

IPV6 TRANSITION MECHANISM DOS ATTACK DETECTOR



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL : IPV6 TRANSITION MECHANISM ATTACK DETECTOR

SESI PENGAJIAN : 2015/2016

Saya SIRAIJ HARRIS BIN UMAR

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis dan projek adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. ** Sila tandakan (/)

_____ SULIT

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

_____ TERHAD

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

_____/_____/_____ TIDAK TERHAD

Sirajit Harris

(TANDATANGAN PENULIS)
Alamat Tetap: No 42, Jalan
Cendana 14, Taman Rinting,
Masai, Johor

Tarikh: 22/08/2016



(TANDATANGAN PENYELIA)

NAZRUL AZHAR BAHAMAN

Nama Penyelia

Tarikh:

22/8/2016

IPV6 TRANSITION MECHANISM DOS ATTACK DETECTOR

SIRAIJ HARRIS BIN UMAR



This report is submitted in partial fulfilment of the requirement for the
Bachelor of Computer Science (Computer Networking)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2016

DECLARATION

I hereby declare that this project report entitled

IPV6 TRANSITION MECHANISM DOS ATTACK DETECTOR

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

STUDENT : *SirajjHarris* Date: 22/8/2016
(SIRAIJ HARRIS BIN UMAR)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
اونيورسيتي تيكنيكل مليسيا ملاك
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 : 
(DR NAZRULAZHAR BAHAMAN)

Date: 22/8/16


ABSTRACT

This study is conducted to discover the solution for the problem IPv6 transition mechanism is exploited as attack medium. The usage of transition mechanism allows Denial of Service attacks to bypass currently deployed security layers. The transition mechanism focused in this study is 6to4 tunnelling where IPv6 packet is encapsulated into IPv4. The solution proposed in this study is to develop an Intrusion Detection System (IDS) that is capable of detecting attacks in transition mechanism. In order to successfully develop the IDS, the first step involve identifying the possible threats in the transition mechanism. Subsequently, the identified threat is used as reference to develop the IDS. A test bed is also designed in this study to test the effectiveness of IDS developed. Besides, published information regarding topics and problems related to this project reviewed and discussed in this study. Previous research related to the topic is studied and used as a reference in this study. The project methodology used in this study is prototyping in which preliminary version of the end product is developed and tested. Changes and improvement is incorporated until the final product is achieved. The study carried out successfully and the outcome of this study is the development of an Intrusion Detection System that is capable of detecting attacks that is channelled into the tunnel of IPv6 transition mechanism.

ABSTRAK

Kajian ini dijalankan untuk mengenalpasti penyelesaian untuk masalah *IPv6 transition mechanism* disalahgunakan sebagai medium untuk melakukan serangan *Denial-of-Service*, DOS. Penggunaan mekanisma transisi membolehkan serangan untuk melepasi sistem keselamatan komputer. Mekanisma transisi yang digunakan di dalam kajian ini ialah terowong 6to4 dimana paket IPv6 berada di dalam paket IPv4. Penyelesaian yang dicadangkan di dalam kajian ini ialah menghasilkan sistem mengesan pencerobohan yang berupaya mengesan serangan di dalam mekanisma transisi. Langkah pertama untuk menghasilkan sistem tersebut ialah mengenalpasti jenis serangan di dalam mekanisma transisi. Seterusnya, serangan yang dikenalpasti digunakan sebagai panduan untuk menghasilkan sistem pengesanan pencerobohan. *Test bed* juga direka di dalam kajian ini untuk menguji keberkesanan sistem yang dihasilkan. Selain itu, maklumat diterbitkan yang berkaitan dibincangkan di dalam kajian ini. Kajian terdahulu yang berkaitan dengan topik kajian dipelajari dan digunakan sebagai panduan di dalam kajian ini. Metodologi projek yang digunakan dalam kajian ini ialah prototaip di mana versi awal produk dihasilkan dan diuji. Penambahbaikan dilakukan pada versi dihasilkan sehingga produk akhir yang menepati objektif dihasilkan. Kajian ini telah dijalankan dengan jayanya dan hasil daripada kajian ini ialah penghasilan sistem pengesanan pencerobohan yang berupaya mengesan serangan yang melalui terowong mekanisma transisi.

TABLE OF CONTENTS

CHAPTER	SUBJECT	PAGE
	DECLARATION	i
	ABSTRACT	ii
	ABSTRAK	iii
	TABLE OF CONTENTS	iv
	LIST OF TABLES	viii
	LIST OF FIGURES	ix
		
CHAPTER I	INTRODUCTION	1
	1.1 Introduction	1
	1.2 Problem Statement	2
	1.3 Project Question	3
	1.4 Objective	4
	1.5 Project Scope	5
	1.6 Project Contribution	5
	1.7 Thesis Organization	6
	1.8 Conclusion	7

CHAPTER II	LITERATURE REVIEW	8
2.1.	Introduction	8
2.2.	Related Work	9
2.2.1.	Internet Protocol	9
2.2.2.	Internet Protocol Version 6	10
2.2.3.	IP in IP	12
2.2.3.1.	Protocol 41	12
2.2.4.	Transition Mechanism	14
2.2.5.	Intrusion Detection System	18
2.3.	Critical Review Of Current Problem And Justification	21
2.3.1.	DOS Attack	21
2.3.2	6to4 Mechanism Threat	23
2.4	Previous Research	26
2.5	Proposed Solution	33
2.6	Conclusion	33
CHAPTER III	PROJECT METHODOLOGY	34
3.1	Introduction	34
3.2	Project Methodology	35
3.2.1	Planning Phase	36
3.2.2	Analysis and Design Phase	36
3.2.3	Implementation phase	38
3.2.4	Testing phase	38
3.3	Project Schedule and Milestones	40
3.4	Conclusion	43
CHAPTER IV	ANALYSIS AND DESIGN	44
4.1	Introduction	44
4.2	Problem Analysis	45
4.3	Requirement Analysis	45
4.3.1	Data Requirement	45
4.3.2	Functional Requirement	46
4.3.3	Other Requirement	47
4.4	High Level Design	49

4.4.1	IPv6 Testbed Architecture	49
4.4.2	Physical and Logical Network Design	50
4.4.3	Logical Design	51
4.5	Software Design	52
4.6	Conclusion	55
CHAPTER V	IMPLEMENTATION	56
5.1	Introduction	56
5.2	Environment Setup	57
5.3	Attack Signature Identification	59
5.4	Implementation Status	63
5.5	Result	64
5.6	Conclusion	67
CHAPTER VI	TESTING	68
6.1	Introduction	68
6.2	Test Plan	69
6.2.1	Test Organization	69
6.2.2	Test Environment	69
6.3	Test strategy	70
6.3.1	Classes of test	70
6.4	Test Design	71
6.4.1	Test Descriptions	71
6.5	Test Results and Analysis	73
6.6	Conclusion	74
CHAPTER VII	CONCLUSION	75
7.1	Introduction	75
7.2	Project Summarization	76
7.3	Project Contribution	76
7.4	Project Limitation	77
7.5	Future Works	77
7.6	Conclusion	78



LIST OF TABLE

TABLE	TITLE	PAGE
1.1	Problem Statement	2
1.2	Summary Of Project Questions	3
1.3	Summary Of Research Objectives	4
1.4	Summary Of Project Contributions	5
2.1	Previous Research Summary	32
3.1	Project Schedule And Milestones	40
6.1	Test Organization	69
6.2	Signature Matching Test	72
6.3	Testbed Connectivity Test Result	73
6.4	Packet Capture And Decoding Test Result	73
6.5	Signature Matching Test Result	74

LIST OF FIGURES

FIGURE	TITLE	PAGE
2.1	Internet Protocol Version 4 Header	9
2.2	Comparison Of Ipv4 And Ipv6 Header	11
2.3	IP In IP Illustration.	12
2.4	Protocol 41 Packet	13
2.5	6to4 Tunnelling Architecture	15
2.6	6to4 Tunnelling Process	17
2.7	Network Based Ids	19
2.8	Host-Based Ids	19
2.9	Denial of Service	21
2.10	Route Advertisement Attack	24
2.11	Testbed Intrusion Detection Mechanism On Transition Mechanism	26
2.12	System Module Composition	29
3.1	Project Flow	35
3.2	Project Gantt Chart	42
4.1	Block Diagram Of Ipv6 Transition Mechanism Attack Detector	46
4.2	IP Transition Mechanism Attack Detector Context Diagram	46
4.3	IP Transition Mechanism Attack Detector Data Flow Diagram	46
4.4	Network System Architecture	49
4.5	Network Physical Design	50
4.6	Logical Network Design	51
4.7	Program Flowchart	52
5.1	Environment Setup	57
5.2	Environment Setup Illustration	57
5.3	Software Environment	58
5.4	Denial6 Test Case 2 Packet Capture	59
5.5	Denial6 Test Case 5 Packet Capture	60

5.6	Denial6 Test Case 7 Packet Capture	60
5.7	Wireshark Snippet Of Sendpees6 Attack	61
5.8	Wireshark Snippet During Thcsyn6 Attack	62
5.9	Output Produced By Attack Detector	64
5.10	Detection Log Produced	65
5.11	Detection Log	66
5.12	System Attack Notification	67
6.1	Black Box Testing	70
6.2	Testbed Connectivity Test	71
6.3	Packet Capture Test	71



CHAPTER I

INTRODUCTION

1.1 Introduction



IPv6 is the improved version of network layer protocol that is designed to overcome the shortfall of IPv4 protocol in meeting the demand of growing number of user in the global internet. IPv6 was developed with the intention of replacing IPv4 protocol. Unfortunately, IPv6 and IPv4 is a separate protocol and it is not backward compatible with the existing IPv4 protocol. IPv6 transition mechanism were introduced to enable IPv6 host to communicate with IPv4 host and network and to allow isolated IPv6 network to reach each other over IPv4 infrastructure. There are vulnerabilities identified in this protocol suite and transition mechanism has been manipulated as a platform to perform threats that exploit those vulnerabilities. Most current security mechanism unable to detect attacks that are hidden in the transition mechanism tunnel payload. Thus it is crucial to develop tool that can detect attack through transition mechanism.

One of the possible countermeasures for this issue is by using improved intrusion detection system that is capable of decapsulating the tunnel header and checks the packet in the payload. Python is the programming language that can be used to develop the intrusion detection system. Scapy, a packet manipulation utility for Python helps to process the packets. The deployment of this improved intrusion detection system enable attacks in the tunnel can be detected and necessary actions can be taken.

1.2 Problem Statement

The problem that has been identified is summarised in Table 1.1 below

Table 1.1 Problem Statement

PS	Problem Statement
PS1	IPv6 Transition Mechanism lacks security mechanism to defend against threats such as Denial-of-Service

PS1: IPv6 Transition Mechanism lacks security mechanism to defend against threats such as Denial-of-Service

Vulnerabilities of IPv6 Transition Mechanism is exploited and it is used as a medium to perform various threats or attacks. Tunnelling mechanisms have no built-in security at all; no authentication, no integrity check, and no confidentiality (Çalışkan, 2014). This make tunnelling mechanism susceptible to be used in attacks such as DOS attack.

1.3 Project Question

Three Project Question (PQ) is constructed based on the problem statement that needs to be answered in this project. The summary of project question is shown in Table 1.2.

Table 1.2 Summary of Project Questions

PS	PQ	Project Question
PS1	PQ1	What are the possible DOS threats in the transition mechanism?
	PQ2	How to detect the presence of the threat?
	PQ3	How to ensure that the threat can be detected effectively?

PQ1: What are the possible threats to the transition mechanism?

Identify the characteristics and the signature of the threats that could possibly occur through the implementation of IPv6 transition mechanism

PQ2: How to detect the presence of the threat?

The method or type of tools to be used to detect the presence of threats or attack

PQ3: How to ensure that the threat can be detected effectively?

How to know whether the solution able to detect the threats during attack effectively with minimal false positive and false negative?

1.4 Objective

Based on the project questions formulated in previous section, appropriate project objectives (PO) are developed as follows: The Project Objective (PO) is summarized into Table 1.3.

Table 1.3 Summary of research objectives

PS	PQ	PO	Project Objective
PS1	PQ1	PO1	To identify what are the possible threats in the transition mechanism.
	PQ2	PO2	To develop a tool that can detect the presence of the threat.
	PQ3	PO3	To test and verify the effectiveness of the tool.

PO1: To identify what are the possible threats in the transition mechanism

Study on how the transition mechanism is exploited by attacker and to identify the signature of the attack.

PO2: To develop tool that can detect the presence of the threat

Produce a tool that will scan all the network activity inside the transition mechanism to detect any presence of transition mechanism attack.

PO3: To test and verify the effectiveness of the tool

Design testing method by simulating the attack to test the effectiveness of the tool in detecting the presence of threats.

1.5 Project Scope

The Scope of this research paper will be focusing on the aspects stated below:

1. Denial of Service attacks that exploit transition mechanism
2. Intrusion Detection System that detects DOS attacks in IPv6 transition mechanism
3. Test bed to test the effectiveness of the Intrusion Detection System

1.6 Project Contribution

The contribution of this project is summarized in Table 1.4:

Table 1.4 Summary of project contributions

PS	PQ	PO	PC	Project Contribution
PS1	PQ1	PO1	PC1	Identification of threats in transition mechanism
	PQ2	PO2	PC2	Propose a tool that is capable to detect attacks on transition mechanism
	PQ3	PO3	PC3	Propose a test bed to test the effectiveness of Intrusion Detection System on transition mechanisms

1.7 Thesis Organization

This report consists of six chapter namely Chapter 1: Background, Chapter 2: Literature Review, Chapter 3: Methodology, Chapter 4: Design and Implementation, Chapter 5: Testing and Result Analysis and Chapter 6: Conclusion.

Chapter 1: Introduction

This chapter will discuss about introduction, project background, research problem, research question, research objective, scope, project significant and report organization.

Chapter 2: Literature Review

This chapter will explain related work of this recommendation system, such as IPv6 standards, type of attacks and transition mechanisms.

Chapter 3: Methodology

This chapter will explain the methodology used to carry out the project and description of activities carried out during each phase of the methodology.

Chapter 4: Analysis and Design

This chapter discusses on the analysis on the problem and requirement. Besides this chapter covers the high level design, user interface design and the system architecture.

Chapter 5: Implementation

This chapter cover the activity involved in the implementation phase, the software development environment setup, software configuration management and the implementation status.

Chapter 6: Testing

This chapter discusses on the activity involved in the testing phase, the test plan that includes test environment, test schedule and test strategy and also the test result and analysis.

Chapter 7: Conclusion

This chapter summarize the project and discusses on how the objective has been achieved, the strength and weakness of the project and what are the contributions of this project

1.8 Conclusion


In this chapter, problem statement, objective, scope, project significant and expected output of the projects are clearly identified. The next chapter, Chapter 2 will discuss the related work of this project.



CHAPTER II

LITERATURE REVIEW

2.1. Introduction



The exponential growth of number of computer and other smart devices causes depletion of IPv4 address. IPv6 were introduced to overcome the address shortfall of IPv4 with huge address space and better efficiency. However IPv6 is not interoperable directly with IPv4. Thus transition mechanism is used to smooth out the transition to the latest internet protocol. The implementation of transition mechanism raised security concerns as it allow cybercriminal to launch attacks namely Denial-of-Service attack. Many researches has been done to discover the defence against attacks on transition mechanism. Intrusion detection system is a form of countermeasure identified as a defence layer against the attacks.

In this chapter, published information regarding topics related in this project is reviewed and discussed. Besides, the problems related to this research is studied and analysed. Previous research in the area of this topic is studied and the possible solution to the problem is proposed.

2.2. Related Work

This section explains in detail the subjects or knowledge area related to this project which includes the network layer protocols, transition mechanism and intrusion detection system.

2.2.1. Internet Protocol

Internet protocol is the protocol that governs how packets or blocks of data are transmitted from source to destination host through the use of fixed length addresses known as the IP address. The internet protocol also provides for fragmentation and reassembly of long datagrams, if necessary, for transmission through "small packet" networks (RFC791, 1981). The basic function of Internet Protocol is addressing and fragmentation.

Bits							
0	4	8	16	19	31		
Version		Length		Type of Service		Total Length	
Identification				Flags	Fragment Offset		
Time to Live		Protocol		Header Checksum			
Source Address							
Destination Address							
Options							
Data							

Figure 2.1 Internet Protocol version 4 header

Figure 2.1 above shows the header structure of IPv4 packet. The information needed for the Internet Protocol to perform its function is stored in the packet header. Addressing is used by Internet Protocol to identify host on the network and for the selection of path for transmission known as routing. The address are made up of four octets (32 bits). The source and destination address field will be used by Internet Protocol to determine which path should the packet be routed. The process of selecting the path is usually carried out by gateway and intermediary devices such as a router.

Besides, the packet header contain fields for fragmentation that holds information of identification, flags and the fragment offset that will be used for fragment reassembly. The version field indicates the version of Internet Protocol of the packet. The length of internet header is determined through the Internet Header Length (IHL) field shown in 32 bits words. The type of service contains information for quality of service. Total length of the datagram in octets is stored in Total Length field. Time to live field placed in the header indicates the maximum hop the packet travels before it is being discarded to prevent it from looping around the network. The type of protocol in the data portion is shown in the protocol field. The error checking mechanism of the IP packet use checksum which are calculated upon arrival to detect the presence of errors in the packet.

2.2.2. Internet Protocol Version 6

Internet Protocol Version 6 is the improved version of network layer protocol and it is developed to address the shortfall of IPv4. IPv4 address are made up of 32 bits address with very limited address space and it is facing exhaustion. IPv6 expand the addressing space by increasing the address bits to 128 bits which supports up to 340 undecillion addresses or $3.4 * 10^{38}$ addresses. IPv6 also increase the addressing capabilities by supporting increased levels of addressing hierarchy, stateless address auto-configuration and introduction of a new type of address known as anycast address that is used to send a packet to group of nodes (RFC2460, 1998). Besides, IPv6 simplifies the header of the packet by removing or made optional processing overhead

fields in the header. Furthermore, IPv6 improves the security by authentication and privacy capabilities.

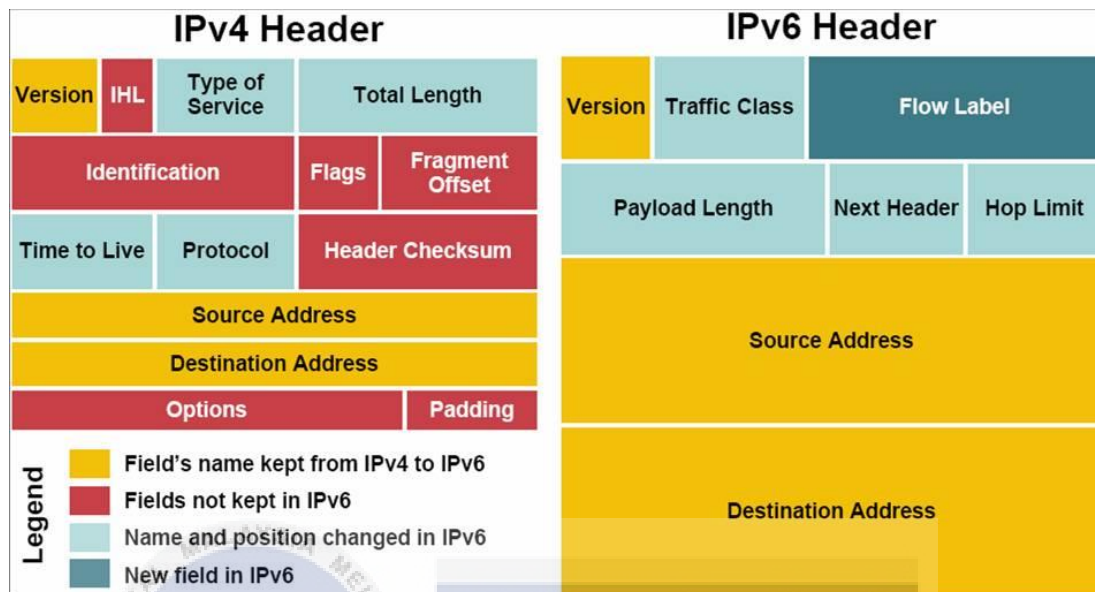


Figure 2.2 Comparison of IPv4 and IPv6 header

Figure 2.2 above shows the difference in header field between IPv4 and IPv6. IPv6 packet header is more streamlined by removing fields that requires processing such as the fragment fields and checksum.

The version field that identifies the Internet Protocol version used and the source and destination address kept from IPv4 to IPv6. The 8 bit Traffic Class field is used by host or intermediary devices to differentiate between different priorities of IPv6 packets. The sequence of the packet is labelled using the 20 bit Flow Label field. The 8 bit Next Header field identifies the type of header immediately following the IPv6 header (RFC2460, 1998). Hop Limit is similar to IP version 4 Time to Live field which limit the number of maximum number of hops the packet allowed to travel. Besides, the 16 bits payload length indicates how long in octets the payload and extension header of the packet.

2.2.3. IP in IP

An IP packet can be in the payload or the data portion of another IP packet. This is known as IP in IP and it is a form of tunnelling. IP in IP packet gets encapsulated at a node called encapsulator and then decapsulated at the decapsulator before arriving at the destination. Figure 2.3 below shows the structure of IP in IP (Perkins, 1996). There are 2 IP header in IP in IP packet. One for the outer packet and one for the inner packet. The outer IP header contains source and destination address of the tunnel endpoints. The inner IP header contains addresses that determines the actual sender and receiver.

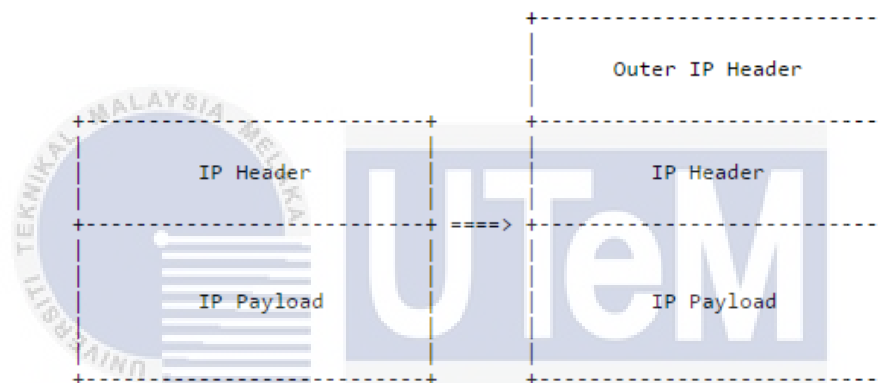


Figure 2.3 IP in IP illustration.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.2.3.1. Protocol 41

There is an 8-bit header field in IPv4 packet that identify the next level protocol (RFC791, 1981). The equivalent header field in IPv6 packet is called Next Header (RFC2460, 1998). The protocol number known as Assigned Internet Protocol Numbers is handled by Internet Assigned Numbers Authority (IANA). Protocol number 41 assigned by IANA denotes that the payload of the packet is IPv6 encapsulation (Internet Assigned Number Authority, 2016). IPv6 encapsulation is a mechanism in which a packet is encapsulated and transmitted as a payload within another packet (RFC2473 , 1998).

This is known as tunnelling. Source node at the tunnel entry-point encapsulate a packet and channel it through the tunnel and the opposite end of the tunnel known as the tunnel exit-point, the packet is de-capsulated.

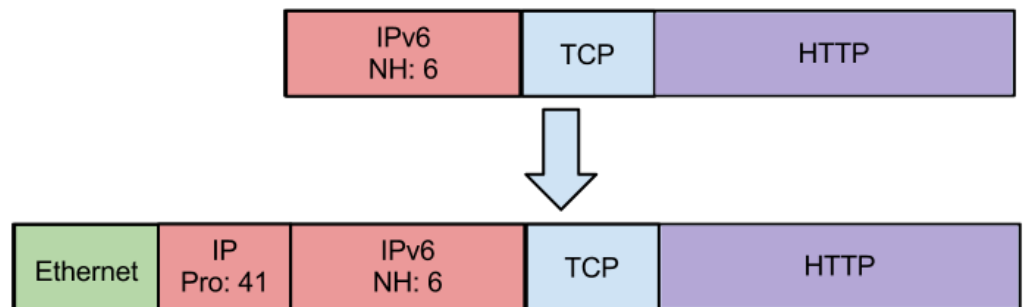
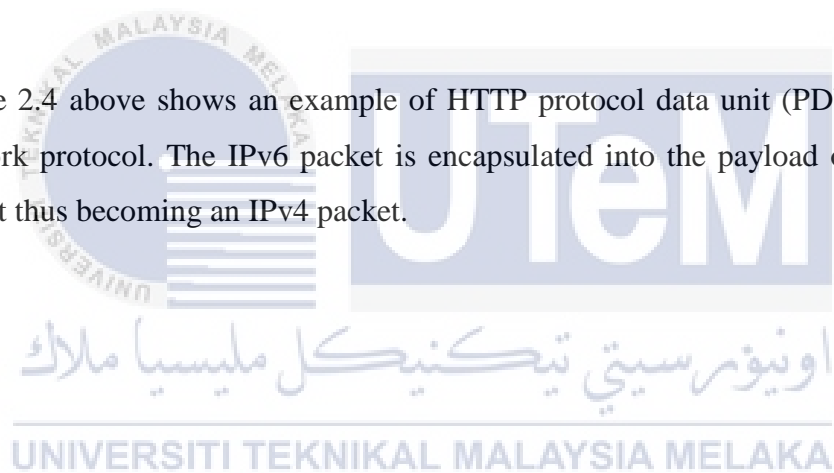


Figure 2.4 Protocol 41 Packet

Figure 2.4 above shows an example of HTTP protocol data unit (PDU) using IPv6 network protocol. The IPv6 packet is encapsulated into the payload of protocol 41 packet thus becoming an IPv4 packet.



2.2.4. Transition Mechanism

IPv6 adoption rate in the network is still low despite having huge address space and has many improvements such as streamlined protocol header over the widely used IPv4 protocol (Hagen, 2014). The factor hindering the deployment of IPv6 is its incompatibility with IPv4. IPv6 would not be able to communicate directly with IPv4. This would be a major issue as most host and router in the network around the globe still uses IPv4 (Lawton, 2001). IPv6 transition mechanism is introduced to overcome the issue and to facilitate the transition to IPv6. IPv6 designed to allow IPv6 nodes to maintain complete compatibility with IPv4, which should greatly simplify the deployment of IPv6 in the Internet, and facilitate the eventual transition of the entire Internet to IPv6. (RFC2893, 2000). IPv6 transition mechanism include dual stack in which both IPv4 and IPv6 are implemented in the host as well as at the router

2.2.4.1. Tunnelling

Tunnelling is also employed as a transition mechanism. Tunnelling is method of encapsulating packet in another packet for transmission. Tunnelling can be either configured or established automatically (RFC2893, 2000). Configured tunnelling is setup using configuration information at the encapsulating node. On the contrary, automatic tunnelling mechanism utilize a special address that will be used to determine the tunnel endpoint thus only little configuration is needed.

6to4 tunnels is commonly used when the ISP yet to support IPv6 Internet. The common architecture of 6to4 tunnels includes the connection between multiple border router and from a border router to relay router to IPv6 Internet (Hei & Yamazaki, 2004). The border router and relay router has pseudo-interface or tunnel endpoint that is logically equivalent to IPv6 interface. 6to4 transition mechanism can be implemented to interconnect multiple isolated IPv6 island. The traffic from one of the island transmitted over IPv4 network through a tunnel to another island. Besides, 6to4 tunnel interconnects the IPv6 island over IPv4 internet to a 6to4 relay router which acts as a transit point or gateway to the native IPv6 network. 6to4 router locate 6to4 relay router by using its 6to4 relay anycast address of 192.168.99.1 or IPv6 relay anycast address of 2002:c058:6301:: (RFC3068, 2001). Traffic from native IPv6 network to 6to4 network with prefix 2002::/16 will be forwarded to the nearest advertising 6to4 relay that will tunnel them to the corresponding IPv4 address. Besides, the relay router may advertise route to native IPv6 network using exterior routing protocol such as BGP4+ to the 6to4 router (RFC3056, 2001).

The IPv6 prefix allocated by IANA for 6to4 is 2002::/16. This address will then be appended with 32 bits of IPv4 address to produce valid IPv6 /48 prefix (RFC3056, 2001). This prefix will be used as interim unique IPv6 address for transmitting IPv6 packet over global IPv4 network. The IPv6 address will be assigned to the tunnel with the tunnel source is the global IPv4 address of the pseudo-interface. This tunnel interface will be configured and used as exit interface for IPv6 packets.

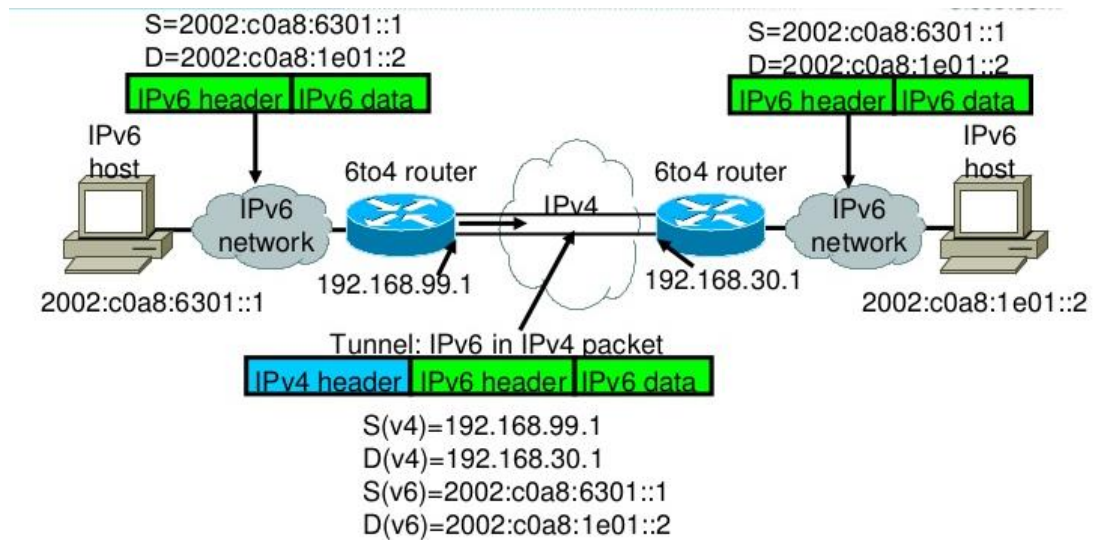
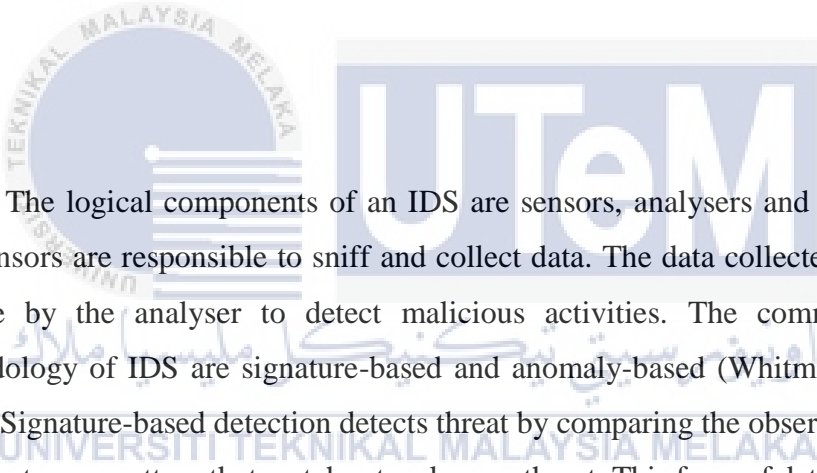


Figure 2.6 6to4 Tunneling Process

Figure 2.6 above illustrates the packets header and data during 6to4 tunneling process. As the 6to4 router receive packet destined to remote IPv6 network, the IPv6 packet will be encapsulated into IPv4 packet with protocol type 41 (RFC2893, 2000) that indicate the payload is IPv6. The source IP address is the tunnel source address which is the global IPv4 address of the 6to4 router. If the destination address or next hop address is made up of the 2002::/48 prefix, the 32 bits IPv4 address will be extracted from the address and used as the destination IP address for the IPv4 header. The packet will then be routed normally over IPv4 network to the destination. Once arrived at the destination 6to4 router, the IPv4 packet is de-capsulated and the packet is forwarded to its destination.

2.2.5. Intrusion Detection System

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices (Scarfone & Mell, 2007). Intrusion Detection System (IDS) is a system that monitors traffic for policy violation or malicious activities, log the detection and notify the security administrator. Intrusion detection system is implemented in the network to detect the presence of intruders especially those that manage or trying to bypass the security defence layer such as a firewall, anti-virus and access control so that preventive measures can be taken.



The logical components of an IDS are sensors, analysers and user interface. The sensors are responsible to sniff and collect data. The data collected will then be analysed by the analyser to detect malicious activities. The common detection methodology of IDS are signature-based and anomaly-based (Whitman & Mattord, 2008). Signature-based detection detects threat by comparing the observed event with the signature or pattern that matches to a known threat. This form of detection requires regular updates to its signature database to ensure it is up to date with various forms of new attacking techniques. On the other hand, anomaly-based detection IDS compares observed event with a baseline or activity that is considered normal to identify significant deviations (Scarfone & Mell, 2007). Anomaly based IDS requires training period before full implementation in which it tries to create a profile for normal behaviours. The created profile will then be used to detect anomalies mostly using artificial intelligence techniques such as neural network. Anomaly based IDS has higher false positive rate compared to signature based (Li, et al., 2005).

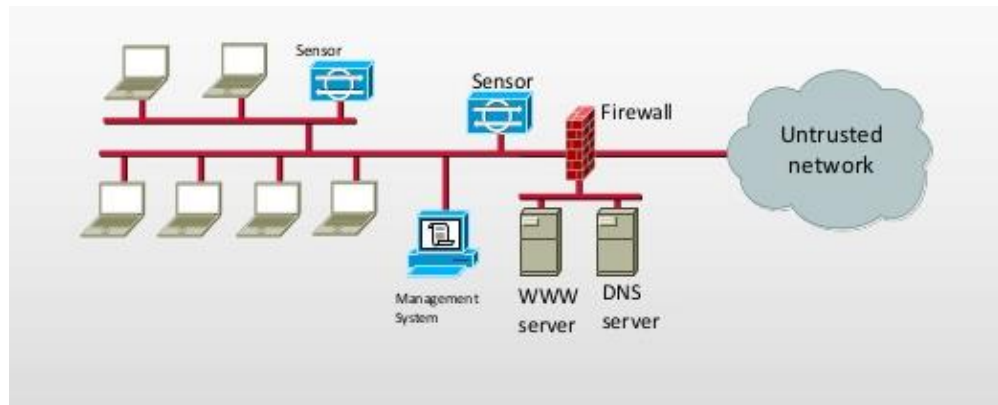


Figure 2.7 Network Based IDS

There are basically two types of IDS; Network-Based and Host-Based. Network Based IDS monitors traffic for suspicious activity for a network segment. Figure 2.7 above is an illustration of network based IDS where the sensor of the IDS is placed in the network. It monitors traffic at selected points in the network. The network based IDS uses an interface in promiscuous mode to sniff all the traffic. The interface is connected to the monitored network segment. The traffic that matches the signature or significant deviation from normal profile will trigger alert to the management console. Network based IDS is easier to deploy compared to host based IDS.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

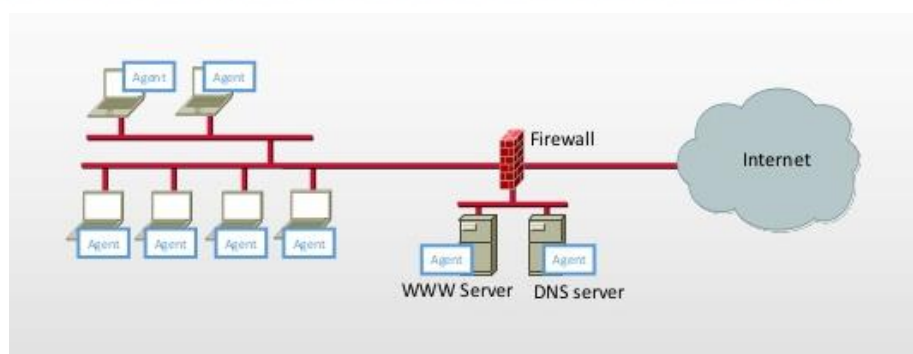
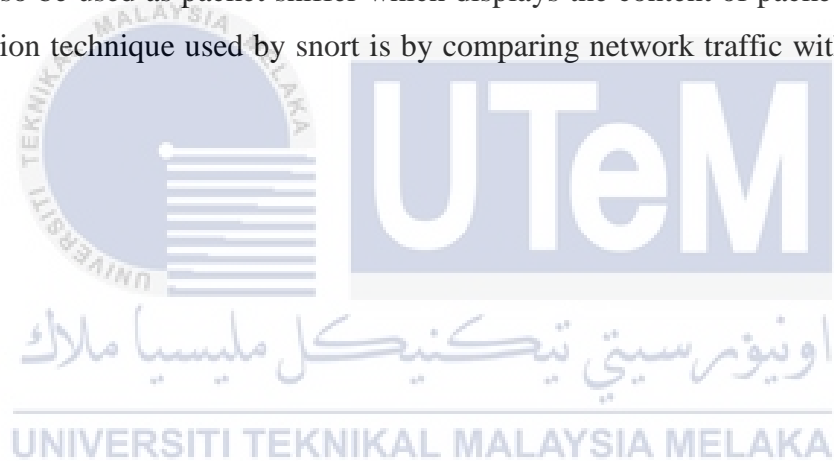


Figure 2.8 Host-based IDS

On the other hand, host-based monitors the characteristics of a single host for suspicious activity. Figure 2.8 shows how host-based IDS is deployed. The host is monitored by IDS agent installed in the host and it monitors the system audit and event logs. Host-based IDS monitors system activities unlike network based IDS that monitors packet. However, host-based IDS capable of monitoring port based activity. Any system changes that matches with attack signature or anomalies will be notified to the management console.

Snort is one of the best free and open source intrusion detection system engine. It is a form of network based intrusion detection system (NIDS) (Roesch, 1999). Snort can also be used as packet sniffer which displays the content of packets sniffed. The detection technique used by snort is by comparing network traffic with user defined rules.



2.3. Critical Review Of Current Problem And Justification

2.3.1. DOS Attack

DOS attack stands for Denial-of-Service attack. A Denial of Service is characterized by an explicit attempt by an attacker to prevent legitimate users from using resources (Lau, Rubin, Smith, & Trajkovic, 2000). An attacker that is performing DOS attack will ‘flood’ the victim or target with false request thus reducing the target bandwidth and available system resource, prevent access to a service or disrupt service to legitimate system or user.

DOS attack is also possible done from multiple source often in thousands simultaneously. This form of DOS attack is known as Distributed Denial of Service, DDoS attack. Attacker performing DDoS attack controls zombie or infected computers remotely and commands them to send false request to the target system, service or host. Figure 2.9 below shows the difference between Denial of Service and distributed Denial of Service.

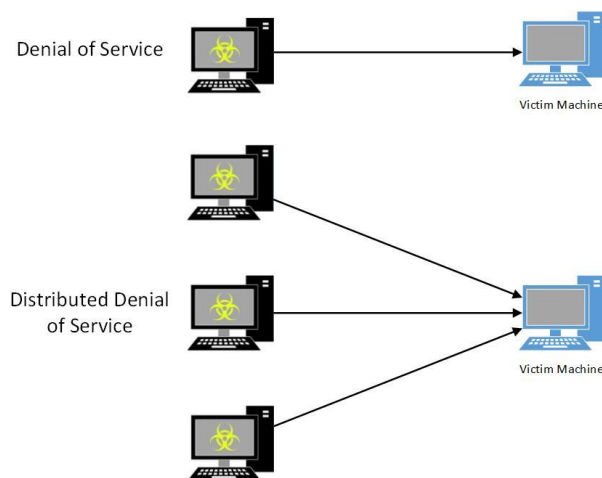


Figure 2.9 Denial of Service

ICMP flooding is one of the simplest method used by attacker to perform Denial of Service attack. This form of attack uses ICMP request and response packets that is used to test connectivity between hosts. The attacker send huge amount of request packet to the victim causing the victim to utilize all of its resources to respond to the request. Since computer nowadays has huge computing resources, this form of attack is less successful. Smurf attack is an amplified version of ICMP flooding and it is a form of DDoS attack (Kumar, 2007).

During Smurf attack, the attacker send ping packet to broadcast IP addresses with the source IP address is spoofed with the victim IP address. Each single host in the broadcast domain responds to request by sending ICMP response packet to the victim causing resource exhaustion of the victim computer or even the entire network. Besides, DOS attack is also carried out using SYN flood. The attacker sends large amount SYN request to the target causing the target to send SYN/ACK packet and wait for the final ACK to complete the three-way handshake process of TCP. The target will wait for the ACK until unable to accept other legitimate incoming connection (Lau, et al., 2000).

Intrusion detection systems (IDS) is one of the countermeasure of Denial of Service attack. It is capable to detect the presence of DOS attack through the signature of the attack and notify the administrator for further action. Besides, it is crucial for host computer and server is up to date with the latest security patches to guard against DOS attack. Moreover, IP broadcast used by flood and Smurf attack should be disabled so that it can't be used as amplifier. The router can be effective in defeating DOS attack by configuring security features such as Ingress and Egress filtering rules and TCP Intercept and Committed Access Rate (CAR) (Piskozub, 2002).

2.3.2 6to4 Mechanism Threat

6to4 mechanism provide automatic IPv6 to IPv4 tunnelling in 6to4 router and 6to4 relay router to interconnect IPv6 networks over IPv4. This characteristic enable a number of security threats, mainly Denial of Service (DOS) (RFC3964, 2004). In 6to4 mechanism, all 6to4 routers must accept IPv4 packet from other 6to4 router as well as 6to4 relays. Besides, 6to4 relays must accept traffic from native IPv6 host. This characteristic of 6to4 mechanism raises security concerns. Common attacks involving 6to4 mechanism are Attacks with Neighbour Discovery (ND) Message, spoofing traffic to 6to4 nodes, reflecting traffic from 6to4 nodes and local IPv4 broadcast attack.

Neighbour Discovery (ND) is a process in IPv6 to determine relationship with neighbouring node. ND is used to resolve link-layer address of neighbouring node similar to ARP and to determine whether the neighbour is reachable or has changed (RFC4861, 2007). Besides, it is used to advertise the router's presence, configuration parameters as well as routes and better next hop address for a specific destination. In Attack with Neighbour Discovery (ND) message, an attacker send specifically crafted Route Advertisement or Neighbour Advertisement message through 6to4 tunnel to the 6to4 pseudo-interface with forged source address to the target to cause partial or complete failure of operation of host on IPv6 link (RFC6104, 2011). This is because IPv6 relies on the message for operation such as determination of prefix for stateless auto configuration.

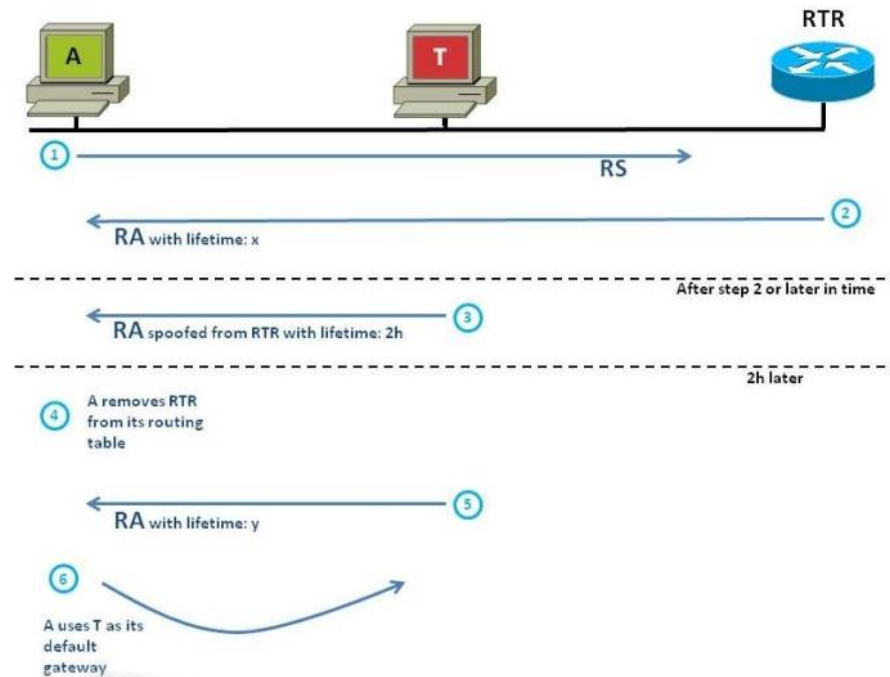


Figure 2.10 Route Advertisement Attack

Figure 2.10 above shows an example of an attack that exploits the route advertisement mechanism. In the attack, attacker use the Route solicitation message to craft a spoofed Route advertisement message to the victim. Victim updates its routing table by including the attacker as the gateway. The attacker than perform man in the middle attack or Denial of Service attack.

Spooing and reflecting traffic to 6to4 nodes attack done by attacker to cover his track as it is difficult to trace packet going through 6to4 node. Attacker uses the 6to4 mechanism to spoof or conceal the IPv4 source address as the IPv4 header is discarded at the end of the tunnel and uses a forged IPv6 source address. Attacker takes advantage of this to accomplish DoS attack. Besides that, the target will reply the traffic to the forged source IPv6 address resulting in the node participating in reflection DoS. The reflection technique can be used by attacker to perform distributed DoS attack by involving large number of nodes in the attack.

Local IPv4 broadcast attack happens when attacker sends encapsulated IPv6 packet with prefix 2002::/16 corresponding with destination address the broadcast address of the 6to4 router. The router will then encapsulate the packet with type 41 and IPv4 broadcast address as the destination. This malicious packet will be received by all the node in the broadcast domain and all responses are forwarded to the 6to4 router and cause problem to the router. This attack is only possible when the router does not check the destination address for invalid address such as broadcast address. Besides, the attacker can also send a packet with invalid IPv6 destination address and source address the 6to4 router broadcast address. This will result in the router sending ICMPv6 error message to the broadcast address producing broadcast storm. This is also a form of DoS attack.

There are tools widely available on the internet that can be used by anyone to initiate attack on IPv6. The most popular and comprehensive toolkit is The Hacker Choice (THC) toolkit. The tool capable of attacking protocol weakness of IPv6 and ICMPv6. The rsmurf6 tool from the THC toolkit accomplish smurf attack by sending spoofed echo-request to multicast address to perform amplification attack (Naidu & Patcha, 2013).

2.4 Previous Research

Research Title: Implementation of IPv6 Network Testbed: Intrusion Detection System on Transition Mechanism

There are a couple of researches done on intrusion detection system on IPv6. One of the research designed a testbed which can be used for experiments to learn the activity of intruder and attacker on transition mechanism (Bahaman, et al., 2011). The testbed includes native IPv6 network, IPv4 network as well as IPv4/IPv6. IPv6 over IPv4 tunnel is used to interconnect IPv6 network to IPv6/IPv4 network over a IPv4 network.

IDS and packet analyser is placed at the 6to4 tunnel. The traffic from attacker to the victim were observed. The testbed architecture is shown in the Figure 2.11 below. ICMPv6 flood attack is used as sample attack in the experiment. The data taken from the experiment is the connectivity, hop count, round trip time, throughput, threat and intrusion detection and packet flow. The researchers found out that the testbed is effective and the hardware and software used is capable of performing its intended function.

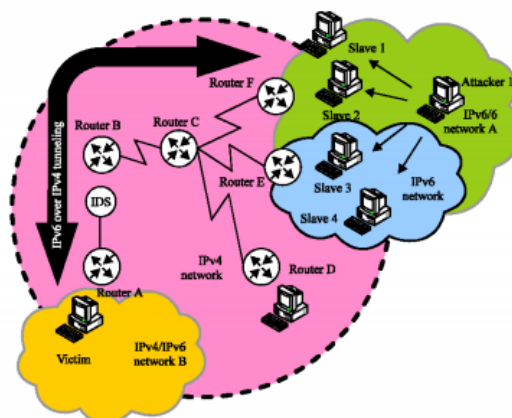


Figure 2.11 Testbed for intrusion detection mechanism on transition mechanism.

Research Title: Ipv6 Security Threats and Possible Solutions

A study has been conducted on IPv6 security threats and the possible solutions. According to the research, despite the introduction of IPv6 security mechanism, their evasion and misuse is still possible. Besides, transition mechanism provide new, previously unknown possibilities of intrusion and misuse of computer system (Zagar & Grgic, 2006).

The security threats to IPv6 are reconnaissance attack where the attacker perform scanning and data mining usually using the IPv6 multicast address to gain information about the target network for further attacks. Besides, routing header can be misused to bypass access control. Access control can also be bypassed by using fragmentation as security mechanism does not reassemble.

Besides ICMPv6 and multicast is misused to cause multiple response targeted to the victim to perform Denial-of-Service attack. ICMPv6 message such as Router Advertisement and Neighbour Discovery can be exploited to perform attack. In transition mechanism, IPv4 or IPv6 address can be spoofed to perform Denial-of-Service attack to target IPv6 node, IPv4 node or other 6to4 node (Zagar & Grgic, 2006). Moreover, the tunnelling facilitates attacker to avoid filtering checks.

One of the solution to the threats is by implementing firewall as it is the most essential defence mechanism. Firewall must be able to support IPv6 because of the different structure of IPv6 packet. Firewall filters traffic based on the separately predefined rules for IPv6 and IPv4. The filtering rules must be separated between IPv4 and IPv6 because the difference in the network layer mechanism. For example, ICMPv6 cannot be filtered by the firewall because it is crucial for proper IPv6 functioning.

It is recommended by this research to implement Host-based IDS on every host and Network based ids on each network segment. Intrusion Detection System implemented must support IPv6 due to the new header format. IDS is recommended to check the extension header and drop undefined “Next Header” and irregular or duplicate options such as hop-by-hop and destination. Furthermore, IDS should also be able to recognize IPv6 tunnelled in IPv4. IPv6 tunnel end point should be before the firewall and IDS is placed at network entry point behind the firewall.

Testbed designed in the study using 5 dual boot computers with Linux and Windows as well as supporting dual-stack with IPv6 connected in a LAN. The LAN is connected to Croation Academic Research Network version 6 (CAR6Net). NMap application has been used to perform TCP connect scan, SYN scan, Xmas tree scan (FIN,URG,PUSH), ICMP scan and UDP scan. All the scan is used to identify opened ports on the target machine. All the scan failed to bypass Linux firewall but some managed to pass through Windows firewall. IDS placed in front of firewall can detect intrusion attempt and behind firewall can detect intrusion that manage to bypass.

Since IPv6 supporting IDS was not present at the time of research, Ethereal application is used to detect intrusion and it successfully capture reconnaissance attack in the experiment as different port is attempted during short period of time. The finding of this study is that it is recommended to implement packet filtering and intrusion detection to safeguard against IPv6 threat. Besides it is suggested to filter all unnecessary services, discard fragment less than 1280 except the last fragment, selectively filter ICMPv6 and use dual-stack or static tunnel instead of dynamic tunnel.

Research Title: Study on Intrusion IPv6 Detection System on Linux

Intrusion detection system (IDS) can complement firewall as firewall only capable of protecting network from outside threat and unable to prevent internal intrusion. Common Intrusion Detection Framework is one of the model for IDS. This model consists of event generators, event analysers, response unit and event database.

The study made by the researcher focus on methods of using Linux platform to detect intrusions in coexisting IPv4 and IPv6 network environment. The method in designing IPv6 intrusion detection centres on four aspect; capturing and understanding IPv6 packet and research and implementation on packet pre-processing, extraction of intrusion features, fast detection and matching of intrusion attack (Yu, 2009). Since IPv6 has new structure of packet with different header fields and extension header, the IDS has to be capable to understand the whole packet. The composition of researched system modules is shown in the Figure 2.12 below. The intrusion detection engine is based on rule set. The controls to the whole function is done using web-based visual console.

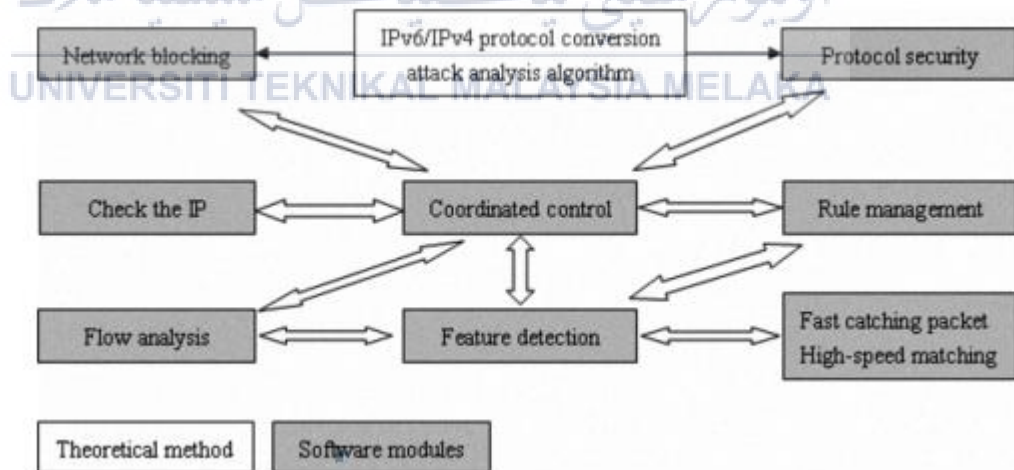


Figure 2.12 System module composition

The placement of Intrusion detection system in a high speed network may cause the IDS to be saturated with traffic and causes packet to be dropped. The method suggested in the research is using multiple node processing system. The functions of the IDS are split where a node capture packets and another node processes the packets. Information system security assurance is achieved by protection, detection, reaction and recovery. The implementation of IDS can provide detection so that reaction and recovery can be done to secure the information system.



Research Title: Threats Posed by Multicast Packets, Extension Headers and Their Counter Measures

The specification of IPv6 has vulnerabilities such as the usage of multicast address and extension headers (Naidu & Patcha, 2013). Multicast address is used for important function of IPv6 such as Router Advertisement for stateless auto configuration and Neighbour Discovery message for MAC address resolution. The simplification of header in IPv6 introduces extension header to place optional IP layer options.

The first phase in performing information system attack is usually the reconnaissance phase. The attacker gathers information about the target during this phase. The traditional method of scanning using ping sweep is not practical as the number of address in IPv6 network is large. Sending ping to multicast address ff02:: method is used instead as it will be received by all the local node. This can be done by using *smurf6* from THC-IPv6 toolkit. Besides reconnaissance, *smurf6* can be used to amplify attack because all the node will respond to the request. The amplification attack can be done to all node or use all node to attack a specific node. Multicast address can also be used to perform Router Advertisement spoofing.

Extensions header in IPv6 can be used by attacker to bypass firewall by chaining lots of extension header causing the firewall to miss upper layer header. Besides, the chain of extension header can cause Denial of Service as the victim too busy processing the header. Hop-by-hop Options is an example of extension header that has information that need to be check by all the intermediary node. Large number of router alert options that instruct router to intercept datagram in hop by hop options header can be used to flood the router to perform dos attack. This attack can be done using *denial6* from THC-IPv6 toolkit.

Table 2.1 Previous Research Summary

Author	Domain	Title	Aim	Outcome
Nazrulazhar Bahaman, Anton Satria Prabuwono, Mohd Zaki Mas'us	IPv6 Transition Mechanism	Implementation of IPv6 Network Testbed: Intrusion Detection System on Transition Mechanism	Implement suitable testbed for future use to do some experiment in order to reveal activities done by intruder	Successful and functional design of network testbed
Drago Zagar	IPv6 Threats	IPv6 Security Threats And Possible Solutions	Analyse how actual threats and different types of attack affects IPv6 network	Solution to IPv6 threats
Zhang Yu	IPv6 IDS	Study on Intrusion IPv6 Detection System on LINUX	Intrusion detection system under IPv6 environment	Design of IPv6 IDS
Santosh Naidu, Amulya Patcha	IPv6 Threats	IPv6: Threats Posed By Multicast Packets, Extension Headers and Their Counter Measures	Addresses security concerns like extensive use of multicast packet and extension header	Security threats and countermeasures

2.5 Proposed Solution

Based on the observation on previous research. The proposed solution for security threats on IPv6 transition mechanism is to implement intrusion detection mechanism. The primary aim of the IDS implementation is to detect Denial-of-Service attack on IPv6 transition mechanism. The testbed designed in (Bahaman, et al., 2011) will be used as a reference to test the effectiveness of the IDS implemented.

2.6 Conclusion

IPv6 is the future of IP network that has huge address space and much more efficient than IPv4. While waiting for the rest of the world to fully adopt the latest protocol, transition mechanism is needed to maintain interoperability with the widely used IPv4 protocol. However, the usage of transition mechanism might expose the network from security threats. Thus it is crucial to take necessary action to secure the network while the transition from IPv4 to IPv6 is in progress.

CHAPTER III

PROJECT METHODOLOGY

3.1 Introduction

A methodology is a documented series of systematic methods for dealing with a complex job or task such as solving problems or developing a system (Dewitz, 1996). Selecting a methodology that is suitable to the project is important because it can ensure the optimum cost and time in carrying out the project. This chapter covers the selected methodology that will be used in this project. The project will be carried out in phases and the activities in each phases will be explained in this chapter. The milestone of each phases is listed out in the form of Gantt chart.

3.2 Project Methodology

Prototyping model has been chosen as a methodology for the project to ensure the objective of the project can be fulfilled. Prototype model dictates that a preliminary version of the end product is developed and tested. Changes and improvements are incorporated to produce a better version iteratively until requirement fulfilling version is achieved. This methodology is suitable to this project because it allows a prototype to be developed with known requirement and improvements are made to the prototype when there are changes to the requirement.

The five main phases in this project are planning, analysis, design, implementation and testing. The project flow is illustrated in the block diagram below:

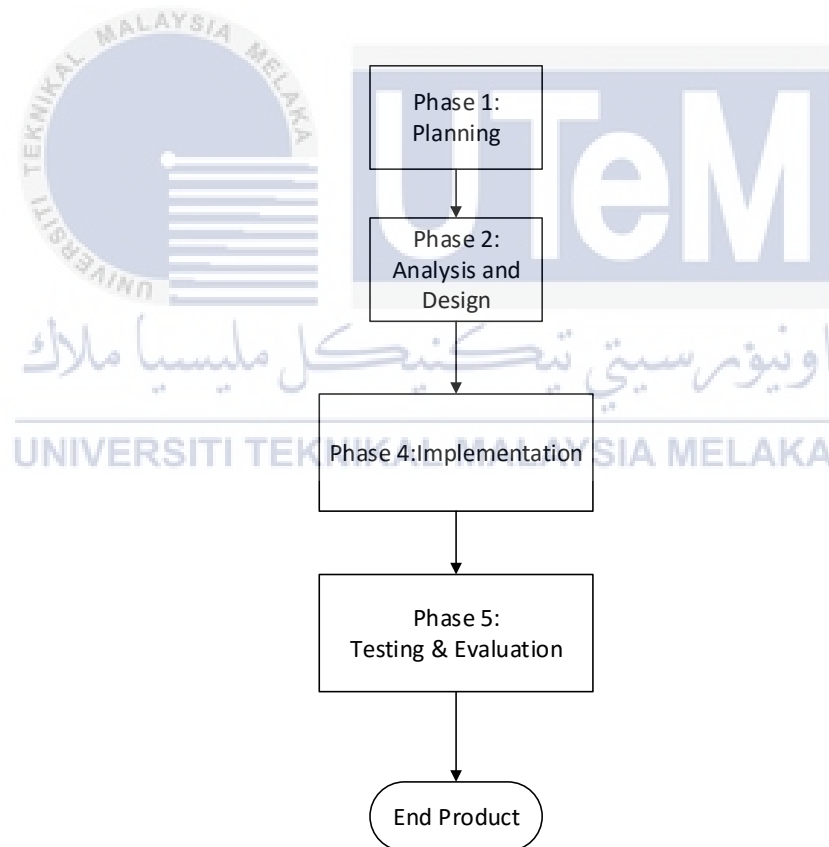


Figure 3.1 Project Flow

3.2.1 Planning Phase

This is the first phase of this project. Plans are created on how this project will be carried out till completion. The activity carried out in this phase are as below:

- Proposal

The problem that lead to the commencement of this project is identified. The problem statement, objective and scope of this project is derived from the problem. The project scheduling is created and the milestone for each phases are identified.

3.2.2 Analysis and Design Phase

The requirements of the project are defined in this phase. The problem is analysed to obtain a well-defined requirement that will determine the direction of the project. The activity carried out in this phase are as below:

- Perform literature review

Previous and related work of this project is reviewed. The methodology, techniques, hardware and software used and parameter of the previous work is studied to serve as a guide to this project.

- Simulate and analyse the transition mechanism attack

The attacks on transition mechanism is analysed to figure out the pattern or signature of the attack. The pattern or the signature will be used to develop the countermeasure in the form of signature based intrusion detection system.

- Perform requirement analysis
During this activity, the data, functional and non-functional requirement is determined. The data requirement indicates what data is input to the system and what output does the system should produce. Functional requirement includes the functions of the system and how it perform its operation. Non function requirement covers the performance and quality aspects of the system.
- Hardware and software analysis
The hardware and software to be used in this project is determine during this activity. Various options of hardware and software is evaluated to identify the one which best suits this project.

In design phase, the requirement from previous phase is translated into design. The activities involved in this project are listed below:

- Architecture / Test bed design
The placement of host devices, routers, connections are determined in this design. How the devices are interconnected are designed during this activity. The architecture in this design will be used to perform attack as well as detection of attacks.
- Software design
The function of detecting an attack and producing alert of the IDS is designed during this activity. The design of the IDS is illustrated using Data Flow Diagram (DFD). Besides, program specifications which includes description, input/output and are created during this activity.

3.2.3 Implementation phase

The end product of this project starts to take shape during this phase. The design from the previous phase is implemented during this phase. Activities during this phase are as below:

- Setting up the architecture / test bed

The test bed design is set up physically during this activity. The network as well as the transition mechanism is configured. The OS of the host devices is installed. The connectivity is tested to ensure the devices are working as intended.

- Develop the intrusion detection system

The software design is developed into a working system. The IDS developed is ran on the monitoring host device.

3.2.4 Testing phase

This is the crucial phase of the project. This phase determine the effectiveness of this project. The activities involved in this phase includes test plan, test design and test results analysis

- Test plan

During this activity, the personnel involved during testing, testing environment, test schedule and testing strategy is determined. Testing environment is the environment or location the testing is carried out. Test schedule indicate how many times the test is repeated and the duration of the test.

- Test design

In this activity, the attack simulation is designed to test the effectiveness of the IDS.

- Test result analysis

The test is carried out during this activity and the data and result is collected. The result is analyzed and compared with the expected result. The failed test case is studied and documented.

Changes were made at the design and implementation phase to solve the failed test case.



3.3 Project Schedule and Milestones

The project schedule is created to ensure efficient time management and to ensure the project can be completed within the stipulated period. Each task has been allocated a time period and the milestone of the project is determined. Table 3.1 below shows the scheduling of this project.

Table 3.1 Project Schedule and Milestones

Week	Activity	Output
1	Seek and decide on a project title and develop a proposal.	Title is chosen. Developed a proposal.
22 - 26 Feb	Submit completed proposal to supervisor for approval.	Proposal submitted. Supervisor is assigned. Project suggestion form.
	Submit approved project title to PSM committees.	Project suggestion form is submitted.
2 29 Feb - 4 Mar	Correction of proposal.	Received their approved proposal form for correction.
3 7 - 11 Mar	Begins with project Chapter 1.	Chapter 1 Introduction.
4 14 - 18 Mar	Complete and submit Chapter 1 for supervisor evaluation. Begins with project Chapter 2.	Supervisor checked on Chapter 1. Correction on Chapter 1. Chapter 2 Literature Review.

Week	Activity	Output
5 21 - 25 Mar	Begins studies on related work and previous research for Chapter 2.	Progress report on Chapter 2.
6 28 Mar - 1 April	Complete and submit Chapter 2 for supervisor evaluation. Begins with project Chapter 3.	Supervisor checked on Chapter 2. Correction on Chapter 2. Chapter 3 Methodology.
7 4 - 8 April	Demonstration for Chapter 3 and begins with Chapter 4.	Supervisor checks on Chapter 3.
8	MID SEMESTER BREAK	
9 18 - 22 April	Demonstration of project and progress on Chapter 4.	Report progress of Chapter 4.
10 25 - 29 April	Demonstration of project and progress on Chapter 4.	Report progress of Chapter 4.
11 2 - 6 May	Demonstration of project.	Submit Chapter 4. Demonstration of project.
12 9 - 13 May	Demonstration of project and done prepare PSM Report.	Demonstration of project is evaluated.

Week	Activity	Output
13 16 - 20 May	Demonstration of project and done prepare PSM Report.	Demonstration of project is evaluated.
14 23 - 27 May	Demonstration of project and done prepare PSM Report.	Demonstration of project is evaluated. Submit full report.
15 30 May - 3 June	Final Presentation	Final evaluation for supervisor and evaluator.
16 6 - 10 June	Correction draft report based on comments from supervisor and evaluator during the final presentation session.	Correction of PSM 1.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ID	Task Name	Mar 2016				Apr 2016				May 2016				Jun 2016	
		6/3	13/3	20/3	27/3	3/4	10/4	17/4	24/4	1/5	8/5	15/5	22/5	29/5	5/6
1	CHAPTER 1: Introduction	■													
2	CHAPTER 2: Literature Review		■												
3	CHAPTER 3: Methodology				■										
4	CHAPTER 4: Analysis & Design							■							
5	Presentation & Submission									■					
6	Report Correction													■	

Figure 3.2 Project Gantt Chart

3.4 Conclusion

The methodology describes the steps and processes involved in conducting this project from beginning till the completion which assist in delivering the outcome expected in this project. Thus proper methodology is crucial and it ensures that the project is executed in a systematic way. The subsequent chapter will discuss in detail on the analysis, design, implementation and testing phase mentioned above.



CHAPTER IV

ANALYSIS AND DESIGN



4.1 Introduction

Analysis and design is the crucial stage during development. It helps to define a clear idea on the product being developed. In this chapter, the analysis and design of IPv6 transition mechanism is discussed in detail. The analysis phase involves gathering requirements for the system. Once requirement is defined, the design phase commences. The design phase includes designing the network architecture as well as the software design.

4.2 Problem Analysis

IPv6 is an improved version of IPv4 that provide features such as stateless auto-configuration and faster routing. Growing number of organizations adopted IPv6 into their network. However, IPv6 and ICMPv6 has inherent protocol weaknesses that can be exploited and expose the network to threats. Toolkit to attack the weakness of mentioned protocol is widely available on the internet. The Hacker Choice-IPv6 or known as THC-IPv6 is an example of attack toolkit.

Conventional security mechanism such as Intrusion Detection System supposedly would be able to detect these attacks. In order to avoid detection, attacker launch attack through the IPv6 transition mechanism tunnel. For example, in 6to4 tunnelling, the attack is placed in the packet inside the payload which allows it to pass the IDS undetected. Intrusion detection system should be improved to be able to detect attack inside the transition mechanism.

4.3 Requirement Analysis

4.3.1 Data Requirement

The data or input that will be received by the attack detector is the mirrored traffic or packets from the tunnel. The attack detector analyse the packets and generate alert notification if an attack is detected. Figure 4.1 is the block diagram of the attack detector.

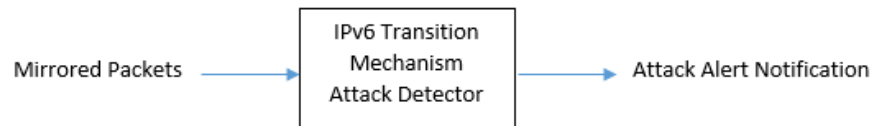


Figure 4.1 Block Diagram of IPv6 Transition Mechanism Attack Detector

4.3.2 Functional Requirement

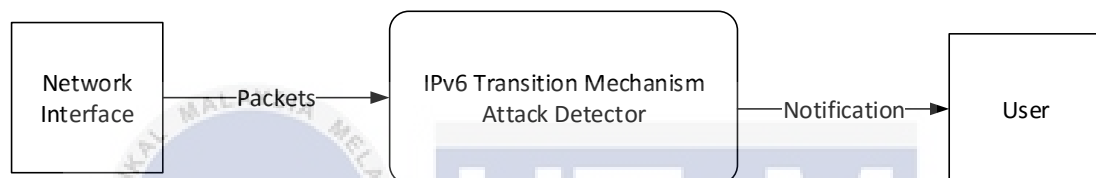


Figure 4.2 IP Transition Mechanism Attack Detector Context Diagram

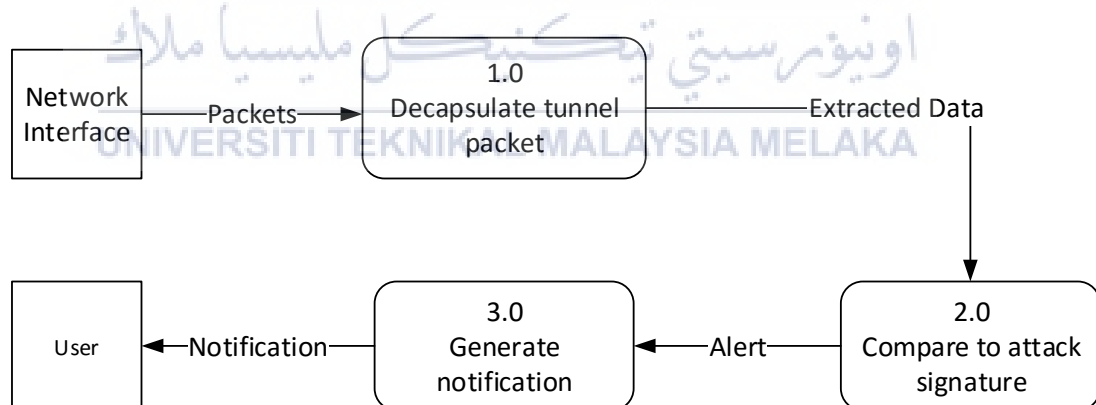


Figure 4.3 IP Transition Mechanism Attack Detector Data Flow Diagram

The IPv6 Transition Mechanism detector operates by receiving packets from the network interface. Since it receives packet from a tunnel, the packet is decapsulated to obtain the payload packet. The payload packet will be analysed by comparing it to attack signature. If an attack is detected, an alert is generated to notify user.

4.3.3 Other Requirement

4.3.3.1 Hardware Requirement

- **IPv6 supporting Router**

Router is a layer 3 networking device that route packets across computer networks. Routers interconnect multiple networks. Routers forward packet using IP protocols. Routers are used in this research to simulate IPv4 cloud. Besides, router is used to create IPv6 transition mechanism 6to4 tunnel that interconnects two IPv6 island over IPv4. The model of the router used is Cisco 2800 series router.

- **Switch**

Switch is a layer 2 networking device that is used to connect multiple networking device together. It receives, process and forward data frame towards the destination using frame switching. Switch is used in this research to replicate the traffic in the tunnel by using port mirroring function of the switch. The attack detector will be connected to the mirror port. Switch model used is Cisco 2960 series.

- **Computer**

Computer running Kali Linux operating system is used in this project. Penetration tools provided in Kali Linux operating system is used to launch the attack. Another computer is used to run the attack detector and sniff for attacks. Multicore CPU and at least 3 GB of RAM is needed in the computer running the attacker detector as the process requires large amount of resource. The third computer is the target computer.

4.3.3.2 Software Requirement

- **PuTTY**

PuTTY is an open source secure shell client application. The application is used in this research to access the router using console cable as well as accessing it remotely using secure shell. Configuration of the router is done through this application. It is also used to remotely accessed another computer in the test bed.

- **Python**

Python is an open source programming language. Python can be run on almost any platform. Python is also fast making it suitable for this research. Besides, Python provide high number of third party package that can be implemented. Python is used to program the attack detector.

- **Scapy**

Scapy is an open source packet manipulation program. It is used in this research to decode captured packets.

4.4 High Level Design

4.4.1 IPv6 Testbed Architecture

This section describes the setup and the configuration of the network environment that will be used as testbed to test the effectiveness of the IDS in detecting attacks.

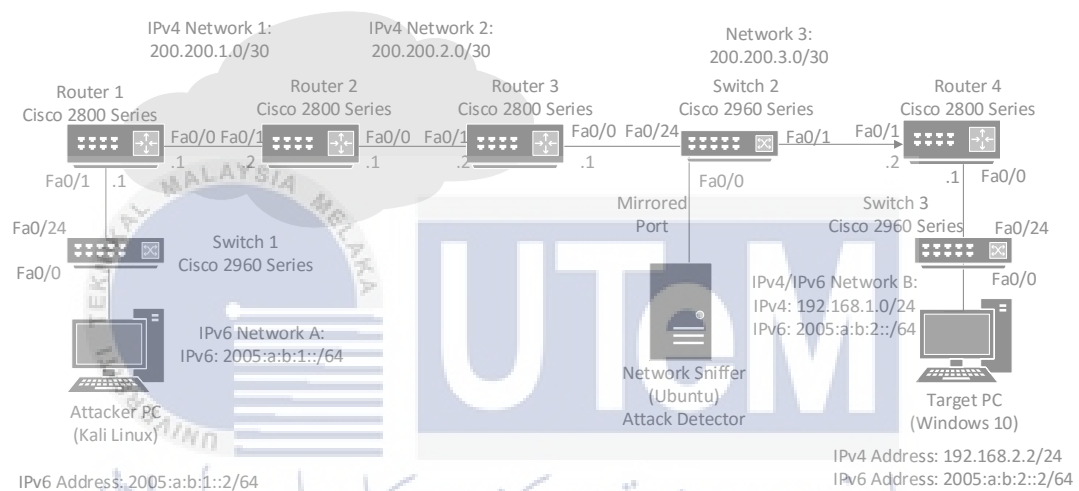


Figure 4.4 Network System Architecture

Figure 4.4 above shows the overall design of the network architecture. In the architecture, IPv6 network A and IPv6 network B are isolated by IPv4 only networks. The connection between Router 1 and Router 4 only support IPv4. Thus, to enable connection between the two IPv6 island, 6to4 tunnelling method of transition mechanism is created to connect Router 1 to Router 4. Router 2 and Router 3 is the intermediary router used to mimic an IPv4 cloud that forwards tunnelled packet towards its destination. A switch is placed in between Router 3 and Router 4 and the traffic in between the router is mirrored to the IDS. Attack will be done from attacker PC running Kali Linux operating system in IPv6 network A to target PC in IPv6 network B through the 6to4 tunnel.

4.4.2 Physical and Logical Network Design

Physical Design

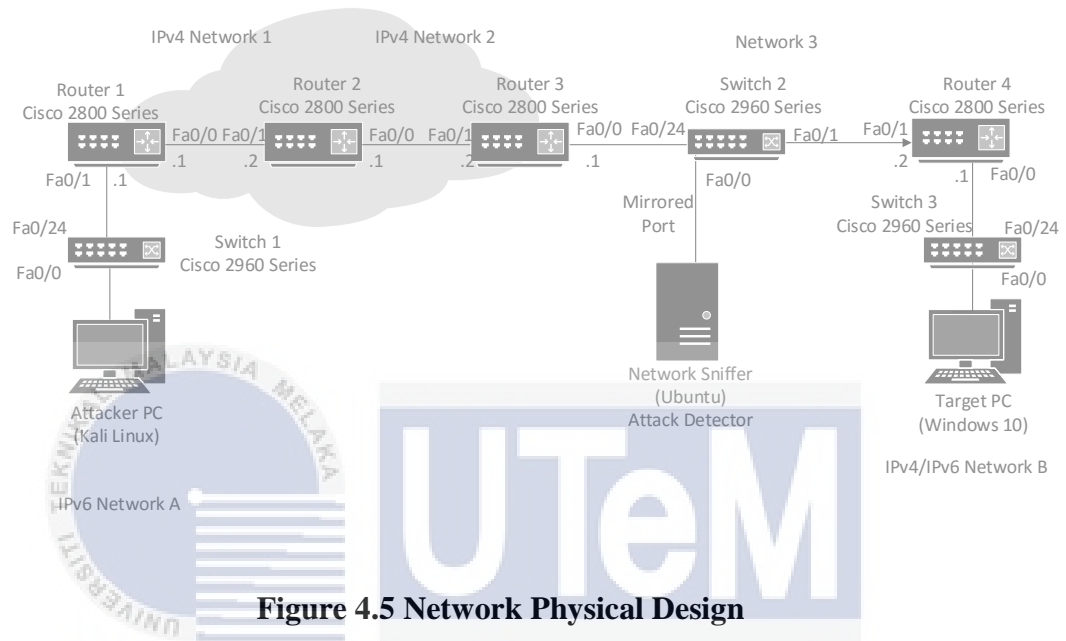


Figure 4.5 Network Physical Design

Figure 4.5 above illustrates the physical setup of the network. The network is made up of 4 routers with each router having 2 Fast Ethernet interfaces. The router used in this research is Cisco 2800 series router and the switch used is Cisco 2960 series. All network connection is done using Ethernet UTP Cat5e cable. In IPv6 Network A, a pc is connected to the Fa0/0 of the switch and the Fa0/24 interface of the switch is connected to the router's Fa0/1 interface. Similar setup is done at IPv6 Network B. Router 2 interfaces is used to connect to the adjacent routers. Router 3 is connected to Switch 2 port Fa0/24 and port Fa0/0 is configured as mirrored port of port Fa0/24. The mirrored port is connected to the PC running IDS application. Router 4 port Fa0/1 is connected to Switch 2's port Fa0/1.

4.4.3 Logical Design

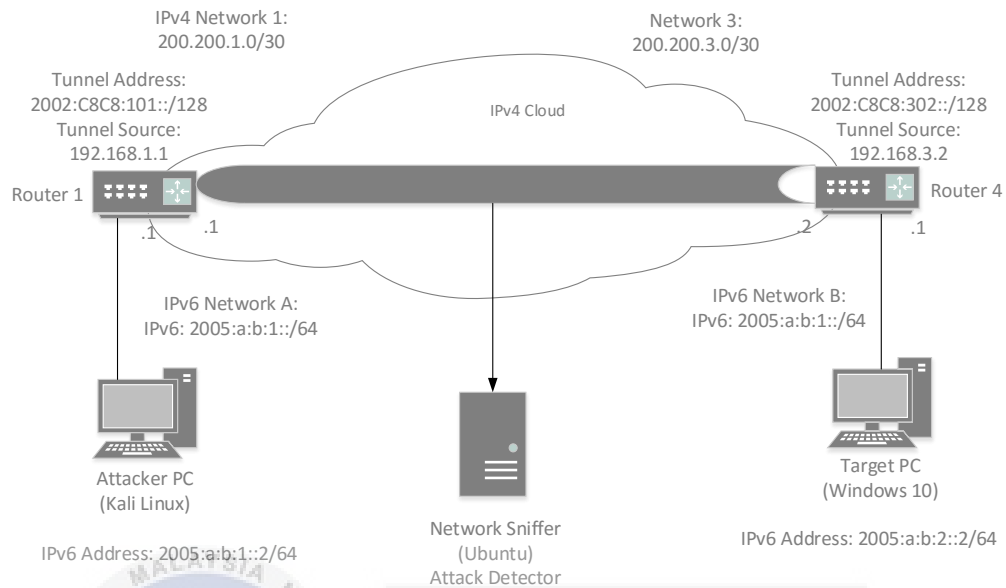


Figure 4.6 Logical Network Design

Figure 4.6 above shows the logical design of the network. Since the network between Router 1 and Router 4 only supports IPv4, IPv6 traffic between IPv6 network A and IPv6 network B is tunnelled in 6to4 tunnelling. The IPv6 packet is encapsulated in IPv4 packet as it exits Router 1 towards Router 4. The tunnel uses 6to4 prefix 2002::/16. The IP address of cloud facing interface of Router 1 is 200.200.1.1 and it is the tunnel source. Thus the tunnel IPv6 address of Router 1 is derived from the IPv4 address and the address is 2002:c8c8:101::/128. The IP address of the cloud facing interface of Router 4 is 200.200.3.2 and the tunnel IPv6 address is 2002:c8c8:301::/128. The IPv6 network address of IPv6 network A is 2005:a:b:1::/64 while the network address for IPv6 network B is 2005:a:b:2::/64.

4.5 Software Design

This section describes in detail the functions of the application in the form of program specification. The flow chart and program specification is listed in Figure 4.7 below

Flowchart

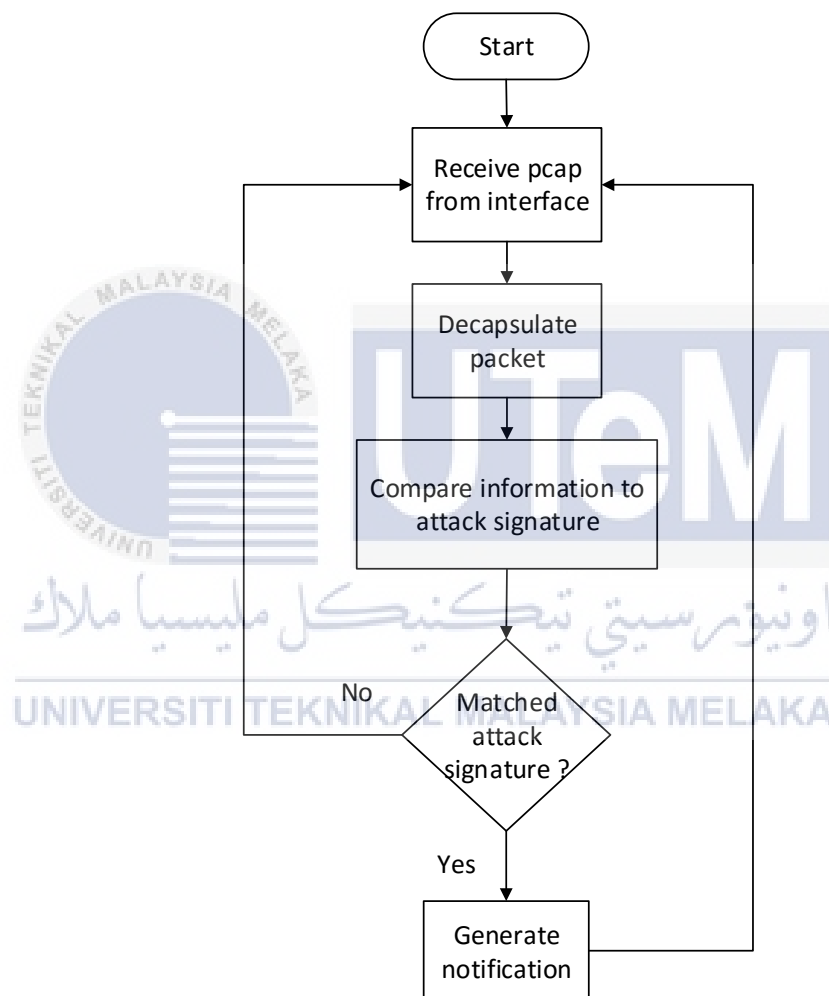


Figure 4.7 Program Flowchart

Program Specification:

Function 1.0	:	De-capsulation of tunnel packet
Descriptions	:	The application receives mirrored tunnel traffic. The IPv6 packet is encapsulated in IPv4 packet. The attack is in the IPv6 packet. In order to perform inspection, the tunnelled packet is de-capsulated and the header field information is extracted
Input	:	Captured network traffic
Output	:	IPv6 packet
Pseudocode	:	<ol style="list-style-type: none"> 1. Receive pcap file from interface 2. Use scapy to extract information 3. Send to information to the next module
Function 2.0	:	Compare to attack signature
Descriptions	:	The IDS type implemented is signature based. The characteristic or the signature of the attack is stored in the application. The captured traffic is compared to the signature.
Input	:	IPv6 packet information
Output	:	Activation of notification module

- Pseudocode : 1. The information received is compared to attack signature
- A. Detect *thcsyn6* DOS attack
 - i. If there is high rate of packet with only the SYN flag is set
 - B. Detect *denial6* DOS attack
 - i. If there is high rate of packet with ipv6 extension hop-by-hop header with a router alert, and the header has unknown options
 - ii. If there is a high rate of packet with ipv6 extension destination header and this header has unknown options
 - iii. If there is high rate of ICMPv6 packet with AH header.
 - C. Detect *sendpees6* DOS attack
 - i. If the packet has ICMPv6 neighbour discovery/neighbour solicitation field, neighbour discovery option source link address and the payload is longer than 150.
2. If an attack is detected, the detection is logged and the notification module is activated

Function 3.0 : Generate Notification

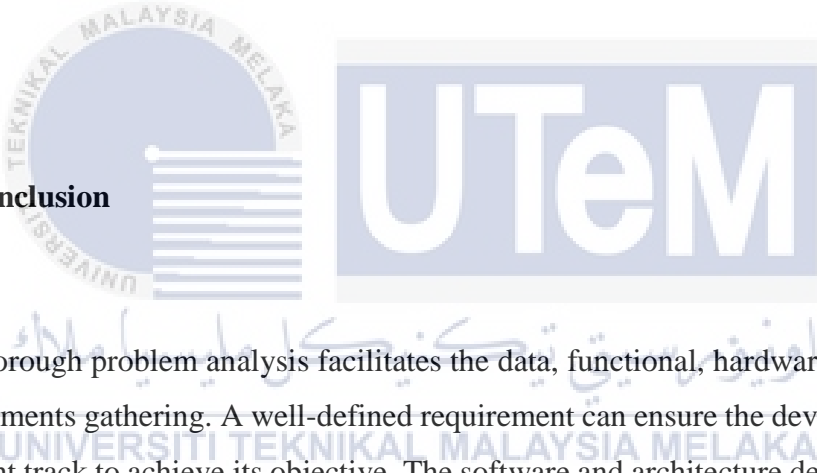
Descriptions : This module generate notification to the user once attack has been detected

Input : Detected attack information

Output : Notification

Pseudocode :
1. Attack information is displayed in the console
2. Popup is generated containing the attack information

4.6 Conclusion



The thorough problem analysis facilitates the data, functional, hardware and software requirements gathering. A well-defined requirement can ensure the development is on the right track to achieve its objective. The software and architecture design facilitates the implementation process. The implementation phase and activity will be discussed in the next chapter.

CHAPTER V

IMPLEMENTATION



This chapter elaborates on the implementation of this IPv6 Transition Mechanism DOS Attack Detector. The analysis and design made in the previous chapter is applied and implemented in this chapter. The environment setup and implementation status for each component is described in this chapter.

5.2 Environment Setup

The environment setup of the testbed in this project is as below:



Figure 5.1 Environment Setup

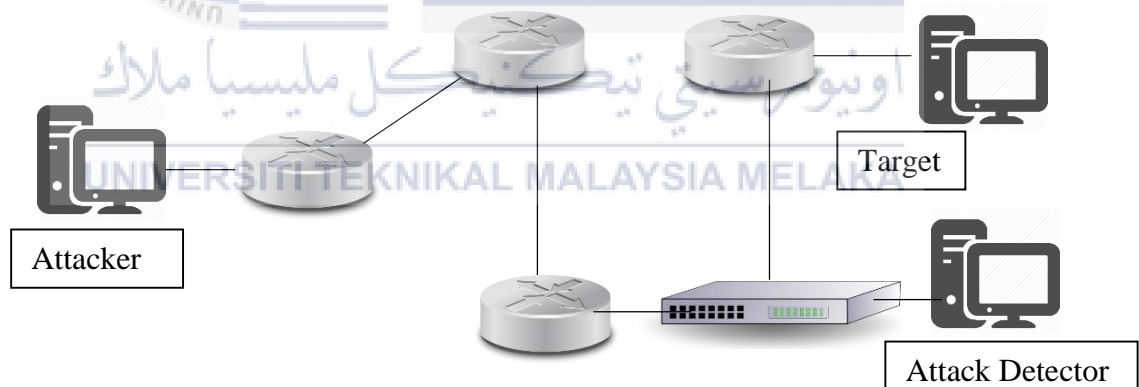


Figure 5.2 Environment Setup Illustration

Figure 5.2 above shows the setup and placement of hardware in this project. The routers are connected together to form an IPv4 cloud with the border router as the 6to4 tunnel endpoint. The network configuration is as Figure 4.6 Logical Network Design from the previous chapter. The Attacker PC is used to launch attack to the Target PC. The 6to4 tunnel traffic is mirrored to the computer running the IPv6 Transition Mechanism DOS Attack Detector to detect the presence of attack.

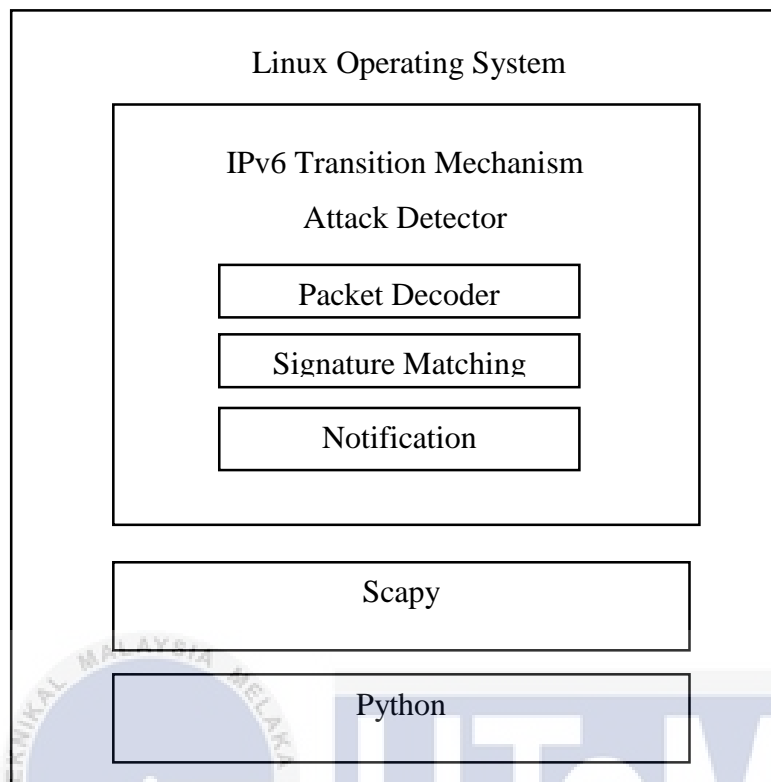


Figure 5.3 Software Environment

Figure 5.3 above shows the software environment of the IPv6 Transition Mechanism DOS Attack Detector software. The software will run inside a Linux Ubuntu operating system. The software is compiled and run using python. Scapy provides a portable framework for low level network monitoring. It is used to capture the packets in the network.

5.3 Attack Signature Identification

This section discusses the signature of the attacks that is involved in this project. The attacks selected in this project are denial6, sendpees6 and thcsyn6.

Denial6

Denial6 is a form of Denial-of-Service attack from the *THC-IPv6* attack toolkit. There are 7 types of attack method in the denial6. 3 attack methods can be performed over transition mechanism. The attack is listed below:

- Denial6 Test case 2: Large destination header filled with unknown option



```

harris@ubuntu: ~
|  optdata= '\x00\x00\x00\x00\x04\x0b\x00\x00\x00'
|  ###[ Scapy6 Unknown Option ]###
|  otype= Tunnel Encapsulation Limit [00: skip, 0: Don't change en-r
oute]
|  optlen= 12
|  optdata= '\x00\x00\x00\x00\x04\r\x00\x00\x00\x00\x04\x0e'
|  ###[ Pad1 ]###
|  otype= Pad1 [00: skip, 0: Don't change en-route]
|  ###[ Pad1 ]###
|  otype= Pad1 [00: skip, 0: Don't change en-route]
|  ###[ Pad1 ]###
|  otype= Pad1 [00: skip, 0: Don't change en-route]
|  ###[ Pad1 ]###
|  otype= Pad1 [00: skip, 0: Don't change en-route]
|  ###[ Scapy6 Unknown Option ]###
|  otype= Tunnel Encapsulation Limit [00: skip, 0: Don't change en-r
oute]
|  optlen= 15
|  optdata= '\x00\x00\x00\x00\x04\x10\x00\x00\x00\x00\x04\x11\x00\x00
0\x00'
|  ###[ Pad1 ]###
|  otype= Pad1 [00: skip, 0: Don't change en-route]
|  ###[ Scapy6 Unknown Option ]###

```

Figure 5.4 Denial6 Test Case 2 Packet Capture

Figure 5.4 shows the destination header of captured denial6 test case 2 packet. As shown in figure, the destination header is filled with unknown option and padding with the aim of wasting processing resource of the victim and cause Denial of Service under heavy load. This attack is identified by the unknown options in the destination header.

- Denial6 Test case 5: AH header and ICMPv6

```

harris@ubuntu: ~
proto= ipv6
chksum= 0x20fd
src= 200.200.1.1
dst= 200.200.3.2
\options\
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 32
  nh= AH Header
  hlim= 254
  src= 2005:a:b:1::2
  dst= 2005:a:b:2::1
###[ AH ]###
  nh= ipv6_icmp
  payloadlen= 2
  reserved= 0
  spi= 0x0
  seq= 0
  icv= '\x00\x00\x00\x00\x80\x00\x89\xed\xfa\xce\xba\xbe\x00\x00\x00\x00
0\x00\x00\x00\x00'

```

Figure 5.5 Denial6 Test Case 5 Packet Capture

Figure 5.5 above shows the capture packet of denial6 attack. This attack floods the victim with ICMPv6 Echo packet with Authentication Header(AH). The victim will have to process AH of all the packet thus depleting the available resource. This type of attack is detected by high rate of ICMPv6 packet with AH

- Denial6 Test Case 7: Large hop by hop header with unknown option

```

harris@ubuntu: ~
|  optdata= '\x00\x00\x00\x00'
|  ###[ Scapy6 Unknown Option ]###
|  otype= 63 [00: skip, 1: May change en-route]
|  optlen= 4
|  optdata= '\x00\x00\x00\x00'
|  ###[ PadN ]###
|  otype= PadN [00: skip, 0: Don't change en-route]
|  optlen= 4
|  optdata= '\x00\x00\x00\x00'
|  ###[ Scapy6 Unknown Option ]###
|  otype= 2 [00: skip, 0: Don't change en-route]
|  optlen= 4
|  optdata= '\x00\x00\x00\x00'
|  ###[ Scapy6 Unknown Option ]###
|  otype= 3 [00: skip, 0: Don't change en-route]
|  optlen= 4
|  optdata= '\x00\x00\x00\x00'
|  ###[ Scapy6 Unknown Option ]###
|  otype= Tunnel Encapsulation Limit [00: skip, 0: Don't change en-r
oute]

```

Figure 5.6 Denial6 Test Case 7 Packet Capture

Thcsyn6

Thcsyn6 is a form of Denial of Service attack from the THC-IPv6 that does SYN flooding. This attack exploits the TCP three-way handshake connection establishment process. When a host receive a SYN flagged packet, it has to respond with SYN ACK or SYN NACK packet. This exhaust the victim resource and affect legitimate traffic.

Protocol	Length	Info
TCP	94	29700 → 9772 [SYN] Seq=0 Win=16440 Len=0
TCP	94	21135 → 63765 [SYN] Seq=0 Win=16440 Len=0
TCP	94	26290 → 61357 [SYN] Seq=0 Win=16440 Len=0
TCP	94	14655 → 27924 [SYN] Seq=0 Win=16440 Len=0
TCP	94	37067 → 51677 [SYN] Seq=0 Win=16440 Len=0
TCP	94	41842 → 43025 [SYN] Seq=0 Win=16440 Len=0
TCP	94	64661 → 59699 [SYN] Seq=0 Win=16440 Len=0
TCP	94	40352 → 38187 [SYN] Seq=0 Win=16440 Len=0
TCP	94	60011 → 37863 [SYN] Seq=0 Win=16440 Len=0
TCP	94	29273 → 42571 [SYN] Seq=0 Win=16440 Len=0
TCP	94	50829 → 25491 [SYN] Seq=0 Win=16440 Len=0
TCP	94	40640 → 64050 [SYN] Seq=0 Win=16440 Len=0
TCP	94	62662 → 19416 [SYN] Seq=0 Win=16440 Len=0
TCP	94	41419 → 13547 [SYN] Seq=0 Win=16440 Len=0
TCP	94	58401 → 48612 [SYN] Seq=0 Win=16440 Len=0
TCP	94	11631 → 9650 [SYN] Seq=0 Win=16440 Len=0
TCP	94	13561 → 27818 [SYN] Seq=0 Win=16440 Len=0

Figure 5.8 Wireshark Snippet During Thcsyn6 Attack

Figure 5.8 shows the flooding of TCP SYN packet during Thcsyn6 attack. Thus, this attack can be detected if there is high rate of TCP SYN in the network,

5.4 Implementation Status

The development status of each for each of the component or the module of the IPv6 Transition Mechanism DOS Attack Detector is described below:

Component 1 : De-capsulation of tunnel packet

Descriptions : The application receives mirrored tunnel traffic. The IPv6 packet is encapsulated in IPv4 packet. The attack is in the IPv6 packet. In order to perform inspection, the tunnelled packet is de-capsulated and decoded.

Duration to complete : 3 weeks

Completion Date : 23 / 7 / 2016

Component 2 : Compare to attack signature

Descriptions : The IDS type implemented is signature based. The characteristic or the signature of the attack is stored in the application. The captured traffic is compared to the signature.

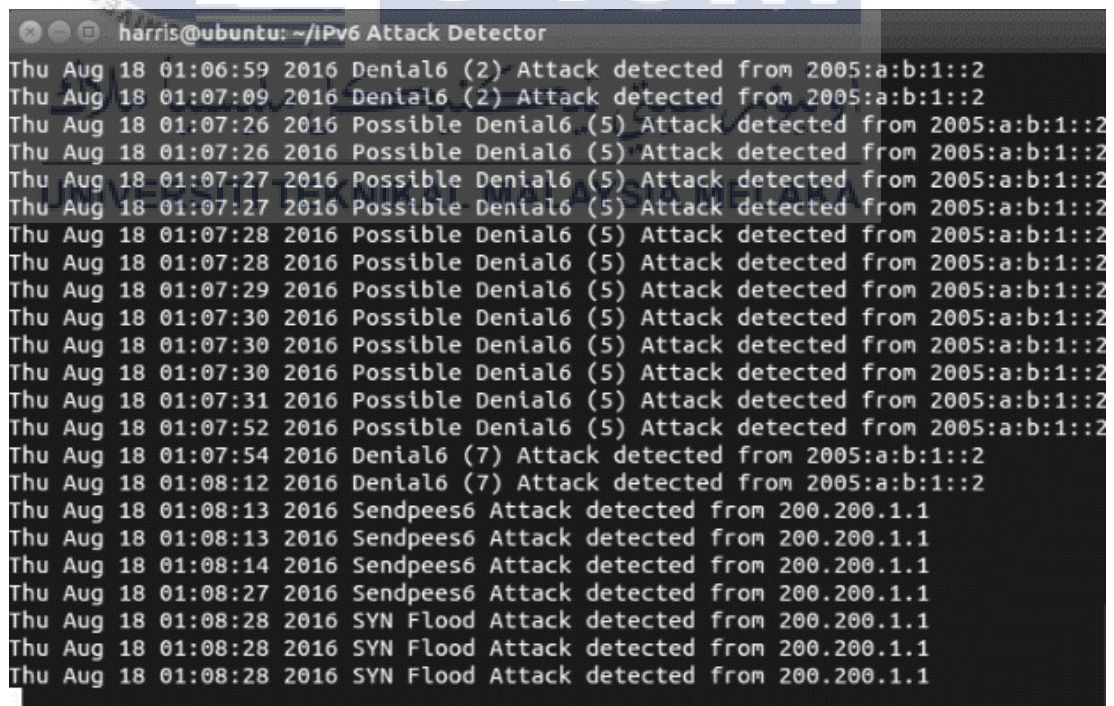
Duration to complete : 2 weeks

Completion Date : 6 / 8 / 2016

Component 3	: Generate Notification
Descriptions	: This module generate notification to the user once attack has been detected
Duration to complete	: 1 weeks
Completion Date	: 13 8 / 2016

5.5 Result

This section discusses the output of the implementation phase. Figure 5.9 below shows the detection of attack by the IPv6 Transition Mechanism Attack Detector.



```

harris@ubuntu: ~/IPv6 Attack Detector
Thu Aug 18 01:06:59 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:09 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:26 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:26 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:27 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:27 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:28 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:28 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:29 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:30 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:30 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:30 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:31 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:52 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:07:54 2016 Denial6 (7) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:08:12 2016 Denial6 (7) Attack detected from 2005:a:b:1::2
Thu Aug 18 01:08:13 2016 Sendpees6 Attack detected from 200.200.1.1
Thu Aug 18 01:08:13 2016 Sendpees6 Attack detected from 200.200.1.1
Thu Aug 18 01:08:14 2016 Sendpees6 Attack detected from 200.200.1.1
Thu Aug 18 01:08:27 2016 Sendpees6 Attack detected from 200.200.1.1
Thu Aug 18 01:08:28 2016 SYN Flood Attack detected from 200.200.1.1
Thu Aug 18 01:08:28 2016 SYN Flood Attack detected from 200.200.1.1
Thu Aug 18 01:08:28 2016 SYN Flood Attack detected from 200.200.1.1

```

Figure 5.9 Output Produced by Attack Detector


```

harris@ubuntu: ~/IPv6 Attack Detector
harris@ubuntu:~$ cd IP*
harris@ubuntu:~/IPv6 Attack Detector $ sudo python ids.py
[sudo] password for harris:
WARNING: No route found for IPv6 destination :: (no default route?)
IPv6 Transition Mechanism Attack Detector is now running...
Thu Aug 18 10:41:24 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:41:40 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:21 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:31 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:35 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:41 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:47 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:44:56 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:02 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:09 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:15 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:22 2016 Denial6 (2) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:28 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
Thu Aug 18 10:45:34 2016 Possible Denial6 (5) Attack detected from 2005:a:b:1::2
^C^Z
[1]+  Stopped                  sudo python ids.py
harris@ubuntu:~/IPv6 Attack Detector $

```

Figure 5.10 Mixed Attack Detection

Figure 5.10 above shows that IPv6 Transition Mechanism Attack Detector able to detect multiple different attacks at the same time.

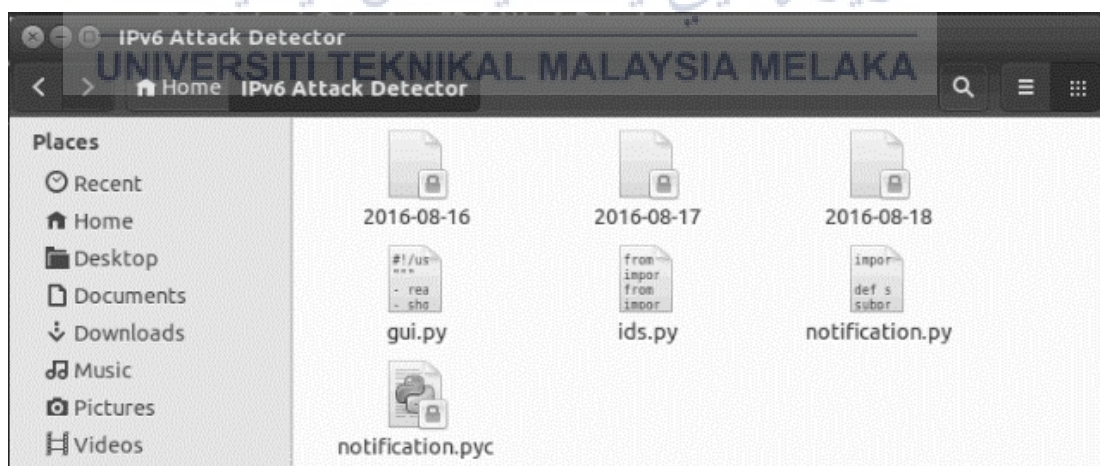


Figure 5.11 Detection Log Produced

Figure 5.11 above shows the detection log produced by the IPv6 Transition Mechanism Attack Detector. The log is produced according to the detection date.

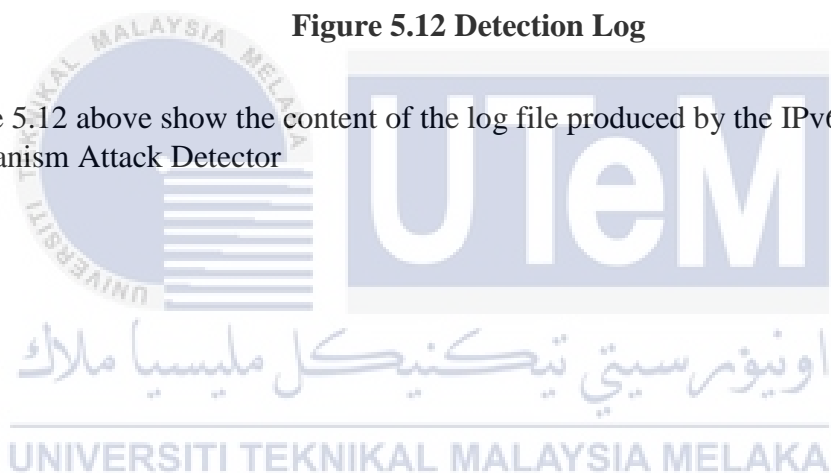
```

2016-08-17 x
Wed Aug 17 14:29:35 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:36 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:36 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:37 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:37 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:37 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:38 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:38 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:39 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:39 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:40 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:40 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:40 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:41 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:41 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:42 2016 SYN Flood Attack detected from 200.200.1.1
Wed Aug 17 14:29:42 2016 SYN Flood Attack detected from 200.200.1.1

```

Figure 5.12 Detection Log

Figure 5.12 above show the content of the log file produced by the IPv6 Transition Mechanism Attack Detector



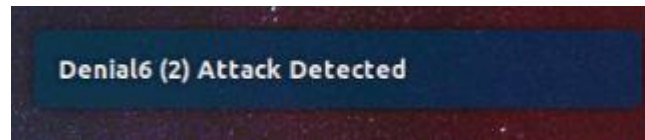


Figure 5.13 System Attack Notification

Figure 5.13 above shows the system notification generated by the IPv6 transition mechanism attack detector when an attack has been detected.

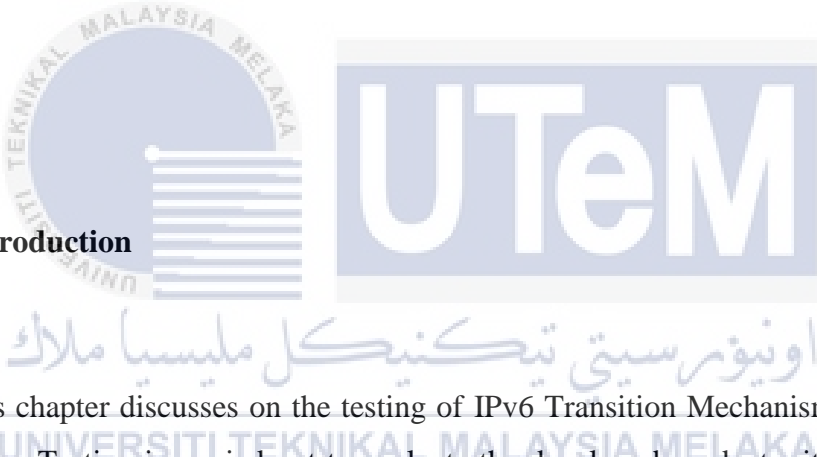
5.6 Conclusion

In conclusion, the activity in the implementation phase transforms the output of analysis and design phase into a product that is the IPv6 Transition Mechanism DOS Attack Detector. The environment setup and software environment is based on the testbed design and software design respectively. The next chapter will discuss further on the testing process done to ensure the product meets its requirement and function as intended.

CHAPTER VI

TESTING

6.1 Introduction



This chapter discusses on the testing of IPv6 Transition Mechanism DOS Attack Detector. Testing is carried out to evaluate the developed product with the intent of finding whether the satisfies its objective. Beside, testing helps to identify the errors or defects software. The test strategy employed is black box testing. This chapter covers the test plan, test strategy, test design and the results.

6.2 Test Plan

This section describes the planning on how the testing will be carried out in this project.

6.2.1 Test Organization

Test organization involves deciding on the personnel involved in the testing process and the assignment of responsibilities to each personnel. Test organization ensures systematic delegation of task. Table 6.1 below shows the test organization.

Table 6.1 Test Organization

Tester ID	Title / Position	Responsibilities
Tester 1	System Developer	Develop, document, manage and testing the system. He/she will ensure the system will run smoothly and systematically based on the requirement before delivered the system to the end user.
Tester 2	Project Supervisor	Act as end user for staff and administrator of the system and give their feedback. All the responses will be a guide to enhance the system.

6.2.2 Test Environment

Test environment is the location or the environment of the testing carried out. The testbed designed in Figure 4.4 Network System Architecture during analysis and design phase is used as the testing environment.

6.3 Test strategy

Test strategy defines the method of testing that will be employed in this project. The test strategy selected in this project is black box testing. Black box testing is a testing method used to examine the functionality of the developed software without the need to know about the code structure. Black box testing determines whether the developed software able to produce the required output. This testing strategy is selected because it ensures the development meets its requirement. Figure 6.1 below shows the overview of black box testing.

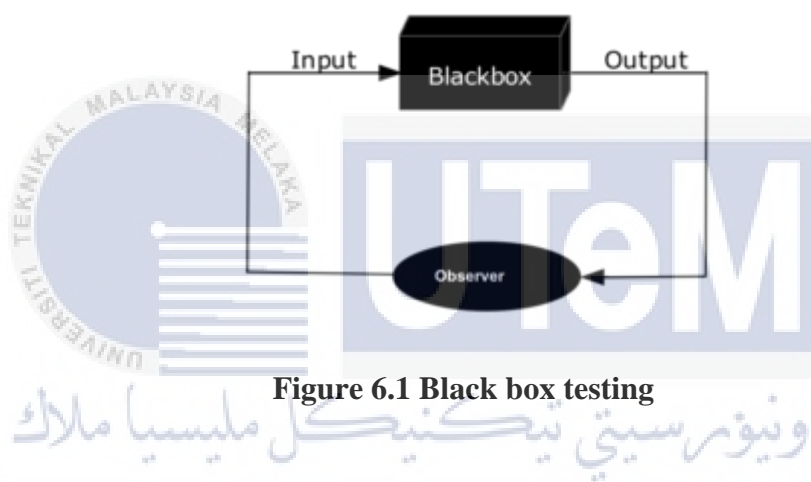


Figure 6.1 Black box testing

6.3.1 Classes of test

There are 2 classes of testing that will carried out namely unit test and functionality test:

i. Unit test

Unit test is the test carried out on individual component of the software by the developer. This test requires understanding of the program code. The aim of the test is to ensure the component is working fine.

ii. **Functionality test**

Functionality test is carried out to test the functions of the program and determine whether the output produced by the function meets the requirement or doesn't.

6.4 Test Design

This section describes the design of the test being carried in the form of test cases and the expected results.

6.4.1 Test Descriptions

Figure 6.2 Testbed connectivity test

Test	Testbed Connectivity Test
Test Case ID	TC01
Test Purpose	Verify design and configuration of the testbed by ensuring there is connectivity between the devices
Test Procedure	Perform ping6 from the attacker to the target network
Expected Result	Successful ping from the attacker to the target network

Figure 6.3 Packet Capture Test

Test	Packet capture and decoding test
Test Case ID	TC02
Test Purpose	Verify that the packet capture and decoding component of the program is able to capture packets and decode the packet.

Test Procedure	<ol style="list-style-type: none"> 1. Temporarily code the program to display all the packet received 2. Use ping6 utility to send ICMPv6 packet towards the target network. 3. Use thc-ipv6 tool to send thcsyn6, sendpees6 and denial6 packet towards target network 4. Perform multiple attack 5. View the terminal for the output of captured packet and its decoded contents
Expected Result	All the captured packet and the decoded content is shown in the terminal

Table 6.2 Signature Matching Test

Test	Signature Matching Test
Test Case ID	TC03
Test Purpose	Verify the developed program able to detect attack launched
Test Procedure	<ol style="list-style-type: none"> 1. Use THC-IPv6 toolkit in the attacker PC to launch DOS attack using thcsyn6 2. Use THC-IPv6 toolkit in the attacker PC to launch DOS attack using sendpees6 3. Use THC-IPv6 toolkit in the attacker PC to launch DOS attack using denial6 4. Check whether the program able to detect the attacks by monitoring output.
Expected Result	Program generate notification when the attack is being carried out detailing the attack.

6.5 Test Results and Analysis

This section documents the results of the test cases carried out during the testing phase.

Table 6.3 Testbed Connectivity Test Result

Test	Testbed Connectivity Test
Test Case ID	TC01
Tester	System Developer
Expected Result	Successful ping from the attacker to the target network
Test Results	Success

The successful ping from the attacker shows that the network configurations is correct. ICMPv6 packet were able to traverse the network from the attacker through the IPv4 network cloud using 6to4 tunnel to the target and return. This verify the operation of 6to4 tunnel.

Table 6.4 Packet capture and decoding test result

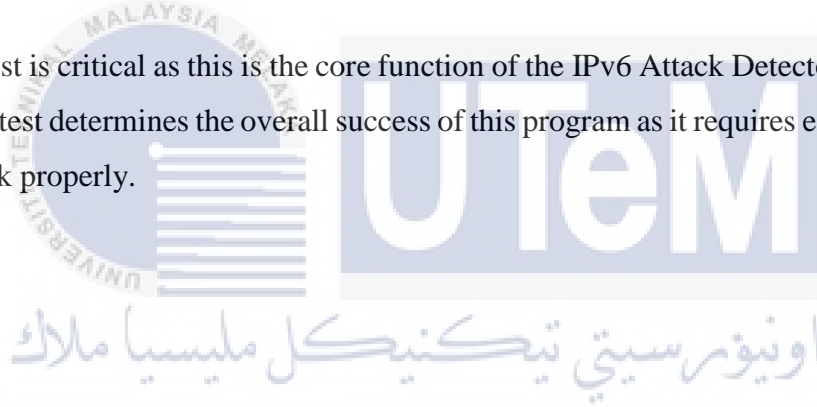
Test	Packet capture and decoding test
Test Case ID	TC02
Tester	System Developer
Expected Result	All the captured packet and the decoded content is shown in the terminal
Test Result	Success

The test result shows that the program able to capture the packets from the interface and decode the packet. This test is crucial to ensure that the program able to capture packet and reduce false negative from the attack detector.

Table 6.5 Signature Matching Test Result

Test	Signature Matching Test	
Test Case ID	TC03	
Tester	System Developer / Supervisor	
Expected Result	Program generate notification when the attack is being carried out detailing the attack.	
Results	thcsyn6	Success
	denial6 (2)	Success
	denial6(5)	Success
	denial6(7)	Success
	sendpees6	Success
	Multiple Attack	Success

This test is critical as this is the core function of the IPv6 Attack Detector. The success of this test determines the overall success of this program as it requires each component to work properly.



6.6 Conclusion

As a conclusion, testing is a very important phase as it verifies that the development and setup done during the implementation phase meets the requirement and objective. The functionality of testbed and each component of the attack detector is tested and documented. The next chapter will discuss the conclusion of the project.

CHAPTER VII

PROJECT CONCLUSION



7.1 Introduction

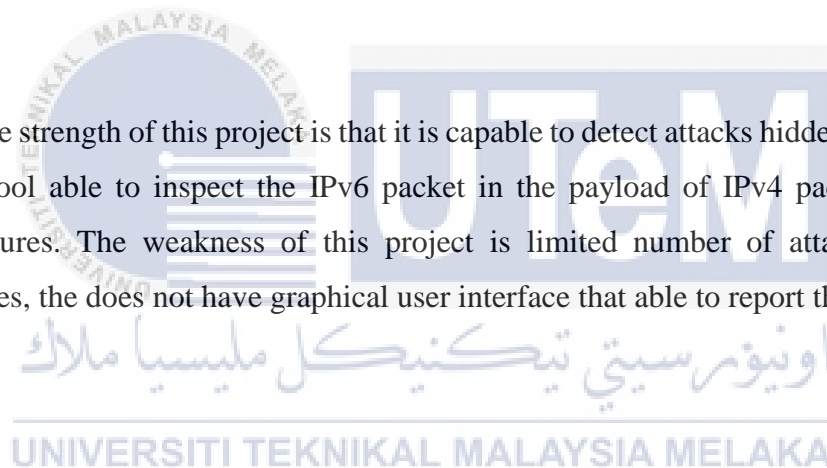
This chapter discusses and conclude overall of this project. The project is summarized by stating its objective and how the objective is achieved. Besides, the contribution of this project and to whom is examined as well as the limitation of the project. The future works of this project is also explored.

7.2 Project Summarization

IPv6 Transition Mechanism can be exploited to perform attack. Thus the objective of this project is to identify the signature of the possible threats in the transition mechanism. The threats are identified by launching the attack and study the pattern to identify the signature.

Once the attacks signature identified, the second objective which is to develop a tool that can detect the presence of the threat achieved by developing the IPv6 Transition Mechanism DOS Attack Detector. A test bed is designed to meet the third objective which is to test the effectiveness of the tool created.

The strength of this project is that it is capable to detect attacks hidden in the tunnel. The tool able to inspect the IPv6 packet in the payload of IPv4 packet for attack signatures. The weakness of this project is limited number of attack signatures. Besides, the does not have graphical user interface that able to report the detection.



7.3 Project Contribution

The development of the attack detector in this project helps organization to minimize the security risk associated by implementing 6to4 IPv6 transition mechanism. The implementation of attack detector prevents attacks to be carried out through the tunnels bypass security layer and go unnoticed by the administrator.

Besides, the test bed designed in this project can be used for further development of security mechanism such as improved IDS or firewall. The test bed can be used to simulate real life environment and test the effectiveness of the development.

7.4 Project Limitation

The IPv6 attack detector performs its operation by scanning all the traffic that passes through the network at strategical location. Since the number of packets can be very high and the attack detector need to scan each of the packet, large amount of processing resources is required. The host PC running the attack detector must have enough processing power to cope the demand of the attack detector. Besides the code has to be efficient to ensure it does not consume too much resource.

7.5 Future Works

There are still plenty room for improvement for the IPv6 Transition Mechanism DOS Attack Detector. One of the area of improvement is the user interface to be more user friendly for novice user. The improved user interface should be able to perform analysis on detection and displays the results to the user.

Besides, the detection mechanism of the attack detector can be improved further by combining the current signature based detection with anomaly based detection. The hybrid of signature and anomaly based detection will be able to detect both known and new attacks.

7.6 Conclusion

Finally, the project successfully meets all the objective by the development of the IPv6 Transition Mechanism DOS Attack Detector as well as the design of a test bed. Improvement can be made in the future to make the IPv6 Transition Mechanism more robust and effective. The attack detector can contribute as a security layer when transition mechanism is being implemented while awaiting the rest of the internet to fully migrate to IPv6.

REFERENCES

- Bahaman, N., Prabuwno, A. S., & Mas'ud, M. (2011). Implementation of IPv6 Network Testbed. *Journal of Applied Sciences*, 11(1), 118-124.
- Çalışkan, E. (2014). *IPv6 Transition and Security Threat Report*. Talinn: NATO Cooperative Cyber Defence Centre of Excellence (CCDCOE).
- Carpenter, B. (2011). RFC 6343: Advisory Guidelines for 6to4 Deployment. Internet Engineering Task Force.
- Carpenter, B., & Moore, K. (2001). RFC 3056: Connection of IPv6 Domains via IPv4 Clouds. Internet Engineering Task Force.
- Chown, T., & Venaas, S. (2011). RFC6104: Rogue IPv6 Router Advertisement Problem Statement. Internet Engineering Task Force.
- Conta, A., & Deering, S. (1998). RFC 2473: Generic Packet Tunneling in IPv6. Internet Engineering Task Force.
- Deering, S., & Hinden, R. (1998). RFC 2460: Internet Protocol, Version 6 (IPv6) Specification. Internet Engineering Task Force.
- Dewitz, S. (1996). *Systems Analysis and Design and The Transition to Objects*. New York: McGraw-Hill.
- Gilligan, R., & Nordmark, E. (2000). RFC2893: Transition Mechanisms for IPv6 Hosts and Routers. Internet Engineering Task Force.
- Hagen, S. (2014). *IPv6 Essentials* (3rd ed.). California: O'Reilly Media.
- Hei, Y., & Yamazaki, K. (2004). Traffic analysis and worldwide operation of open 6to4 relays for IPv6 deployment. Tokyo: IEEE Conference Publication.
- Huitema, C. (2001). RFC3068: An Anycast Prefix for 6to4 Relay Routers. Internet Engineering Task Force.
- Information Sciences Institute, University of Southern California. (1981). RFC 791 : DARPA Internet Program Protocol Specification . Virginia: Internet Engineering Task Force.

- Internet Assigned Number Authority. (2016). *Assigned Internet Protocol Numbers*. Retrieved from <http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>
- Kumar, S. (2007). Smurf-based Distributed Denial of Service (DDoS) Attack Amplification in Internet. San Jose: Second International Conference on Internet Monitoring and Protection.
- Lau, F., Rubin, S. H., Smith, M. H., & Trajkovic, L. (2000). Distributed Denial of Service Attacks. *IEEE International Conference on Systems, Man and Cybernetics*. Nashville: IEEE.
- Lawton, G. (August, 2001). Is IPv6 finally gaining ground? *Computer*, pp. 11-15.
- Li, Z., Das, A., & Zhou, J. (2005). Theoretical basis for intrusion detection. New York: IEEE Conference Publication.
- Naidu, S., & Patcha, A. (2013). IPv6: Threats Posed By Multicast Packets, Extension Headers. *IOSR Journal of Computer Engineering* , 15(2), 66-75.
- Narten, T., Nordmark, E., Simpson, W., & Soliman , H. (2007). RFC4861: Neighbor Discovery for IP version 6 (IPv6). Internet Engineering Task Force.
- Perkins, C. (October, 1996). RFC2003: IP Encapsulation within IP.
- Piskozub, A. (2002). Denial of service and distributed denial of service attacks. Lviv-Slavsko: IEEE Conference Publication.
- Redwan, M., Ramadass, S., & Manickam, S. (2013). Intrusion Detection System in IPv6 Network Based on Data Mining Techniques - Survey. Kuala Lumpur: Institute of Research Engineers and Doctors.
- Roesch. (1999). Snort - lightweight intrusion detection for networks. Seattle: USENIX LISA.
- Whitman, M., & Mattord, H. (2008). *Principles of Information Security* (3rd ed.). Cengage Learning.
- Yu, Z. (2009). Study on Intrusion IPv6 Detection System on LINUX. Wuhan: IEEE Conference Publishing .

Zagar, D., & Grgic, K. (2006). IPv6 Security Threats and Possible Solutions. Budapest: World Automation Congress.

Zulkiflee, M., Robiah, Y., Abu, N., & Shahrin, S. (2012). Improvising Intrusion Detection for Malware Activities on Dual-Stack Network Environment. *World Academy of Science, Engineering and Technology*, 67, 642-649.

