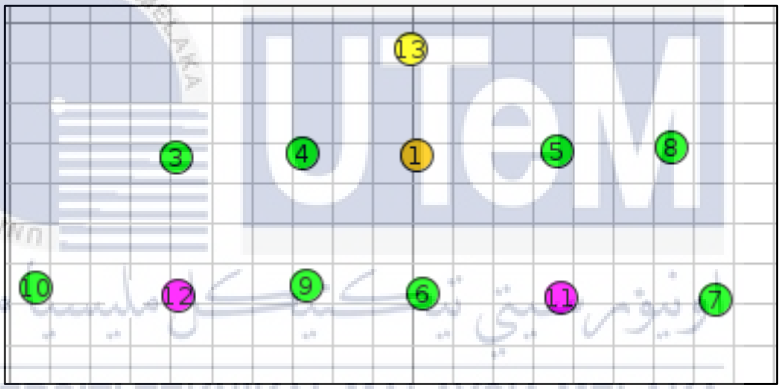


Figure 5-6 Experiment 3(a) Network Structure

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.3.3.2 20 NODES: RPL NETWORK WITH TWO ATTACKERS WITH IDS

In this experiment the nodes total are set as 20 where the position of each nodes are sequenced according as a RPL network as in Figure 5-6. Both IDS by Matsunaga et. al.(2015) and Shreenivas et. al.(2017) are using the same network structure.

Table 5.8 Experiment 3(b) parameters - Contiki A

Contiki OS	A
Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	5th Minute
Number of nodes in the network	22
Experiment time	14 minutes

IDS by	Matsunaga et. al. (2015)
UDP Sender ID (sink node)	1
UDP Client ID	3-20
Attacker ID	21-22
IDS	23

Table 5.9 Experiment 3(b) parameters - Contiki B

Contiki OS	B
Number of attacker	2
Type of attack	Malware Attack
Time when attacks occured	5th Minute
Number of nodes in the network	22
Experiment time	14 minutes
IDS by	Shreenivas et. al. (2017)
UDP Sender ID (sink node)	1
UDP Client ID	3-20
Attacker ID	21-22
IDS	23

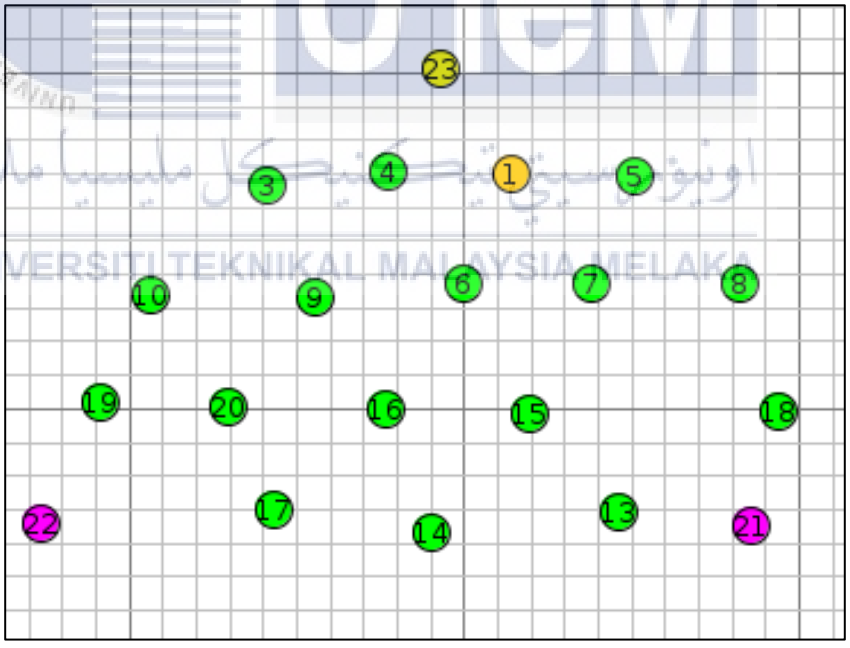


Figure 5-7 Experiment 3(b) Network Structure

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.4 Conclusion

In conclusion, the implementation phase comprises detailing in depth on how the project development is evolving. The methods and procedures for this project are detailed and step-by-step in order to ensure that it runs smoothly. Furthermore, the

environment and configuration setup must be established thoroughly during this step to avoid any false implementation.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 6 : TESTING AND ANALYSIS

6.1 Introduction

The project's testing procedure is described in the sixth chapter. Furthermore, testing is essential to confirm that the finished product and system fulfil the requirements and function properly. In addition, this step will strengthen the project's involvement in achieving the project's goal. The texting will be involving the experiments that will be done in this project. There would be eight experiments in total. All components and modules will be verified to guarantee that the experiment is done according to the project requirement and precisely.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

6.2 Testing And Analysing Method

6.2.1 Testing Experiment

The Cooja simulator run in total of eight file simulator of the experiment. The experiment included Experiment 1 (a) and (b), Experiment 2 (a) and (b), Experiment 3 (a) and (b) for both Matsunaga et. al (2015) and Shreenivas et. al. IDS (2017).

6.3 Analysis Experiment

The analysis of the experiment in this project are based on the number of nodes of the experiment that is 10 nodes and 20 nodes. There are total of eight experiment that have been done in this project.

6.3.1 Power Consumption

6.3.1.1 Experiment Of 10 Nodes

In this experiment, experiment 1(a), experiment 2(a) experiment 3(a)(i) – contiki A, and experiment 3(a)(ii)- contiki B are being compared within the 10 nodes power consumption.

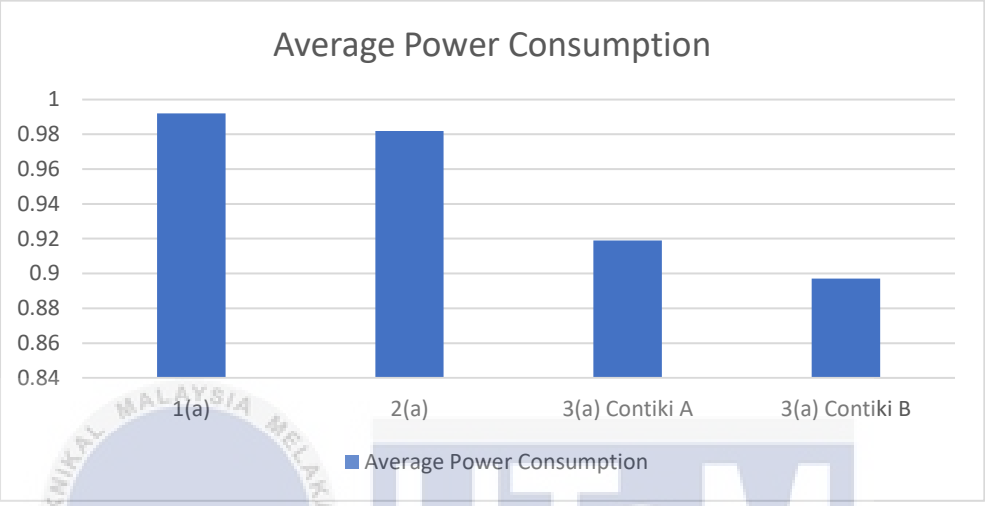


Figure 6-1 Average Power Consumption of 10 nodes

The graph showing that IDS by Shreenivas et. al. (2017) are consuming lowest power consumption than other 10 node implementation. IDS by Matsunaga et. al.(2015) are the second lowest among the 10 node experiment. The Matsunaga et. al.(2015) IDS implementation where the module improvised *k* threshold for the rank metrics are consuming more power than IDS ETX metrics by Shreenivas et. al. (2017). IDS by Shreenivas et. al. (2017) consume power lesser by 0.78% than Matsunaga et. al.(2015).

6.3.1.2 EXPERIMENT OF 20 NODES

In this experiment, experiment 1(b), experiment 2(b), experiment 3(a)(i) – contiki A, and experiment 3(a)(ii)- contiki B, are being compared within the 20 nodes power consumption.

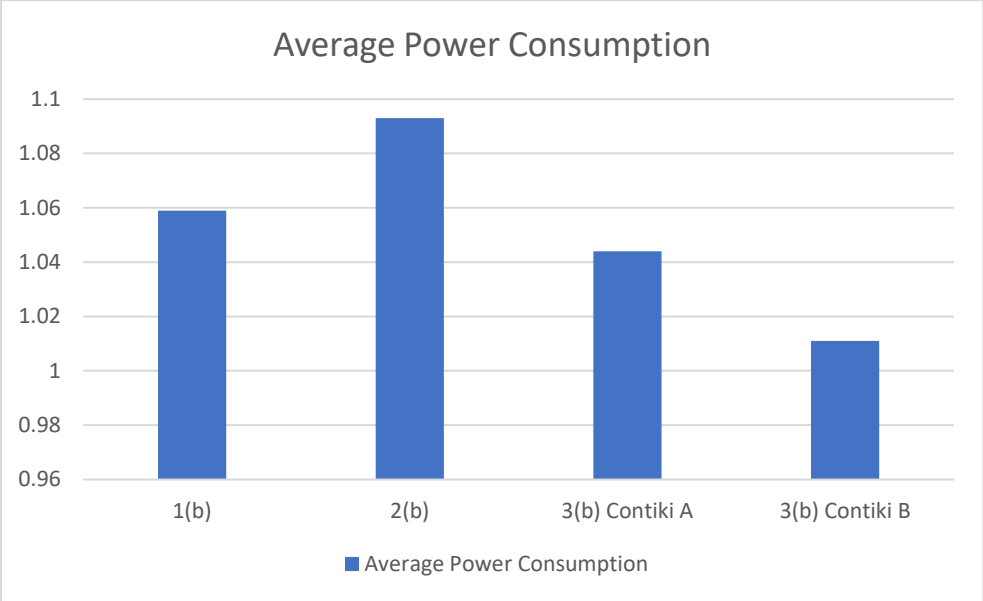


Figure 6-2 Average Power Consumption of 20 nodes

The graph showing that IDS by Shreenivas et. al. (2017) are consuming lowest power consumption than other 20 node implementation. IDS by Shreenivas et. al. (2017) consume power lesser by 0.25% than Matsunaga et. al.(2015). The Matsunaga et. al.(2015) IDS implementation where the module improvised *k* threshold for the rank metrics are consuming more power than IDS ETX metrics by Shreenivas et. al.(2017).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

6.3.2 Average Of ETX Value

6.3.2.1 Experiment Of 10 Nodes

In this experiment, experiment 1(a), experiment 2(a), experiment 3(a)(i) – contiki A, and experiment 3(a)(ii)- contiki B are being compared within the 10 nodes of ETX average value.

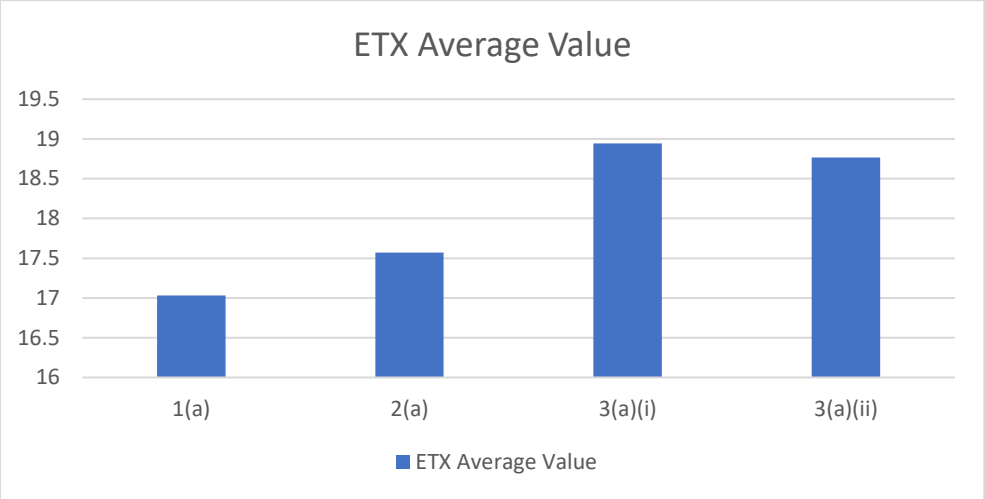


Figure 6-3 Average ETX value of 10 nodes

The graph showing that Experiment of 1(a), experiment of no attacker and no IDS have lowest ETX value than other 20 node implementation. This is because the network has no event occurred within the experiment. IDS by Shreenivas et. al. (2017) use ETX value lesser by 0.25% than Matsunaga et. al.(2015). This shows that in 10 node size network, IDS by Shreenivas et. al. (2017) have more stable and quality network link than IDS by Matsunaga et. al.(2015).

6.3.2.2 EXPERIMENT OF 20 NODES

In this experiment, experiment 1(b), experiment 2(b), 3(a)(i) – contiki A, and experiment 3(a)(ii)- contiki B, are being compared within the 20 nodes of ETX average value.

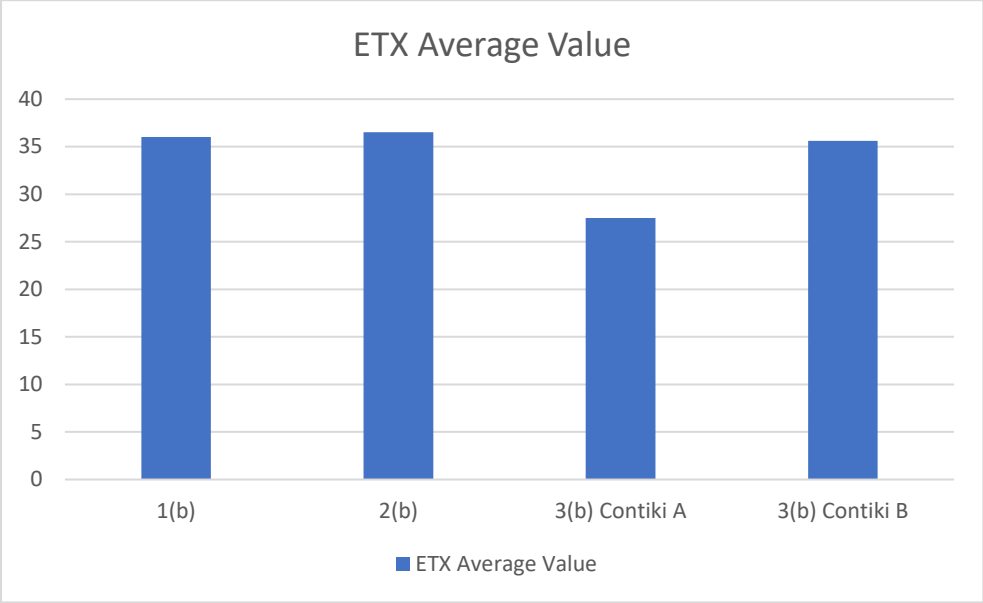


Figure 6-4 Average ETX value of 20 nodes

The graph showing that IDS by Matsunaga et. al.(2015) have lowest power consumption than other 20 node implementation. IDS by Matsunaga et. al.(2015) use ETX value lesser by 5.99% than Shreenivas et. al. (2017). This shows that in 20 node size network, IDS by Matsunaga et. al.(2015) have more stable and quality network link than IDS by Shreenivas et. al(2017).

6.4 Summary Of Analysis

From the experiment of the small and larger network size for average power consumption, the comparisons are as below graph, where the power consumption of 20 nodes are higher than 10 nodes value. This situation occurs since the size of network affect the 6Mapper response handling in the network causing more energy used. The possible reasons why Experient 3(ii) is low among all is because the energy consumed by sending the additional ETX value is negligible when compared with the overall energy usage by the network (Shreenivas et al., 2017).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

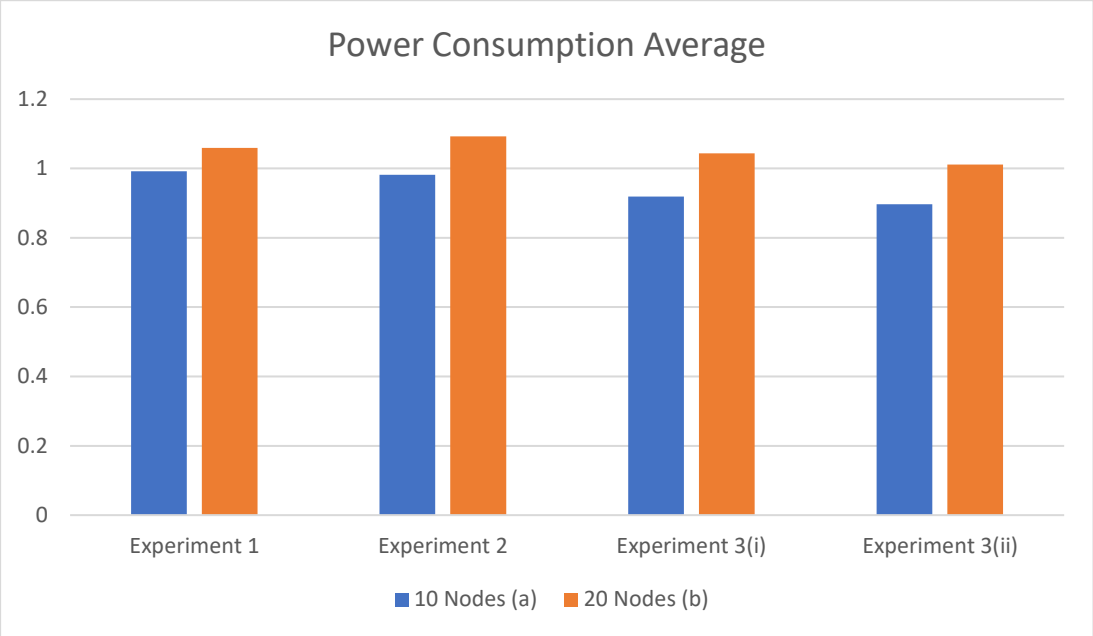


Figure 6-5 Power Consumption in 10 nodes and 20 nodes

From the experiment of the small and larger network size for ETX average value, the comparisons are as below graph, where the ETX value of 20 nodes are higher than 10 nodes value. This situation occurs since the size of network affect the network quality value. The possible reasons why Experient 3(ii) is higher than Experiment 3(i) is because the ETX metrics proposed in the IDS Shreenivas et al. (2017), where the false ETX value advertised to attract attacker to think that the network have a weaker link, the high ETX value while a stronger link would have higher network quality, the lower ETX value. The metrics is a method to detect attackers.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

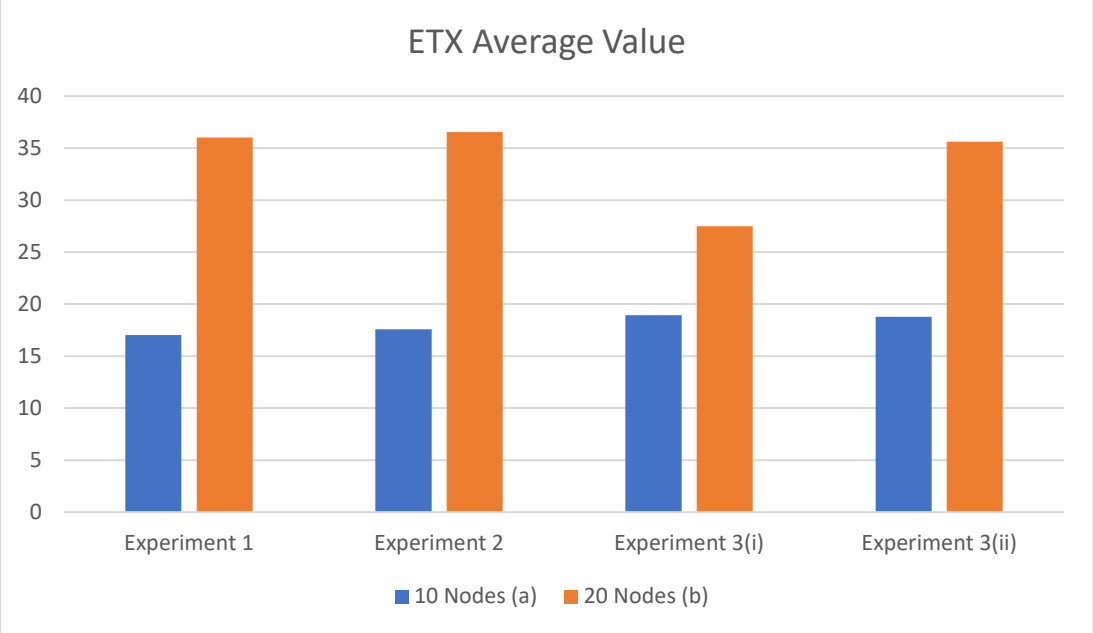


Figure 6-6 ETX value in 10 nodes and 20 nodes

6.5 Conclusion

The conclusion of the analysis is IDS by Shreenivas et. al. (2017) in 10 and 20 nodes network structure have the better power consumption as it consume less than Matsunaga et. al.(2015), which means the network node of Shreenivas et. al. (2017) have lower temperature value as it consume lesser power. However in ETX average value, in Shreenivas et. al. (2017) IDS have higher ETX value rather than Matsunaga et. al (2015) on smaller network structure (10 nodes), but higher in larger network structure (20 nodes), which means IDS by Shreenivas et. al. (2017) can achieve better network stability in a small network rather than larger network if to be compared with Matsunaga et. al (2015).

In conclusion, the test phase is carried out in order to assess the simulated outcome of the experiment. Within the 10 node and 20 node experiments, the power usage varies. The smaller size network, the power consumption would be smaller (Raza et al., 2013). IDS aids in the reduction of power consumption. IDS metrics also varies in terms of the ETX average value, which is IDS can effect on the stability of the network, as the lesser the ETX value, the more stable and quality are the network link. As a result, the following chapter will include project summary, involvement, limitations, and future implementation.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 7 : PROJECT CONCLUSION

7.1 Introduction

This chapter will summarize the project's conclusion and progress. It will discuss the overall project progress and achievement of this project's contribution, as well as the project's capabilities, weaknesses, and future improvements. Furthermore, every project specific will be clarified and comprehensible by giving the project overview. This chapter will also go through the changes that will be made and how they will be implemented for the next phase of the experiment.

7.2 Project Summarization

The objective of the project are to investigate the implementation of hybrid detection method Intrusion Detection System in simulation IoT, 6LoWPAN network environment. Hybrid IDS implementation in IoT environment in this case, as a computer simulation. In this project, the implemenation of IoT computer simulated environment can be done by using network simulator tools, such as Cooja, C/C++, SENSE, Raspberry Pi, TineyOS, TOSSIM and many more. For this project implementation are using Cooja, where it produce network traffic, nodes information using the Cooja futures. The only requirement needed are the source code file C to be use as the motes, which comes in few modes (Tmote Sky mode, Cooja mote, Z1 mode).

Next, objective to compare the effectiveness between the hybrid detection method of IDS(s) in simulated IoT environment based on the experiment analysis. For this project, there are total 8 experiments are done. The analysis results of the comparison between IDS by Matsunaga et al. (2015) and Shreenivas et al.(2017) showing that Shreenivas et al. (2017) are better at reducing the power consumption

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

in small and larger network, while Matsunga et sl.(2015) are better at keeping network quality better in a larger network.

Lastly the objective to make recommendations of best simulation environment settings based on the experiment analysis of hybrid detection method IDS in simulated IoT, 6LoWPAN network environment. The best simulation settings to run the source code are within the same version Contiki, Contiki 2.7, as for this project instant Contiki 2.7. Few following errors occurred when using the 'Collect-view' in cooja are due to missing files of MoteFinder.java and motelist-linux in Contiki/tools where the file are in directory *tools/collect-view/src/org/contikios/Contiki/collect* and */tools/sky/motelist-linux* respectively.

Hybrid IDS in this project are investigated within the use of modules and IDS rules inside the source code. The module was used are compared within the power consumption and the ETX value. However, the true positive and false negative rate data are missing. Simulation environments vary within the time executed and possibly the position of each nodes. The result averages are however used as the data for the experiment to be compared. In conclusion two over three of the experiment objectives achieved.

UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

7.3 Project Contribution

This project was developed to compare hybrid intrusion detection that was improvises from previous hybrid IDS by SVELTE. The comparison is made as part of effort for future work of computer simulation of improvised SVELTE hybrid IDS. All these functions produced contribute a lot of new method to implement IDS modules inside a computer simulation. Therefore, in conclusion, this experiment project would help to compare between the improvised version of SVELTE IDS that is by Matsunaga et. al. (2015) and Shreenivas et. al. (2017).

7.4 Project Limitation

Constraints of this project evaluation are not provided; the results may be slightly different from original source code. In this project also using SVELTE source code as the base of the source code are old version of Contiki which causing few errors

and faults during using the mote source file. This causes limited environments can be executed within the experiment.

7.5 Future Work

This project can be improved its performance in the future by adding new implementation or improving current source code features. Possible implementation or enhancement that can be deemed to include:

i. The implementation of within different environment of experiment such applying Firewall for each experiment to see how the IDS would work against attacks.

ii. Experiment using larger network to test how the power consumption, ETX value and detection rate working once the network user getting higher.

iii. Proving the integrity of this experiment by the detection rate of the IDS as the results may be vary within different network environment.

7.6 Conclusion

In conclusion the comparison of hybrid IDS in this project may help future work on comparing existing computer simulation IDS. This could be an initiative to produce in real life intrusion detection system in the future.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

REFERENCES

Ahmed, A. W., Ahmed, M. M., Khan, O. A. & Shah, M. A. 2017. A comprehensive analysis on the security threats and their countermeasures of IoT. *International Journal of Advanced Computer Science Application,* 8**,** 489-501.

Amin, S. O., jig Yoon, Y., Siddiqui, M. S. & Hong, C. S. A novel intrusion detection framework for IP-based sensor networks. 2009 International Conference on Information Networking, 2009. IEEE, 1-3.

Bhattacharyya, T. R. & Pushpalatha, M. 2018. Routing protocols for internet of things: a survey. *International Journal of Engineering Technology,* 7**,** 196-199.

Bostani, H. & Sheikhan, M. 2017. Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach. *Computer Communications,* 98**,** 52-71.

Bouziani, O., Benaboud, H., Chamkar, A. S. & Lazaar, S. A Comparative study of Open Source IDSs according to their Ability to Detect Attacks. Proceedings of the 2nd International Conference on Networking, Information Systems & Security, 2019. 1-5.

Cervantes, C., Poplade, D., Nogueira, M. & Santos, A. Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things. 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), 2015. IEEE, 606-611.

De Couto, D. S., Aguayo, D., Bicket, J. & Morris, R. A high-throughput path metric for multi-hop wireless routing. Proceedings of the 9th annual international conference on Mobile computing and networking, 2003. 134-146.

Hajiheidari, S., Wakil, K., Badri, M. & Navimipour, N. J. 2019. Intrusion detection systems in the Internet of things: A comprehensive investigation. *Computer Networks,* 160**,** 165-191.

Jow, J., Xiao, Y. & Han, W. 2017. A survey of intrusion detection systems in smart grid. *International Journal of Sensor Networks,* 23**,** 170-186.

Liao, H.-J., Lin, C.-H. R., Lin, Y.-C. & Tung, K.-Y. 2013. Intrusion detection system: A comprehensive review. *Journal of Network Computer Applications,* 36**,** 16-24.

Mardini, W., Aljawarneh, S., Al-Abdi, A. & Taamneh, H. Performance evaluation of RPL objective functions for different sending intervals. 2018 6th international symposium on digital forensic and security (ISDFS), 2018. IEEE, 1-6.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Matsunaga, T., Toyoda, K. & Sasase, I. 2015. Low false alarm attackers detection in RPL by considering timing inconstancy between the rank measurements. *IEICE Communications Express,* 4**,** 44-49.

Midi, D., Rullo, A., Mudgerikar, A. & Bertino, E. Kalis—A system for knowledge-driven adaptable intrusion detection for the Internet of Things. 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), 2017. IEEE, 656-666.

Nandhini, P. & Mehtre, B. Intrusion Detection System Based RPL Attack Detection Techniques and Countermeasures in IoT: A Comparison. 2019 International Conference on Communication and Electronics Systems (ICCES), 2019. IEEE, 666-672.

Napiah, M. N., Idris, M. Y. I. B., Ramli, R. & Ahmedy, I. 2018. Compression header analyzer intrusion detection system (CHA-IDS) for 6LoWPAN communication protocol. *IEEE Access,* 6**,** 16623-16638.

Nygaard, F. 2017. *Intrusion detection system in IoT*. NTNU.

Oh, D., Kim, D. & Ro, W. W. 2014. A malicious pattern detection engine for embedded security systems in the Internet of Things. *Sensors,* 14**,** 24188-24211.

Othman, S. M., Alsohybe, N. T., Ba-Alwi, F. M. & Zahary, A. T. 2018a. Survey on intrusion detection system types. *Int. J. Cyber Secur. Digit. Forensics,* 7**,** 444-463.

Othman, S. M., Alsohybe, N. T., Ba-Alwi, F. M. & Zahary, A. T. 2018b. Survey on intrusion detection system types. *Int. J. Cyber Secur. Digit. Forensics,* 7**,** 444-463.

Perdisci, R., Giacinto, G. & Roli, F. 2006. Alarm clustering for intrusion detection systems in computer networks. *Engineering Applications of Artificial Intelligence,* 19**,** 429-438.

Pongle, P. & Chavan, G. J. I. J. o. C. A. 2015. Real time intrusion and wormhole attack detection in internet of things. 121.

Raza, S., Wallgren, L. & Voigt, T. 2013. SVELTE: Real-time intrusion detection in the Internet of Things. *Ad hoc networks,* 11**,** 2661-2674.

Sedjelmaci, H., Senouci, S. M. & Al-Bahri, M. A lightweight anomaly detection technique for low-resource IoT devices: A game-theoretic methodology. 2016 IEEE international conference on communications (ICC), 2016. IEEE, 1-6.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Shreenivas, D., Raza, S. & Voigt, T. Intrusion detection in the RPL-connected 6LoWPAN networks. Proceedings of the 3rd ACM international workshop on IoT privacy, trust, and security, 2017. 31-38.

Wallgren, L., Raza, S. & Voigt, T. 2013. Routing attacks and countermeasures in the RPL-based internet of things. *International Journal of Distributed Sensor Networks,* 9**,** 794326.

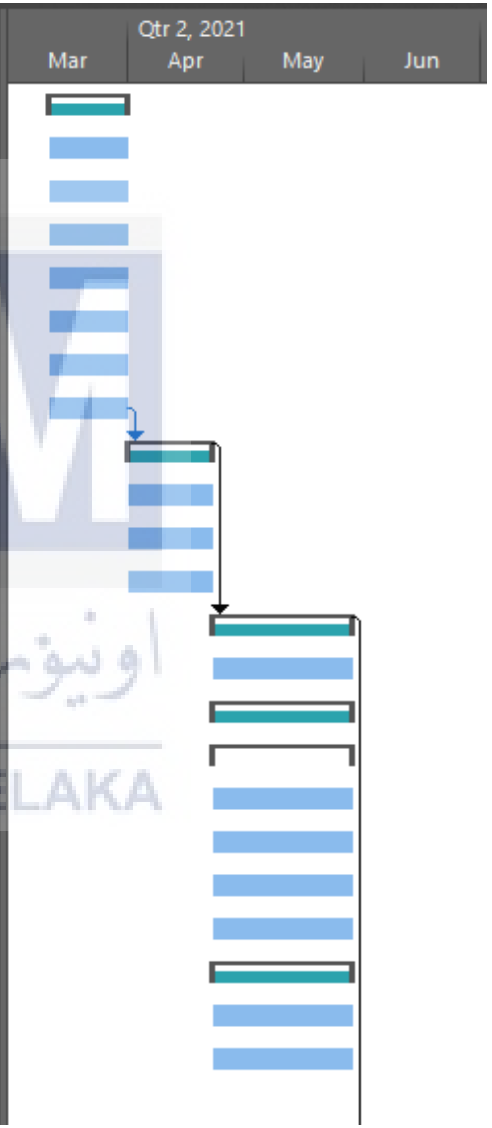
Zarpelão, B. B., Miani, R. S., Kawakani, C. T. & de Alvarenga, S. C. 2017. A survey of intrusion detection in Internet of Things. *Journal of Network Computer Applications,* 84**,** 25-37.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPENDIX

UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاک
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

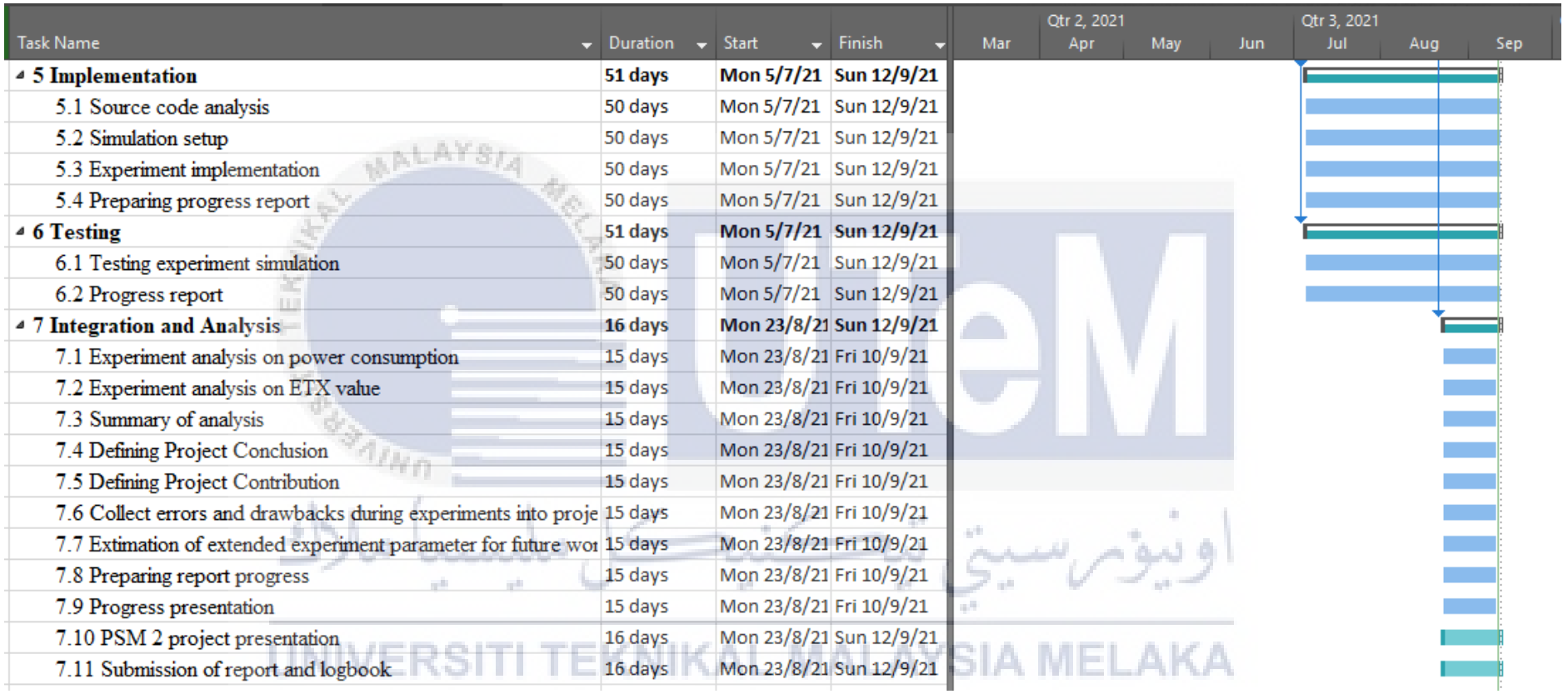
Task Name	Duration	Start	Finish
1 Planning	**14 days**	**Fri 12/3/21**	**Wed 31/3/21**
1.1 Identifying problem statement	14 days	Fri 12/3/21	Wed 31/3/21
1.2 Identifying project question	14 days	Fri 12/3/21	Wed 31/3/21
1.3 Identifying project objectives	14 days	Fri 12/3/21	Wed 31/3/21
1.4 Identifying project scope	14 days	Fri 12/3/21	Wed 31/3/21
1.5 Identifying project contribution/project expected outcome	14 days	Fri 12/3/21	Wed 31/3/21
1.6 Preparing progress report	14 days	Fri 12/3/21	Wed 31/3/21
1.7 Proposal presentation	14 days	Fri 12/3/21	Wed 31/3/21
2 Analysis: Literature Review	**16 days**	**Thu 1/4/21**	**Thu 22/4/21**
2.1 Review previous research	16 days	Thu 1/4/21	Thu 22/4/21
2.2 Review related task	16 days	Thu 1/4/21	Thu 22/4/21
2.3 Preparing progress report	16 days	Thu 1/4/21	Thu 22/4/21
3 Design: Methodology	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.1 SDLC planning	26 days	Fri 23/4/21	Fri 28/5/21
3.2 Phase detailing	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.1 Planning phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.1.1 Constructing Problem Statement	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.1.2 Constructing Project Question	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.1.3 Constructing Project Objective	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.1.4 Constructing Project Scope	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.2 Analysis phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.2.1 Define on the methodologies of past research	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.2.2 Review related work on 6LoWPAN, IDS and RPL network.	26 days?	Fri 23/4/21	Fri 28/5/21



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Task Name	Duration	Start	Finish
3.2.3 Design phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.3.1 Define project approach and how it will be implemented	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.3.2 Identifying project activities	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.4 Implementation phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.4.1 Preparing source code	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.4.2 Define experiment parameter	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.4.3 Define source code path	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.4.4 Perform the simulation environment	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.5 Testing phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.5.1 Testing all the source code for each experiment	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.5.2 Find possible solution for any error.	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.6 Integration and analysis phase	**26 days?**	**Fri 23/4/21**	**Fri 28/5/21**
3.2.6.1 : Collecting data into 4 main graph (small network, large network and both)	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.6.2 Analyse by graph	26 days?	Fri 23/4/21	Fri 28/5/21
3.2.6.3 Stating reasons behind the graph	26 days?	Fri 23/4/21	Fri 28/5/21
3.3 Preparing progress report	26 days	Fri 23/4/21	Fri 28/5/21
3.4 Progress presentation	26 days	Fri 23/4/21	Fri 28/5/21
4 Design: Experiment simulation	**21 days**	**Mon 31/5/21**	**Mon 28/6/21**
4.1 Analysis on Project Problem, Requirement, Dataset	20 days	Mon 31/5/21	Sun 27/6/21
4.2 Design simulation design, data validation	20 days	Mon 31/5/21	Sun 27/6/21
4.3 Preparing progress report	20 days	Mon 31/5/21	Sun 27/6/21
4.4 Progress presentation	20 days	Mon 31/5/21	Sun 27/6/21
4.5 PSM 1 project presentation	20 days	Mon 31/5/21	Sun 27/6/21
4.6 Report and logbook submission	21 days	Mon 31/5/21	Mon 28/6/21

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Task Name	Duration	Start	Finish
5 Implementation	**51 days**	**Mon 5/7/21**	**Sun 12/9/21**
5.1 Source code analysis	50 days	Mon 5/7/21	Sun 12/9/21
5.2 Simulation setup	50 days	Mon 5/7/21	Sun 12/9/21
5.3 Experiment implementation	50 days	Mon 5/7/21	Sun 12/9/21
5.4 Preparing progress report	50 days	Mon 5/7/21	Sun 12/9/21
6 Testing	**51 days**	**Mon 5/7/21**	**Sun 12/9/21**
6.1 Testing experiment simulation	50 days	Mon 5/7/21	Sun 12/9/21
6.2 Progress report	50 days	Mon 5/7/21	Sun 12/9/21
7 Integration and Analysis	**16 days**	**Mon 23/8/21**	**Sun 12/9/21**
7.1 Experiment analysis on power consumption	15 days	Mon 23/8/21	Fri 10/9/21
7.2 Experiment analysis on ETX value	15 days	Mon 23/8/21	Fri 10/9/21
7.3 Summary of analysis	15 days	Mon 23/8/21	Fri 10/9/21
7.4 Defining Project Conclusion	15 days	Mon 23/8/21	Fri 10/9/21
7.5 Defining Project Contribution	15 days	Mon 23/8/21	Fri 10/9/21
7.6 Collect errors and drawbacks during experiments into proje	15 days	Mon 23/8/21	Fri 10/9/21
7.7 Extimation of extended experiment parameter for future wor	15 days	Mon 23/8/21	Fri 10/9/21
7.8 Preparing report progress	15 days	Mon 23/8/21	Fri 10/9/21
7.9 Progress presentation	15 days	Mon 23/8/21	Fri 10/9/21
7.10 PSM 2 project presentation	16 days	Mon 23/8/21	Sun 12/9/21
7.11 Submission of report and logbook	16 days	Mon 23/8/21	Sun 12/9/21

UNIVERSITI TEKNIKAL MALAYSIA MELAKA