

ANALYSIS OF P2P BOTNETS BEHAVIORS IN SKYPE APPLICATION

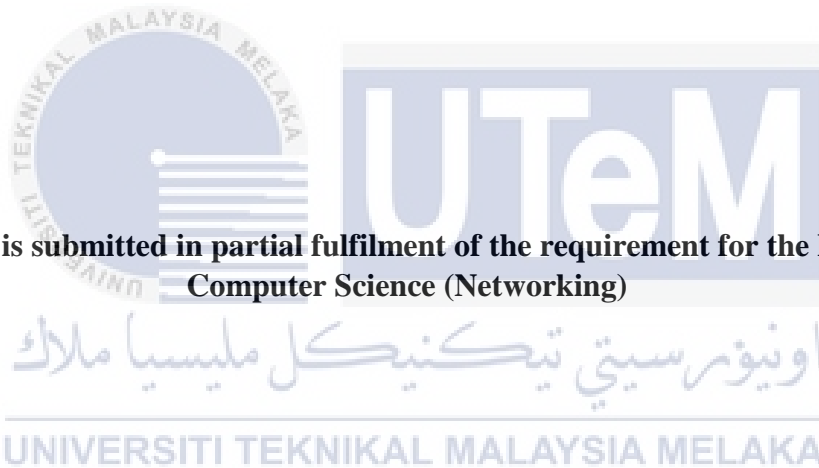
ANIS FARHANI BINTI MOHD KELANA



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ANALYSIS OF P2P BOTNETS BEHAVIORS IN SKYPE APPLICATION

ANIS FARHANI BINTI MOHD KELANA



FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2016

BORANG PENGESAHAN STATUS TESIS

JUDUL: ANALYSIS OF P2P BOTNETS BEHAVIORS IN SKYPE APPLICATION

SESI PENGAJIAN: 2016/2017

Saya: ANIS FARHANI BT MOHD KELANA

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 (/)

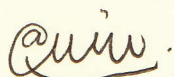
SULIT

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

TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

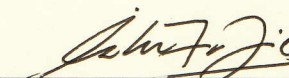
TIDAK TERHAD



(TANDA TANGAN PENULIS)

Alamat tetap: MP 859, JLN 4, TMN MELAKA
PERDANA, 78000, ALOR GAJAH, MELAKA

Tarikh: 26/8/2016



(TANDA TANGAN PENYELIA)

Nama Penyelia: Puan Raihana Syahirah Bt
Abdullah

Tarikh: 26/8/16

DECLARATION

I hereby declare that this project report entitled
ANALYSIS OF P2P BOTNETS BEHAVIORS IN SKYPE APPLICATION

Is written by me and is my own effort and that no part has been plagiarized

Without citations

STUDENT



Anis

DATE: 26/8/2016

(ANIS FARHANI BINTI MOHD KELANA)

اونيور سيتي تیکنیکل ملیسيا ملاک

I hereby declare that I have read this project report and found

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

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 :

Rahana

DATE: 26/8/16

(RAHANA SYAHIRAH BINTI ABDULLAH)

DEDICATION

First and foremost, I will dedicate this project to Allah S.W.T, who always gave his strength and knowledge for me in everyday life.

To my beloved parents, a big thank you for your endless love, support and keep motivate me at all the time.

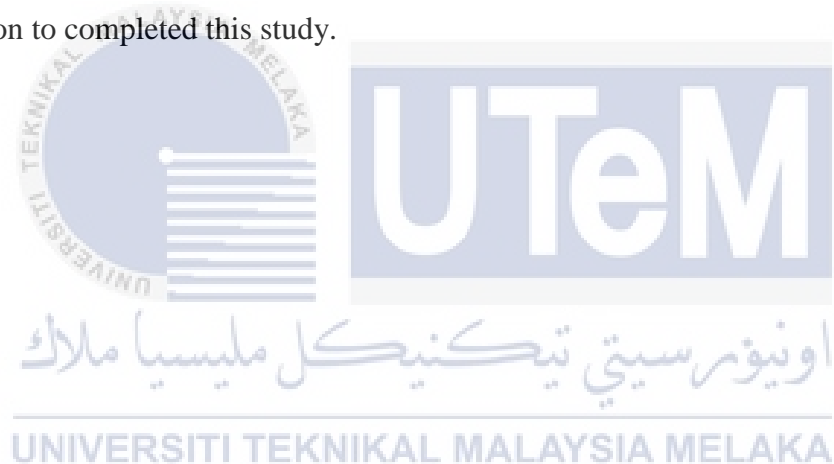
To my supervisor, a big thank you for your guidance and encouragement for me to complete this work.



ACKNOWLEDGEMENTS

I would like to take this opportunity to thank all the persons who have contributed in the different aspects of this study. They have all made it possible for me to complete this study. I would like to acknowledge them for their effort, cooperation and collaboration that have worked towards the success of this study. I would also like to acknowledge the contribution of my family, friend, and my supervisor which always give me a good motivation to complete this long and difficult task in this study.

Next, I deeply grateful to have my supervisor, Madam Raihana Syahirah Binti Abdullah for patiently taking me through the difficult task of this study and always support me with a good motivation to completed this study.



ABSTRACT

Botnets have been recognized as one of the most important threats to the Internet security. They are engaged in DDOS attacks, email spamming and other malicious activities likewise. Traditional botnets usually organized themselves in a hierarchy architecture, which offers professionals opportunities to detect or defend the botnets in their servers. After that, Skype is one of the most used P2P applications on the Internet: VoIP calls, instant messaging, SMS and other features are provided at a low cost to millions of users. Although Skype is a closed source application, an API allows developers to build custom plugins which interact over the Skype network, taking advantage of its reliability and capability to easily bypass firewalls and NAT devices. Since the protocol is completely undocumented, Skype traffic is particularly hard to analyse and to reverse engineer. In this project, focus more on Skype application which is to monitor the normal and abnormal on its network traffic. The case of the "Skype worm" proved to have a high propagation rate, which is spreading almost exponentially during the first days of operation, considering that, as each new person became a victim, all his or her contacts on Skype, Gtalk and other instant messaging systems received these same malicious links.

TABLE OF CONTENT

CHAPTER	SUBJECT	PAGE
	DECLARATION	iii
	DEDICATION	v
	ACKNOWLEDGEMENTS	vi
	ABSTRACT	vii
	TABLE OF CONTENT	viii
	LIST OF TABLE	xi
	LIST OF FIGURE	xii
CHAPTER I	INTRODUCTION	
	1.1 Overview	1
	1.2 Problem Statement	3
	1.3 Objective	4
	1.4 Project Scope	4
	1.5 Project Contribution	4
	1.6 Research Contribution	5
	1.7 Report Organization	5
	1.8 Conclusion	6
CHAPTER II	LITERATURE REVIEW	
	2.1 Introduction	7
	2.2 Peer-to-Peer (P2P)	8
	2.2.1 Definition	8
	2.2.2 Characteristic of P2P	8
	2.2.3 Architecture of P2P	9
	2.2.4 Taxonomy of P2P	9
	2.2.5 Topology of P2P	12
	2.3 P2P Application	12
	2.3.1 Currently Top 10 Most Popular P2P Application	13

2.4 Botnets	16
2.5 Peer-to-Peer Botnets	17
2.5.1 Botnets Behavior Analysis	17
2.6 Overview, Case Study and Taxonomy of P2P Botnets	21
2.6.1 Wireshark Network Analysis	22
2.7 Analysis Technique	22
2.8 Overview and Case Study of P2P Botnets Behaviour in Skype Application.	23
2.8.1 Worm or Malware that Affected the Skype Application	24
2.9 Critical Review of Current Problem and Justification	25
2.10 Proposed Solution	27
2.11 Conclusion	27
CHAPTER III	METHODOLOGY
3.1 Introduction	28
3.2 Methodology	28
3.3 Project Flow	30
3.4 Project Tool and Requirement	31
3.5 Project Schedule and Milestone	32
3.6 Conclusion	35
CHAPTER IV	DESIGN
4.1 Introduction	36
4.2 P2P Botnet in Skype Analysis Approach	36
4.3 Implementation of dataset	38
4.3.1 P2P Botnet Detection Approach	38
4.4 Data Collection	38
4.5 Analysis Result	40
4.5.1 Skype Application	40
4.5.2 TCP	42
4.5.3 Characteristic Analysis	43
4.6 Comparison of P2P Normal and P2P Botnets	45
4.7 Conclusion	50

CHAPTER V	IMPLEMENTATION	
	5.1 Introduction	51
	5.2 Environment Setup	51
	5.2.1 List of Parameters	57
	5.2.2 Assumptions	58
	5.3 Conclusion	58
CHAPTER VI	TESTING AND ANALYSIS	
	6.1 Introduction	59
	6.2 Result and Analysis	59
	6.2.1 Result of Accuracy in each process	60
	6.2.2 Result of Precision in each process	61
	6.2.3 Result of Recall in each process	63
	6.2.4 Comparison result of each process	64
	6.3 Conclusion	65
CHAPTER VII	PROJECT CONCLUSION	
	7.1 Introduction	66
	7.2 Project Summarization	66
	7.3 Project Contribution	67
	7.4 Project Limitation	67
	7.5 Further Project	67
	7.6 Summary	67
	REFERENCES	69
	APPENDIX	70

LIST OF TABLE

TABLE	TITLE	PAGE
Table 1. 1:	Detailed of the Research Problem (RP)	3
Table 2. 1:	TCP Flag and Control Section	25
Table 3. 1:	Software requirement	31
Table 3. 2:	Hardware requirement	32
Table 3. 3:	Milestone	32
Table 4. 1:	Relationship between TCP Flag & Control Section	43
Table 4. 2:	TCP FIN Scan Comparison	46
Table 4. 3:	TCP SYN Scan or known as Half Open ($tcp.flags == 2$)	46
Table 4. 4:	TCP NULL Scan Comparison	49
Table 5.1:	LIST OF PARAMETERS	57
Table 6.1:	COMPARISON RESULTS IN EACH PROCESS	64

LIST OF FIGURE

DIAGRAM	TITLE	PAGE
Figure 1. 1:	Geographic spread of the clicks on bit.ly links	2
Figure 2. 1:	Operational Framework in Literature Review Phase	7
Figure 2. 2:	A Taxonomy of P2P Systems (Quang Hieu Vu, Mihai Lupu, 2010)	10
Figure 2. 3:	A Taxonomy of P2P Systems (Quang Hieu Vu, Mihai Lupu, 2010)	11
Figure 2. 4:	The Decentralized Of P2P Network	11
Figure 3. 1:	System Development Life Cycle (SDLC) Model	29
Figure 3. 2:	P2P Botnet in Skype Analysis Framework	31
Figure 4. 1:	Wireshark Network Protocol Analysers	37
Figure 4. 2:	The flow of Analysis of P2P Botnet in Skype Application.....	37
Figure 4. 3:	P2P Botnet Detection Framework (Raihana Syahirah, 2011).....	38
Figure 4. 5:	Flowchart of each stage in the process	39
Figure 4. 6:	The Skype is running in the Wireshark	41
Figure 4. 7:	The TCP States	42
Figure 4. 8:	Normal TCP connection setup in Wireshark view.....	42
Figure 4. 9:	TCP Flag at packet details view in Wireshark	43
Figure 4. 10:	TCP Fin Scan (Ezzeldin H., 2010)	44
Figure 4. 12:	TCP NULL Scan (Ezzeldin H., 2010)	45
Figure 4. 13:	TCP Fin Scan in P2P normal	46

Figure 4. 14: Shown the abnormal traffic (tcp.flags = = 2).	47
Figure 4. 15: Shown the abnormal port number found in Wireshark.	48
Figure 4. 16: Shown the abnormal IP address which found in the Wireshark	48
Figure 4. 17: TCP NULL Scan in P2P Normal	49
Figure 4. 18: TCP NULL Scan in P2P Abnormal	50
Figure 5. 1: The Outline Diagram for Implementation Phase	51
Figure 5. 2: The malicious traffic that have been run in Wireshark	52
Figure 5. 3: Dataset in Comma Separated Values File (.csv) in Microsoft Excel	53
Figure 5. 4: The interface of RapidMiner Studio version 7.2	54
Figure 5. 5: Design Of the Decision Tree process	54
Figure 5. 6: Design of Naive Bayes process	55
Figure 5. 7: Design of K-Nearest Neighbor (KNN) Process	56
Figure 6. 1: Decision Tree Process Result (Accuracy)	60
Figure 6. 2: Naive Bayes Process Result (Accuracy)	60
Figure 6. 3: K-Nearest Neighbor (KNN) Process Result (Accuracy)	61
Figure 6. 4: Decision Tree Process Result (Precision)	61
Figure 6. 5: Naive Bayes Process Result (Precision)	62
Figure 6. 6: K-Nearest Neighbor (KNN) Process Result (Precision)	62
Figure 6. 7: Decision Tree Process Result (Recall)	63
Figure 6. 8: Naive Bayes Process Result (Recall)	63
Figure 6. 9: K-Nearest Neighbor (KNN) Process Result (Recall)	64



CHAPTER I

INTRODUCTION

1.1 Overview

The outline of this part is about the Botnets that have turned into the essential guilty parties for propelling conveyed dissent of-administration (DDoS) assaults, conveying spam messages, and gathering touchy data on the Internet. Botnets have remote control capacities like Trojan stallions, and they can consequently spread like Internet worms. Analysts likewise find that botnets can be shaped by messages, informal organizations (Athanasopoulos et. al., 2008), and shared (P2P) systems (Starnberger et al., 2008; Nappa et al., 2010). Botnet when all is said in done, are framed in a brought together design and has a main issue of disappointment which is the C&C server. That is, if the C&C server is followed, the whole botnets will be effortlessly identified and close down. To keep away from the shortcoming of brought together engineering, botnet mirror Peer to Peer (P2P) systems engineering and plan a botnet of a P2P control component, with a specific end goal to build its steadiness.

This study concentrates on Skype application which is to screen the typical and irregular on its system activity. The instance of the "Skype worm" demonstrated which leads to have high rate of engendering which is spreading amid in principal days of

operation, considering that, a new individual turned into a casualty, all contacts on the Skype, Gtalk and other texting frameworks got these same vindictive connections

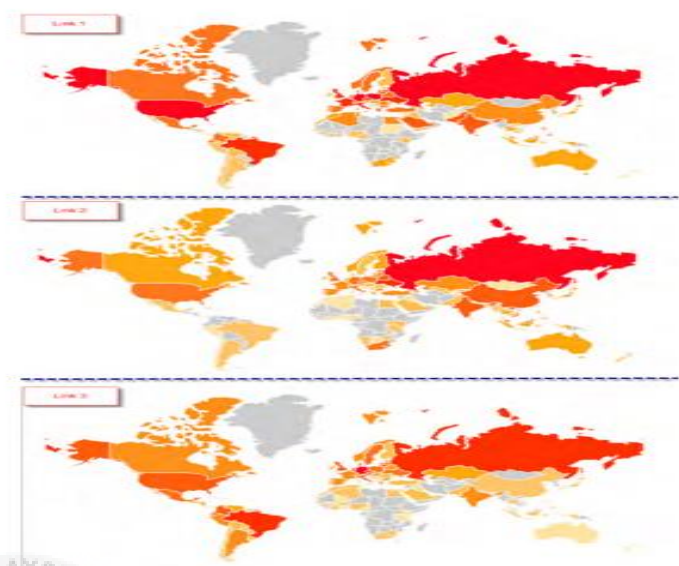


Figure 1. 1: The Geographical spreading of the snaps

The two fundamental dangers included compared to variations of Win32/PowerLoader, which taints the framework and reports back to the C&C (Command and Control Panel), and Win32/Rodpicom, a worm that can spread through various texting applications. The Rodpicom10 worm is a malware that investigates the framework memory to discover the procedures relating to texting programs; it gets to them and sends distinctive spread messages containing pernicious code or some other PC danger to all the casualty's contacts. Rodpicom does not follow up on its own and is normally utilized by different dangers as a proliferation vector. It ought to likewise be noticed that this worm will pick and utilize the framework's dialect keeping in mind the end goal to spread. At the end of the day, the general population behind this risk did not specifically disperse their assault all through the world; but rather the way that the worm sends messages to all the casualty's contacts was in charge of the proliferation levels that achieved right around 750 thousand clients when they tapped on those messages. In the two weeks amid the assault spread, a sum of 69 records distinguished by the ESET items as Win32/Rodpicom.C variations were recognized, relating to 5 unique hashes.

Next, the assault has brought about tainted clients spamming their contact records with messages in both English and German. The English form of the message states: "lol is this your new profile pic?" alongside a URL. The message in German is comparable.

1.2 Problem Statement

In this study, the issue explanation is about the trouble in distinguishing the conduct of Peer-to-Peer (P2P) botnet. After that, the system movement parameter turns into a key issue in the examination. It might bring about the conduct of the worm will trouble to recognize and distinguish. In Skype system activity it might be some trouble to recognize the variation and worm. Next, the botnet assault in shared system will spread quickly and broadly; and the system will influenced by insert to the entire availability in P2P system.

The Research Problem (RP) is classified detailed in table below:

Table 1. 1: Detailed of the Research Problem (RP)

No.	Research Problem
RP1	Demanding way to detect the behavior of Peer-to-Peer botnets
RP2	Difficulty to detect the worm in the Skype network traffic.

1.3 Objective

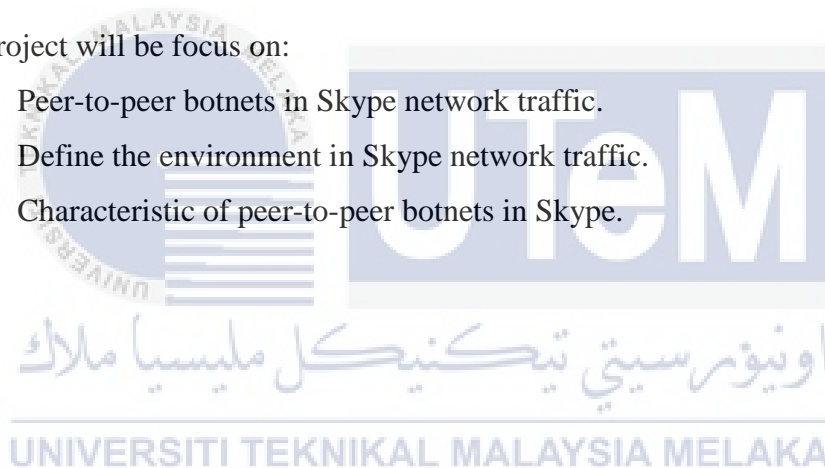
The objectives in this research are:

1. To examine the trademark and conduct of the botnet in distributed (P2P) framework.
2. To identify the network attacks between P2P normal and P2P Botnets in Skype application network traffic.

1.4 Project Scope

The project will be focus on:

1. Peer-to-peer botnets in Skype network traffic.
2. Define the environment in Skype network traffic.
3. Characteristic of peer-to-peer botnets in Skype.



1.5 Project Contribution

This project will be a critical attempt in breaking down irregular information parcels in Skype system activity. After that, this project will examinations the variation or worm, for example, T9000 Malware, Win32/Rodpicom, Win32/Malware!Drop, Win32.Trojan, Ransomeware, and numerous more which can be identify on the skype system activity. This anticipate will likewise enormously advantage to the experts and the individuals who are going to include around there of work as far as examining the sort of variations from the norm of information in system movement. This anticipate can be a rule for them later on examination.

1.6 Research Contribution

There are several research contributions of this study:

1. This study contribute the body of knowledge in P2P botnet detection in Skype Application.
2. In this study will increase the awareness and knowledge on P2P botnet in Skype Application and empower group of participants.
3. This study of P2P botnet will be identifying by using data mining technique.

1.7 Report Organization

Chapter 1, Introduction, this section gives legitimization an outline furthermore foundation data of this examination, issue explanation, objective, scope, venture importance and expected yield.

Chapter 2, Literature Review, this part began with an outline of security patterns. Likewise, it will clarify insights about P2P system, P2P application, P2P Botnets, Skype application, Skype system, Skype Worm and so on.

Chapter 3, Chapter Methodology, this section will give a detailed about the planning process to run the dataset and in this chapter the milestone is used as a guideline to complete the task.

Chapter 4, Chapter design, the chapter will be discussed about the detailed with prove of the detection between P2P Botnet and P2P normal.

Chapter 5, Chapter Implementation, the chapter discussed about the technique used which is Data mining technique to get a result of the performance detection of the dataset.

Chapter 6, Chapter Testing, this chapter will discussed about the testing part and the testing result that have been test in a software that be used to get the result of performance detection of the dataset.

Chapter 7, Chapter Project Conclusion, this chapter will briefly detailed about the summarization, limitation and the further project that will be use in future.



1.8 Conclusion

As conclusion, this chapter introduced about the P2P normal and the P2P botnet that will be detected in Skype environment using the dataset. Next, the dataset will be used continually until the testing part give a good result.

CHAPTER II

LITERATURE REVIEW

2.1 Introduction

In past section, the problem research, the research questions and research objective of the study are unmistakably talked about. In literature review, the related work of the study will be talked about. The goal of this section is to discover a few issues that identified with this study, for example, shared, botnet, Skype application, Skype worm, and analysis method furthermore incorporate past and currently research.

The project title can be breakdown into several item consist of P2P, Botnet, P2P Botnet, Skype , Skype Worm, Analysis and Previous Research. P2P Botnet in Skype Application will focus more on the definition of botnet, and Skype malware that affect the normal network.

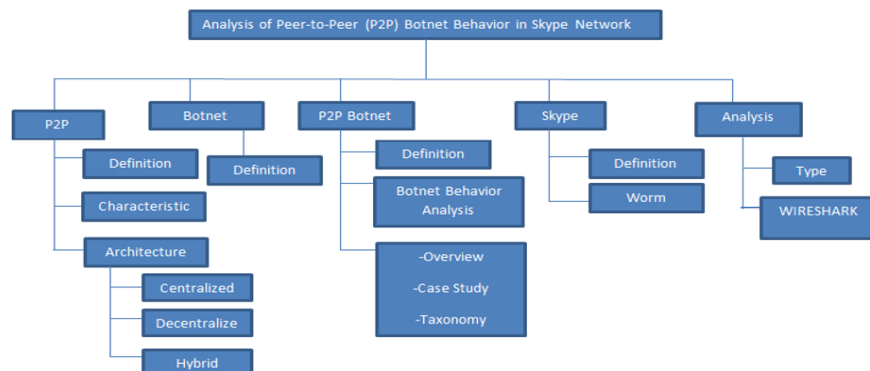


Figure 2. 1: The Framework of the Analysis

2.2 Peer-to-Peer (P2P)

The detail description of the Peer-to-Peer (P2P) is discussed in this section. The definition of the P2P will be defined. Moreover, this section also will find out the characteristic and taxonomy of the P2P itself.

2.2.1 Definition

Each of the peers in peer-to-peer has the same capabilities in communication network. It can begin a correspondence session and it additionally have the capacity to coordinate the trade assets and the administrations between themselves (Margaret Rouse, 2009). Customer and server model is a sort of the P2P model. Each of the customer and server can impart to each other with giving every ability.

In network, the computer, node can be separated into two which is server and client, we call it as P2P. Next, a completely disseminated setup of straight forwardly associated companions are consequences of the non-attendance of concentrated dominant presences in P2P systems. After that, Peer-to-associate (P2P) is a decentralized correspondences model, which is every gathering hosts the same abilities and either get-together can start a correspondence session. The P2P system model permits every node to work as both a client and server.

2.2.2 Characteristic of P2P

- a. Peer is a PC when they can go about as the client and server. Specifically time, the prerequisite of the framework will decided the peer.
- b. In a peer to peer network, the computer will run the same networking protocols and software.

- c. P2P is free framework, since peers can leave or join effortlessly. After join the system, peer ought to have the capacity to trade assets specifically between the peer such as files, storages, data, central processing unit (CPU) power and information.

2.2.3 Architecture of P2P

In the architecture area, as the design of a framework is the foundation of abnormal state actualized applications upon it, a brimming with comprehension of P2P architecture is fundamental to picking up its maximum capacity. Next, the architecture area empowers us to decide the structural variables that are basic to a P2P framework are which is in type of performance, scalability, reliability and other highlights of the framework. Therefore, we commit this part to outline and inspect the architecture of P2P.

2.2.4 Taxonomy of P2P

This taxonomy is categorization is gotten from inspecting existing P2P systems. In general, we can categorize the systems into two general classifications, centralized vs. decentralized, based on the availability of one or more servers, and to what extent the peers depend on the services provided by those servers. Besides these two main categories, there are also hybrid P2P systems which combine both centralized and decentralized architectures to leverage the advantages of both architectures (Hieu Vu and Lupu, 2010). Figure 2.2 shows the taxonomy.

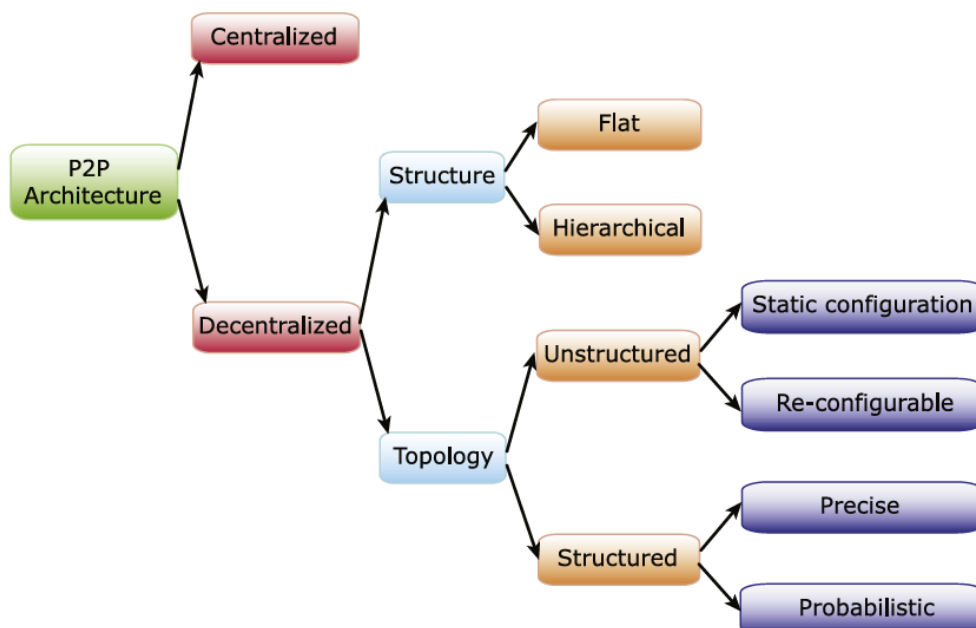


Figure 2. 2: A Taxonomy of P2P Systems (Hieu Vu and Lupu, 2010)

a. Centralized P2P

Like a client-server framework, there are one or more central servers, which peer to find their desire resources or go about as task scheduler to facilitate activities among them. To locate resources, a peer sends messages to the central server to decide the addresses of peers that contain the desired resources or to get work units from the central server straightforwardly. As in every single incorporated framework, these class of P2P frameworks are powerless to malevolent assaults and single purpose of disappointment. Moreover, the centralized server will become a bottleneck for a large number of peers, potentially degrading performance dramatically. Finally, this type of system lacks scalability and robustness.

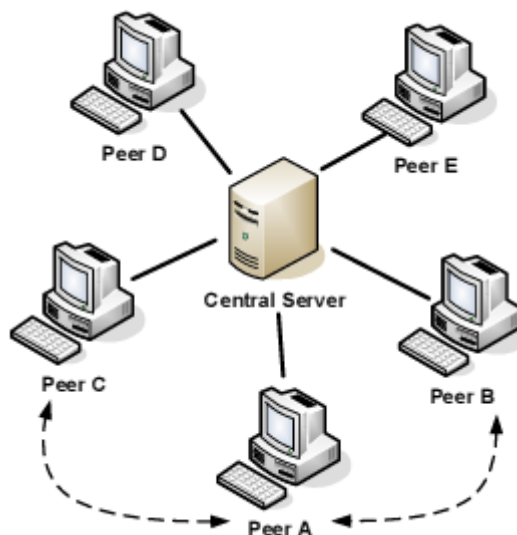


Figure 2. 3: A Taxonomy of P2P Systems (Hieu Vu and Lupu, 2010)

b. Decentralized P2P

In a decentralized P2P framework, peers have measure up to rights and responsibilities. Every peer has just an incomplete perspective of the P2P network and offers information/benefits that might be relevant to just a few queries/peers. The benefits of these frameworks are resistant to single purpose of failure, and possibly enjoy high performance, robustness, scalability. Distribution system can be classified into the structure (Client-Server) and topology (P2P).

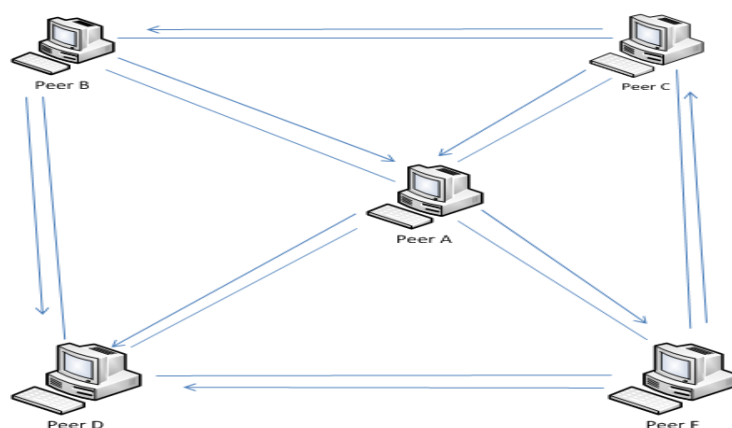


Figure 2. 4: The Decentralized Of P2P Network

2.2.5 Topology of P2P

The P2P model can either be hybrid (structure) or it can be pure (unstructured).

i. Structure (Hybrid Model)

Aimster, Napster, Groove, Magi, Softwax, and iMesh are example application for structure (Hybrid Model). In addition, the SuperPeers can be intermediate solutions in structure, such as KaZaa. Some of the information that contains in SuperPeers, that others peer may not have. Other peers typically lookup information at SuperPeers if they cannot find it otherwise (Milojicic et al., 2002). The advantage of these system is it can provide more guarantee (precise and probability) on search cost (Hieu Vu and Lupu, 2010)

ii. Unstructured(Pure Model)

In pure model, the centralized server does not exist. Gnutella and Freenet are example of pure P2P model (Milojicic et al., 2002). Determination of the neighbors is the key issue in this model. These neighbors can be separate in two type which are fixed and pre-determined statically. However, the neighbors more often determined based on peer's interests (Hieu Vu and Lupu, 2010).

2.3 P2P Application

The P2P application has been particularly popular from last couple of years. P2P frameworks had previously been utilized as a part of numerous application areas; the idea was advanced by file sharing systems, for example, the music-sharing application Napster (initially discharged in 1999). After that, a research from New York University (2003) has report that a distributed application keeps running on your machine, permitting you to interface specifically to other clients' machines and giving different clients the capacity to associate with your machine, so as to exchange records forward and backward between the machines. There are three key attributes characterizing a P2P application, which can find with other companion, ready to impart

substance to other associate and ready to inquiry with other associate. A P2P application is fundamentally utilized for sharing Music, Movies, Games and different files.

2.3.1 Currently Top 10 Most Popular P2P Application

Napster was at its top with about sixty million clients in early 2000, (Brands and Karagiannis, 2009). As of now the phrase peer-to-peer came to be connected with frameworks, for example, Napster. Today, individuals imagined that P2P registering was truly another worldview. Next, file sharing applications frames the initially noted P2P achievement and taps the endlessness of P2P file resources. P2P file sharing is applications that sharing files straightforwardly between network users without the help or obstruction of a central server. Files reside on PCs of clients everywhere throughout the world are shared a little bit at a time between those clients specifically. The decentralized way of P2P file sharing removes the requirement for a central server and removes the likelihood of brought together control on the grounds that P2P file sharing system don't require a central server, they are more versatile and more excess than incorporated file sharing schemes. After that, from the perception and study has been made to the late P2P applications through the few of most recent site, there were numerous P2P file sharing systems in operation. Subsequently, the huge number of P2P applications are accessible in year of 2010 and 2011. There are the most prevalent P2P application that are free and simple to utilize are recorded underneath:



a. Skype

Skype is a computer program that can be utilized to make free voice calls over the Internet to any other individual who is additionally utilizing Skype. It's free and considered simple to download and utilize, and works with generally computers. At Skype, we believe a true P2P system connects all nodes in a network dynamically to participate in traffic routing, processing and bandwidth-intensive tasks that would otherwise be handled by central servers. Next, the Skype team succeeded in P2P communications by leveraging all of

the available resources in a network, all without the need for costly centralized resources.



b. BitTorrent

BitTorrent is the most usually and broadly utilized as P2P system which is much protected and faster, contrasted with other P2P application, BitTorrent is the better. After that, BitTorrent is a convention that empowers quick downloading of substantial files utilizing least Internet data transmission. It costs nothing to utilize and incorporates no spyware or pop-up publicizing. Dissimilar to other download techniques, BitTorrent boosts exchange speed by get-together bits of the file you need and downloading these pieces at the same time from individuals who as of now have them. This procedure makes well known and huge files, for example, videos and TV programs, download much faster than is conceivable with different protocols.

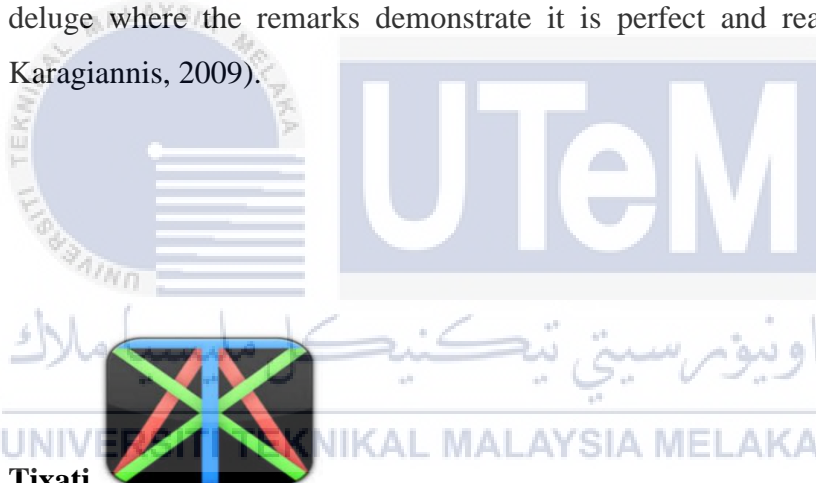
c. μ Torrent

μ Torrent is a freeware, ad-supported, restrictive BitTorrent customer possessed and created by BitTorrent. Next, it otherwise called micro torrent, this is the most famous torrent tool today. μ Torrent has every one of the capacities a download downloader will ever need and it just requires 1 MB of hard drive space and memory. After that, μ Torrent has all the downloading and seeding execution of its rivals however with insignificant effect to whatever is left of PC velocity. μ Torrent is basically just a Windows application however it additionally can be keep running on Mac (Intel) and Linux with Wine.



d. BitComet

BitComet is the third decision because of its superb execution, low RAM use and its capacity to get past a few systems that different customers can't appear. After that, the BitComet provides a nice component with implanted Internet Explorer window to make assets seeking less demanding. It supports HTTP/FTP downloading and utilizes P2P to further increment download speeds. BitComet likewise has a media sneak peek work that can maintain a strategic distance from fakes. In any case this will moderate download speed and fakes can be maintained a strategic distance from by just downloading a deluge where the remarks demonstrate it is perfect and real (Brands and Karagiannis, 2009).



e. Tixati



Tixati is a proprietary cross-stage BitTorrent customer written in C++ which is intended to be light on system resources. After that their engineers discharge standalone and versatile variants with each new customer forms. Next, Tixati is a smooth new torrent program composed by the architect of the WinMiX OpenNap program. Tixati is as of now grasping the movement towards trackerless downpour swarming where the magnet connections, PEX and DHT swarming works exceptionally well in Tixati. The throttling highlights and extensive variety of need conformities will make download rates are in any event as quick as Vuze/Azureus and μ Torrent. The bit field graphs and official dashboard showcase can be effortlessly acquires itself a spot amongst the best deluge customers today (Brands and Karagiannis, 2009).



f. Vuze or Azureus

It has fallen to third because of μ Torrent's ascent. Vuze/Azureus is a Java based system that is utilized for P2P systems. Vuze/Azureus was constantly surely understood for its easy to understand interface and wide assortment of module because of its being open source. An extraordinary element of the project is it likewise has additions of the Vuze/Azureus HD stage for legitimate deluge. Evidently, Vuze/Azureus can use to run media on iPhone, Xbox, or PSP. Vuze/Azureus has great backing with normal upgrades and is accessible for some working frameworks because of its utilization of the cross-stage Java programming language (Brands and Karagiannis, 2009).

2.4 Botnets

A botnet (known as a zombie armed force) is various Internet PCs that, despite the fact that their owners are unconscious of it, have been set up to forward transmissions (counting spam or infections) to different PCs on the Internet.

A botnet is various Internet-associated PCs speaking with other comparable machines in which segments situated on organized PCs communicate and facilitate their activities by command and control (C&C) or by passing messages to each other (C&C may be incorporated with the botnet as P2P). After that, Botnet additionally commonly used to send spam messages, transmit the viruses, and include in different demonstrations of cybercrime.

The Internet worm viruses have various variations and these worms were consolidated with Trojan horses will build up the malicious bots. Besides, this malicious bots have been building into vast number of botnets which is the gigantic and undermining attack systems. For this situation, botnets can comprises of thousands or a huge number of hosts in this way they ready to assault in an exceptionally

disseminated and capable way. Therefore, botnets is developing into a more unpredictable structure and its discovery is a testing issue (Noh S. K. et al (2009)).

2.5 Peer-to-Peer Botnets

P2P botnets are bots that use P2P technology to accomplish certain tasks. P2P Botnets which are more robust and resilient than centralized botnets have emerged of the P2P technology evolve. A peer-to-peer botnet is a decentralized group of malware-compromised machines working together for an attacker's purpose without their owners' knowledge. The P2P is same as in the traditional botnet, which includes a command-and-control server, the bots are typically infected with a Trojan horse and are often used for sending spam or performing DDoS (distributed denial of service) attacks. Peer-to-peer (P2P) botnets have a random organization and operate without a C&C server. Bot software maintains a list of trusted computers (including other infected machines), information drop locations and locations where the machines can update their malware. Thus, P2P botnets has many benefits to be explored as the way to change the network security into the better one.

اوتنور سیتی تیکنیکل ملیسیا ملاک

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.5.1 Botnets Behavior Analysis

The detection of botnets in a network can be categorized through three different types of behavior that will make these behaviors is essential to be understand to prevent, secure and identify botnets intrusion inside the network. The three different types of behavior which are:

i. Host-based behaviors

A host-based IDS monitoring all or parts of the dynamic behavior and the condition of a PC framework. Other than such activities like powerfully assess system bundles focused at this particular host ((optional component with most software solutions commercially available). Next, Host-based behaviors is the behaviors that can likewise be specifically activated inside the infected

machine. Once a bot is running on a host, it compromises software activity. Bots are utilizing a framework/library call which can adjust the register and make or erase systems or projects. Trained the specialists with security learning that can recognize botnets and distinguish areas where there is a high probability of infection.

ii. **Network-based behaviors**

In this type of behaviors, the network-based behaviors will analyze the network activity. After that, the network-based offers security in addition to that provided by traditional anti-threat application, for example, firewalls, interruption detection systems, antivirus software and spyware-detection software. Next, Botmasters need to speak with their bots that cross the network. The majority of the bots are utilizing dynamic DNS query to discover their C&C server since it is the most secure convention. Botnets can utilize IRC or HTTP to communicate. Filtering some particular data on these systems can discover some traded off hosts. The vast majority of big business for the most part banned IRC movement since it is a high hazard to be defiled by malware. Along these lines, if any activity is utilizing the IRC port, it can uncover the nearness of bots. Bots frequently change DNS server to keep the location of the C&C server. Separating the DNS activity can be utilized to discover has who continue changing of DNS name server. Another strategy is by looking at the name of the DNS utilized by botnets that can once in a while be recognized by an unusual name.

iii. **Global correlated behaviors**

Global correlated behaviors is the characteristics that are attached to the basics botnets and can be utilized for an effective identification. These attributes are not going to change unless the botnets is totally re-planned and executed. For example, a famous behavior is known when a C&C server shutdown and each bot will be separated and contact the DNS server with a specific end goal to recover another C&C server address. Along these lines, an expansion of the DNS questions can be found on the system which raises caution of Intrusion

Detection Tools. Apart of using behavioral analysis, a variety of open source tools, commercial tools and techniques are currently used for botnets detection.

The most widely detection and mitigation techniques include the:

i. Flow data monitoring

This techniques utilizes flow-based protocols to get synopsis network and transport-layer data from network devices. Cisco NetFlow is frequently utilized by administration suppliers and ventures to distinguish command-and-control traffic for traded off workstations or servers that have been subverted and are as a rule remotely controlled as individuals from botnets used to launch DDoS attacks, perform keystroke logging, and different types of illicit action.

ii. Honeypots

A honeypot is a PC security mechanism set to distinguish, avoid, or, in some way, counteract attempts at unapproved utilization of data frameworks. Next, a honeypot is a trap that copies a legitimate network, asset or administration yet is in certainty an independent, secure, and observed zone. After that, its essential objective is to draw and recognize malevolent assaults and interruptions. Moreover, its viable more as a reconnaissance and early cautioning framework, it can likewise help security specialists comprehend rising dangers.

iii. DNS log analysis

Botnets regularly depend on free DNS hosting services to indicate a sub domain to IRC servers that have been hijacked by the botmaster. Botnets code regularly contains hard-coded references to a DNS server, which can be spotted by any DNS log examination instrument. So that, if such administrations are distinguished, the whole botnets can be disabled by the DNS server administrator by guiding offending sub areas to a dead IP address. While this procedure is powerful, it is likewise the hardest to actualize since it requires cooperation from third-party hosting providers and name registrars.

iv. Anomaly detection

Anomaly detection is characterize of what normal traffic is like and then look for deviations. Any burst of scanning activity on the network from zombie machines can be detected and blocked. Anomaly detection can be effectively used on the network as well as on endpoints such as servers and laptops. On endpoints, suspicious activity and policy violations can be identified and infections prevented. The technique of anomaly detection as below:

a) Data Mining

In order to understand the dataset information, data mining has been used to take out the information from dataset and convert it into clear structure for further used.

b) Machine Learning

Machine learning concentrates on the advancement of PC projects that can teach themselves to develop and change when presented to new information. In machine learning, there is two type learning which is unsupervised and supervised learning. The unsupervised learning is when the examples given to the learner are unlabelled, there is no mistake or reward sign to assess a potential arrangement. Next, the supervised learning is the machine learning task of surmising a function from label training data. In supervised machine learning, there are some classification techniques such as Decision Tree Learning and Naive Beyes Classification.

2.6 Overview, Case Study and Taxonomy of P2P Botnets

In a P2P each taking part node acts both as a client and as a server ("servent") and "pays" its cooperation by giving access to some of its resources, most much of the time, processing power and/or disk space. Despite the fact that this thought is normal to all P2P systems they contrast impressively in their underlying architecture. Peer-to-peer is a class of uses that exploit resources storage, cycles, content, human presence available at the edges of the Internet. Next, a top to bottom analysis of P2P botnets is given by Wang P. et al. (2009) from University of Central Florida, USA through a detailed description of P2P botnets has been exhibited by systematically focused the study on P2P lifetime, sorts of P2P botnets and countermeasures for P2P botnets. Different looks into that give points of interest of P2P botnets is Grizzard J. et al. (2007) where the background and history of P2P botnets are discussed.

Hossein R. Z. et al. (2010) trusted that the taxonomy of botnets detection techniques need to comprehend before further action is made. They have talked about on two approaches of botnet detection techniques based on setting up honeypots and Intrusion Detection Systems (IDS) guided by a complete study that has been finished. Next, they additionally have clarified on existing P2P botnets topologies and structures. Finally, they need to admit that botnets identification is a testing issue.

The surrounding environment like software or equipment required at the time of testing for P2P network. The network traffic of the environment will be captured by Wireshark. After that, there are two kind of analysis technique, which is static and dynamic analysis (G Savan et al, 2013). The keyword of these analysis are examined in this area is Wireshark Network Analyser.

2.6.1 Wireshark Network Analysis

Wireshark Network Analyser is a famous network packet analyser since it is free open source application and it is reasonable for all working framework. It's answerable to capture the packet and display either readable data or in binary form and to inspect what's happening inside a network cable.

The features of Wireshark are show as following:

- The packet that have been saved can be read using Wireshark.
- It is a GUI platform which is more convenient and make easy to read by the user.
- Filter feature are available to filter the data that user needed.
- Captured traffic can trace Voice over Internet (VoIP) calls over the network.

2.7 Analysis Technique

Botnet analysis is the process of determining the purpose and functionality of a given botnet. This procedure is a vital step to have the capacity to create powerful identification strategies for malicious code. Next, Botnet analysis technique are separated into two, which is Static analysis and Dynamic analysis (Gadhiya, 2013). The technique will discuss more detailed below:

i. Static Analysis

In Static Analysis technique, it can be representation and applied on different of program. Next, if source code is available, static analysis tools can help to prove the models problems for a given system and will finding memory corruption flaws (Gadhiya, 2013).

Static analysis is the way toward breaking down a system's code without really executing it. In this procedure, a binary is normally dismantled in the first place, which indicates the way toward changing the binary code into relating assembler directions. At that point, both control flow and data flow analysis technique can be utilized to reach summary about the usefulness of the project.

The different types of malware can be detected using many static binary analysis technique that were have been introduce. Static analysis has the advantage that it can cover the complete program code and is usually faster than its dynamic counterpart(Bayer, Moser, Kruegel, & Kirda, 2006)

ii. Dynamic Analysis

Executing a given malware sample inside a controlled domain and checking its activities with a specific end goal to investigate the malicious behavior is called dynamic malware analysis. Along these lines it is anything but difficult to see the real behaviour of a program. Another significant preferred standpoint is that it can be robotized subsequently empowering analysis at a vast scale premise. In any case, the principle disadvantage is purported torpid code: That is, not at all like static analysis, dynamic analysis more often than not screens stand out execution way and subsequently experiences inadequate code coverage. Furthermore, malware tests may adjust their conduct or quit executing at all once they distinguish to be executed inside a controlled investigation environment (Gadhiya, 2013).

2.8 Overview and Case Study of P2P Botnets Behaviour in Skype Application.

Skype is an application that provides video talk and voice call service. Clients may trade such computerized records as pictures, content, video and any others, and may transmit both content and video messages. Skype permits the making of video telephone calls.

2.8.1 Worm or Malware that Affected the Skype Application

In this section, will be discussed about the worm that affected the skype network. The worm that will be discussed is T9000 Malware, Win32/Rodpicom, Win32/Malware!Drop, and etc.

1. T9000 Malware

The T9000 main objective is to gather data about the focused on victim which is does by compromising Skype video calling software. After that, the malware has guided into Skype, it records chat messages, video calls, and the audio calls. After that, it stores them in an index uncommonly made by the Trojan called "Intel", which the attacker can dig for the information.

2. Win32/Rodpicom

Win32/Rodpicom is a horrendous Trojan infection which can get inside your PC without telling you by means of an assortment of system assets like fake media players, spam email connections, and unknown free software, for example, Gmail application, Skype application and some more.

3. Win32/Malware!Drop

Win32.Malware!Drop can causes click misrepresentation and irritating web program which can sidetracks to the questionable sites, it might appear to be legitimate ones, at the same time, in truth, are identified with malware. After that, when a client of the PC taps on an apparently dependable connection, it quickly changes to a pernicious URL. These pernicious sites are loaded with tainted promotions that publicize a wide range of downloads that incorporate uTorrent, Firefox and Skype.

2.9 Critical Review of Current Problem and Justification

There have been significant work so as to classify the P2P Botnet Behavior in Skype Application. In this research, the P2P botnet conduct in Skype Application is methodology and there have a few journal that have been approach on this P2P botnet behavior which is Recognizing P2P Botnets Characteristic through TCP Distinctive Behavior, Revealing the Criterion on Botnet Detection Technique and the Analysis of the Skype Peer-to-Peer Internet.

Firstly, In the research of Recognizing P2P Botnet Characteristics through TCP Distinctive Behavior (Brands and Karagiannis, 2009) depicted more about the conduct of the P2P Botnet through TCP Distinctive Behavior which is on the Transmission Control Protocol (TCP) is in charge of exchanging information starting with one system to another. The fundamental capacity of TCP is separating the information into pieces and marks them with arrangement numbers for appropriate information conveyance on a system. As indicated by (Abdullah and Ud, 2011), there are 8 flags; FIN, SYN, RST, PSH, ACK, URG, ECE, CWR in TCP banner. Fundamentally, these flags have decimal numbers and will be describe as Table 2.1.

Table 2. 1: TCP Flag and Control Section

TCP Flags Bit	Control Sections	Corresponding Decimal	Description
8	CWR	128	Indicate that the congestion window has been reduced
7	ECE	64	Indicate that a CE notification was received
6	URG	32	Indicates that urgent pointer is valid that often caused by an interrupt
5	ACK	16	Indicates the value in acknowledgement is valid
4	PSH	8	Tells the receiver to pass on the data as soon as possible
3	RST	4	Immediately end a TCP connection
2	SYN	2	Initiate a TCP connection
1	FIN	1	Gracefully end a TCP connection

In line with that, (Ezzeldin H, 2010) has covered out the TCP Flag combination that probably performs to attack the network by an illegal attacker. A list of TCP Flag combination parameters that needs to give attention are:

- a) TCP SYN (Half Open) Scan (tcp.flags==2)
- b) TCP SYN/ACK Scan (tcp.flags==18)
- c) TCP FIN Scan (tcp.flags==1)
- d) TCP XMAS Scan (tcp.flags==41)
- e) TCP NULL Scan (tcp.flags==0)

Second, In the research of Revealing the Criterion on Botnet Detection Technique, (Abdullah & Abdollah, 2013) described more about Botnet detection technique is the technique used to detect or identify the Botnet activities. There are the characteristics of each techniques which is Anomaly-based detection, Signature-based detection, and Hybrid-based detection. The Anomaly-based detection technique is a part of behavior based detection. Next, the anomaly-based is divided into DNS-based, data mining-based, host-based and network-based. This techniques attempt to detect Botnet based on several network traffic anomalies such as high network latency, high volumes of traffic, traffic on unusual ports and unusual system behavior that could indicate presence of malicious bots in the network. After that, the Signature-based detection is similarly to anomaly-based techniques, which is it also as a part of behavior-based detection. This techniques learn and gain knowledge of useful signatures or behaviors from existing Botnet. Next, in the hybrid-based detection technique, two or more IDS techniques were combined. It can be the combination of DNS-based with anomaly-based, signature-based with anomaly-based or data mining-based with anomaly-based technique. Due to signature-based, DNS-based and data mining-based that have same capability where it is only able to detect known attack but cannot detect unknown attack. Instead, anomaly-based has this extra capabilities to detect unknown attack compare to other technique.

In conclusion, in the study of Analysis of the Skype Peer-to-Peer Internet (Baset and Schulzrinne, 2006), Skype is a peer-to-peer (p2p) VoIP customer created by the organization that made Kazaa. Skype permits its clients to place voice calls and send instant messages to different clients of Skype customers. Fundamentally, it is very much like the MSN and Yahoo IM applications, as it has abilities for voice-calls,

texting, sound conferencing, and buddy lists. Nonetheless, the basic conventions and systems it utilizes are very different (Abdullah and Ud, 2011). In addition, In this examination, the convention of skype is depicted which is a Skype customer (SC) opens a TCP and a UDP listening port at the port number arranged in its association dialog box. SC randomly picks the port number upon establishment.

2.10 Proposed Solution

Based on journal above, I have propose to use the TCP Flags to classify the behavior of the P2P botnet in Skype Application. For the analysis part, I will use the Wireshark Network Analysis to detect the abnormal of TCP flags in the data that I run. In this research, I got the malware data in .exe format, which I found it in the Internet.

The malware data is being run in the Wireshark and the TCP Flags will be filter to detect the abnormal behavior of the P2P botnet in the Skype Application. Next, after that, I will proceed to the analysis result and compare the normal TCP Flags and the abnormal TCP Flags that have been found in the Wireshark.

2.11 Conclusion

This chapter has provided a detailed literature review of this project. This is to make sure that it enable the research to be implemented in a proper manner. Next, this chapter consist of Critical Review of Current Problem and Justification, which is the comparison of three journal which can help in the analysis result. After that, the Proposed Solution in this chapter also will be helpful in the analysis and implementation of this project.

CHAPTER III

METHODOLOGY

3.1 Introduction

This chapter will portray the research methodology, model and stream of ventures that are utilized to play out this investigation of P2P Botnet in Skype. From past part, the related work of the examination had been talked about. After that, in this part, the breakthrough was itemized portrayed, which is the development contains week, date and action for observing and estimation of task execution. This point of reference go about as advancement rule for this anticipate.

3.2 Methodology

This chapter shows the issues identified with the general undertaking philosophy. This anticipate strategy is created with the goal that undertaking can be led in a right arrangement, oversee and keep up the advancement. The best approach ought to be utilized as a part of this examination is the System Development Life Cycle (SDLC) strategy. The SDLC model infers its name because of the falling impact one stage to another. SDLC comprises five fundamental stages which is Planning, Analysis, Design, Implementation, and Maintenance. This model was picked in light of the fact that it is truly reasonable for this anticipate. In this model every procedure should be finished first before proceed to another procedure. It ought to be done orderly and starting with one process then onto the next procedure so that everything will run easily lastly all the outcome gathered on time.

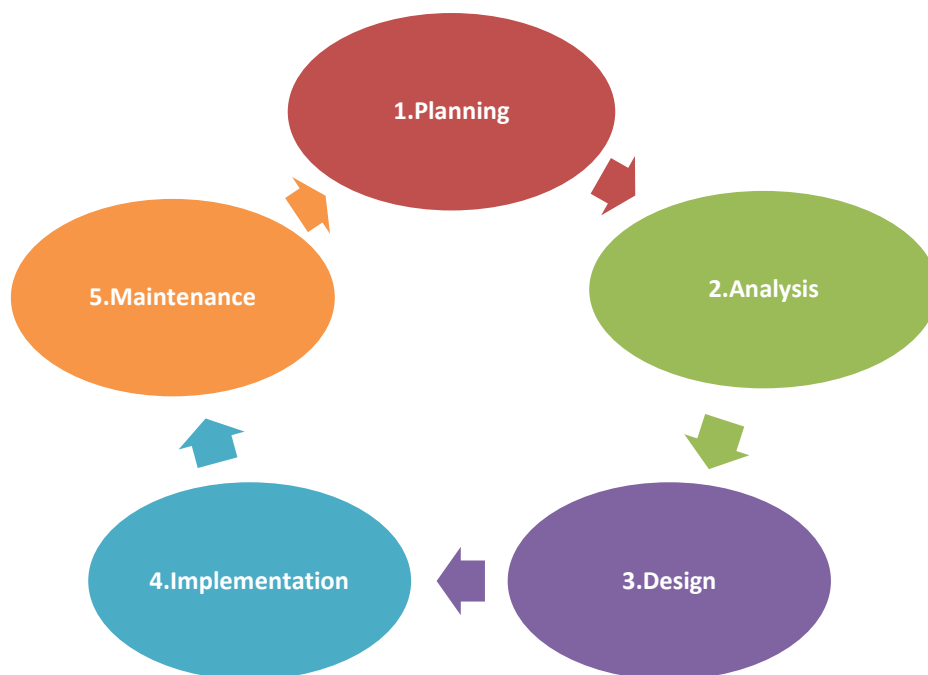


Figure 3. 1: System Development Life Cycle (SDLC) Model

i. Planning phase

In this stage, preparatory study and writing survey on the exploration issue was point by point depicted which is including the presentation of the examination, provides justification with an overview, furthermore foundation data of this examination, issue articulation, objective, and scope.

ii. Analysis Phase

Analysis is the part of gathering information and spotlight on task flowchart, setup of P2P system, P2P information movement caught in Skype utilizing Wireshark. After that, the final item and what is the necessity for this framework will be resolved and recorded. It incorporates what is the desire for the framework, required capacities, interfacing and how it will perform.

iii. Design Phase

A general system design is continuing subsequent to having a decent comprehension about the framework. The portrayal of the configuration is doing with a pen and a bit of paper to decide how the examination of the P2P in Skype will look like and how it will be tried and caught utilizing programming that have been chosen which is; Wireshark. After that, the arranging of the design documentation procedure ought to be begun.

iv. Implementation Phase

Implementation Phase is about the identification and examination of P2P Botnet in Skype and the testing method. This stage comes after a complete understanding of examination necessities and points of interest, it's the genuine improvement process consequent to having a complete and demonstrated blueprint for the proposed examination of the data got. The execution stage may take a long time and that depends on upon the many-sided nature of the structure it presents. Planning may be required to get to know this system and know how to use it.

v. Maintenance Phase

In this phase, periodic maintenance for the system will be completed to ensure that the framework won't get to be old, this will incorporate supplanting the old equipment and constantly assessing framework's execution, and it additionally incorporates giving most recent upgrades to specific segments to ensure it meets the right models and the most recent advances to face current security dangers.

3.3 Project Flow

P2P Botnet Analysis Framework will be used in this project for investigation P2P botnet behaviour in Skype. This framework includes five main phases which is data set, pre-processing, statistical approach, analysis and result.



Figure 3. 2:P2P Botnet in Skype Analysis Framework

3.4 Project Tool and Requirement

There are some of project tools and project requirements are involved in this project.

The software requirements that have been used are:

Table 3. 1: Software requirement

Item	Specification
Operating System	<ul style="list-style-type: none"> • Microsoft Windows Server 2008
Software's	<ul style="list-style-type: none"> • Wireshark • Microsoft Office Project 2010 • Microsoft Office Word 2007 • Microsoft Visio 2007 • RapidMiner Studio version 7.2 • TCP Trace • Microsoft Excel Comma Separated Values File (.Csv)

Table 3. 2: Hardware requirement

Item	Specification
UTEM Computer	<ul style="list-style-type: none"> • 8 GB RAM • Intel CPU • Dual-core • Intel Core 2 Duo 2.50 GHz • Genuine Windows 7 Professional 64-bit 19
Personal Computer (PC)	<ul style="list-style-type: none"> • Processor Intel(R) Xeon® E5506 2.13GHz • Memory 8 GB • Hard Drive 500 GB • 64-bit (GNOME-Gallium 0.4)

3.5 Project Schedule and Milestone

In this project, the milestone and Gantt chart have been design as one of the most useful ways of showing activities displayed against time. This Milestone will show the beginning of the date and the ends of the project time. Each of activity is scheduled to the last date of project. This Milestone is design according to weeks which is from the first week until last week for the final representation. Other than that, each chapter will go through the phase that has been described in methodology. The Gantt chart was put on Appendix section (Appendix 1).

Table 3. 3: Milestone

Week	Activity
1 (22-26 Feb)	Proposal PSM: Submission and Presentation

Week	Activity
	Proposal Assessment and Verification
2 (29 Feb-4 Mar)	Proposal Correction/Improvement Chapter 1
	List of Supervisor/Title
3 (7-11 Mar)	Chapter 1 (System Development Begins)
4 (14-18 Mar)	Chapter 1 & Chapter 2
5 (21-25 Mar)	Chapter 2
6 (28 Mar-1 April)	Chapter 2 & Chapter 3
	Student Status
7 (4-8 April)	Project Demo & Chapter 3 Chapter 4
9	Project Demo & Chapter 4

Week	Activity
(18-22 April)	
10 (25-29 April)	Project Demo & Chapter 4
11 (2-6 May)	Project Demo
12 (9-13 May)	Project Demo & PSM Report
13 (16-20 May)	Project Demo & PSM Report Presentation Schedule
14 (23-27 May)	Project Demo & PSM Report
15 (30 May-3 June)	Final Presentation (PA)

3.6 Conclusion

Finally, the project methodology is already done in this chapter. Project planning involves planning the resources such as time, cost and manpower. This is a very important part of developing a project as it can determine the overall success of the project. Justifications must be made on the most suitable hardware and software. Lastly, to achieve the goals of this project, the plan for the whole PSM 1 and PSM 2 must be followed.



CHAPTER IV

DESIGN

4.1 Introduction

In the past chapter, the project methodology is all around clarified. Next in this Implementation and Analysis chapter which is one of the most imperative periods of a project is to distinguish some essential criteria of a project. After that, it likewise a critical part to assemble the data that required for comparison, analysis, and other project requirements. The important tools and the procedures of the investigation will be talked about with the real samples. Besides, in this Implementation and Analysis phase are additionally useful for the data collection which will be valuable to set parameters in the following up and coming stage.

4.2 P2P Botnet in Skype Analysis Approach

One of the main tools that is utilized for the analysis is the Wireshark Network Protocol Analyser. Wireshark network protocol analyser is a free and open source packet analyser and it is utilized for network troubleshooting, analysis, software and communications protocol development, and education. After that, the Wireshark is easy to use and free.



Figure 4. 1: Wireshark Network Protocol Analysers

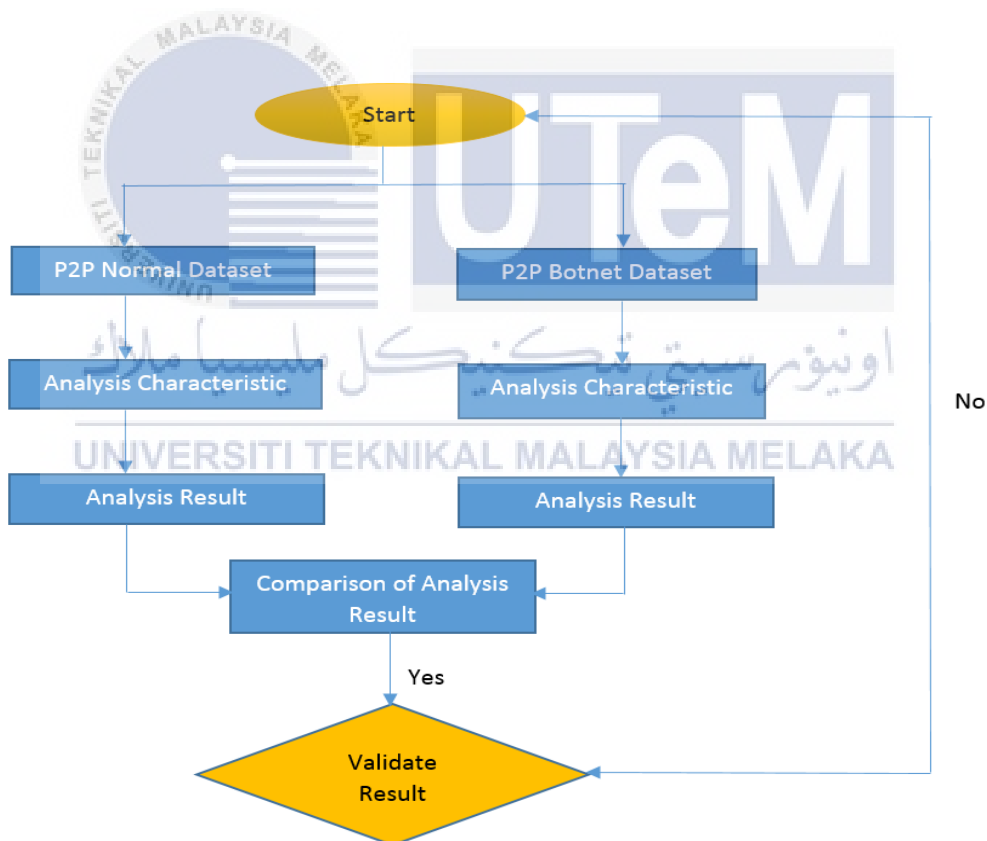


Figure 4. 2: The flow of Analysis of P2P Botnet in Skype Application

Based on the figure 4.2.2 will described the flowchart of the analysis of P2P Botnet in Skype application. On the first process is selecting the data set which is P2P normal and P2P botnet data set. This project is utilizing network traffic that is adapted by

Raihana. The implementation of the Raihana's dataset will examine on the dataset implementation. Next, the normal traffic for TCP in application were analysed through Wireshark analysis.

4.3 Implementation of dataset

The dataset was used to get the analysis result and performance detection result. Next, the approach below tell how the dataset is being captured.

4.3.1 P2P Botnet Detection Approach

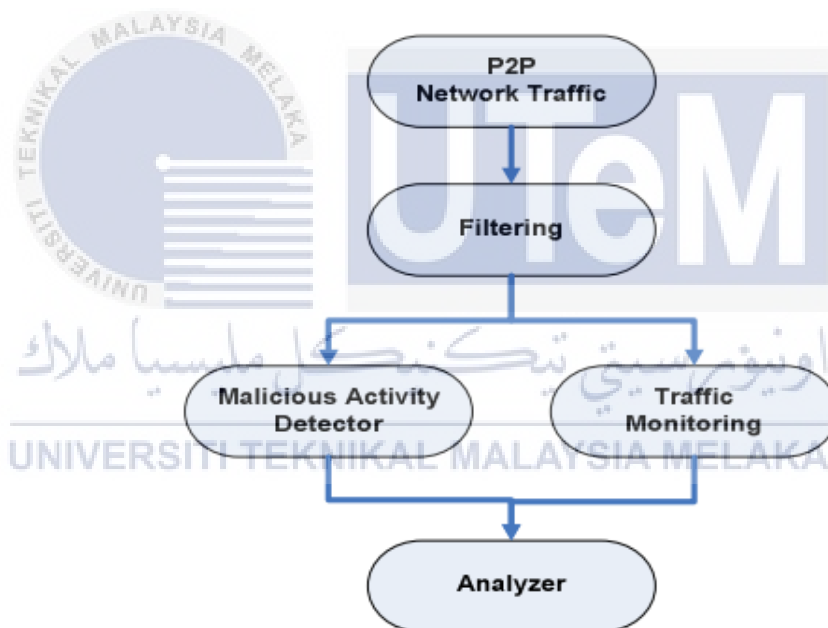


Figure 4. 3:P2P Botnet Detection Framework (Raihana Syahirah, 2011)

4.4 Data Collection

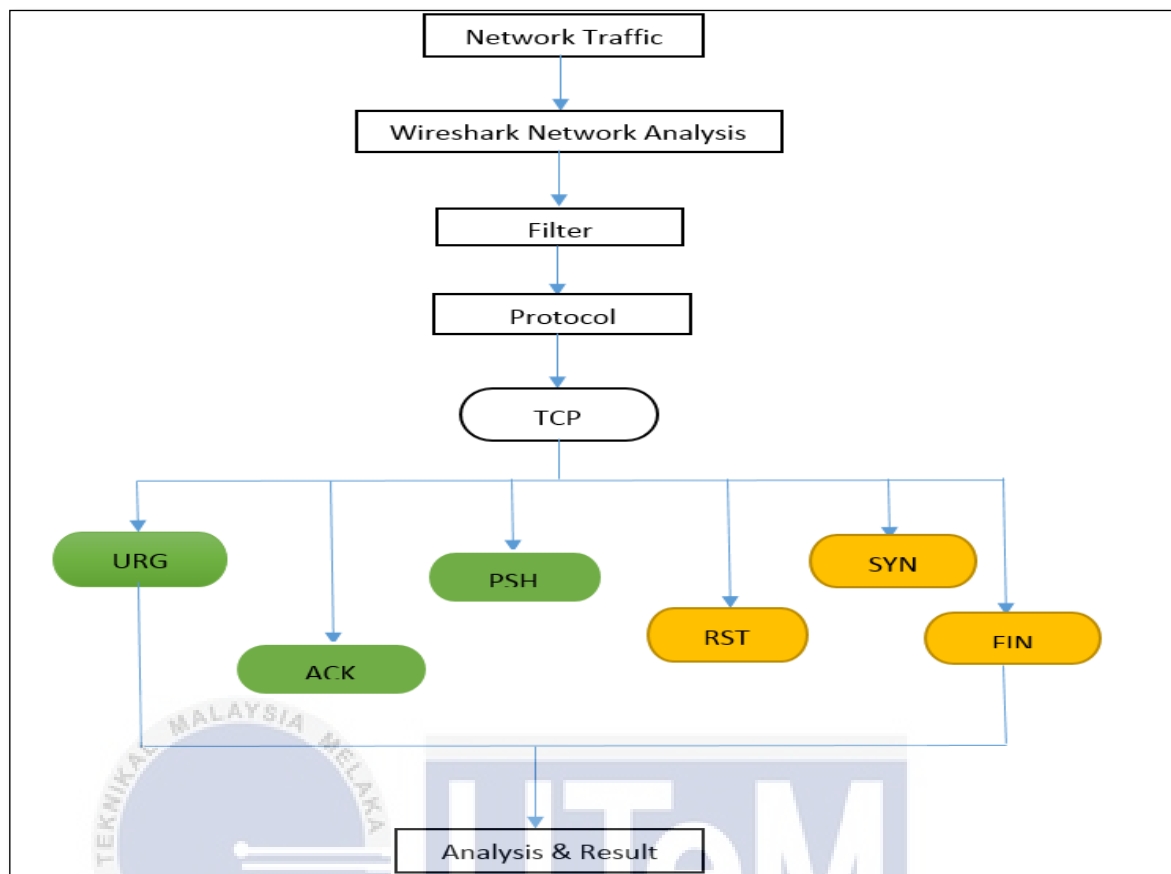


Figure 4. 4: Flowchart of each stage in the process

Based on the figure 4.4.1 that examining in right on time of this chapter, the more detail of the flowchart for analysis in this section. This project is utilizing network traffic that is adapted by Raihana Syahirah A. The TCP protocol in network traffic were filter through Wireshark analysis. In this project, the SYN, FIN and RST of the TCP will chose for this analysis which in light of the fact that there are numerous normal for the TCP that can be analyse. After that, the data is assess by filter the Flags of the TCP Protocol and continue to the analysis part.

4.5 Analysis Result

In this section, the analysis result will be cover up on P2P normal and P2P botnets traffic or abnormal traffic where the analysis is done in term of detection on the attributes of P2P botnets in Skype Application. Firstly, to demonstrate the normal for P2P botnets in Skype Application, the running skype must be filter in the Wireshark. Next, the normal for P2P botnets in Skype Application was demonstrates from the protocols that have been examined which is TCP. Next, each of the analysis results of recognizing the characteristics that will be clarified in narrative manner that includes flow diagram analysis, snapshot captured packet for comparison of P2P normal and P2P botnets characteristics and the justification on how the indication of trademark can be recognize. The details of analysis result will be enlightened on the following sub topic below:

4.5.1 Skype Application

The connection of Skype Application must be demonstrated utilizing Wireshark. The Skype Application prefers User Datagram Protocol (UDP) for voice transmissions. Next, Ports 80 and 443 are the requirements for Skype and be open for outgoing Transmission Control Protocol (TCP) transmissions. Moreover, Ports 5060 and 8000 ought to stay open for incoming and outgoing UDP transmissions.

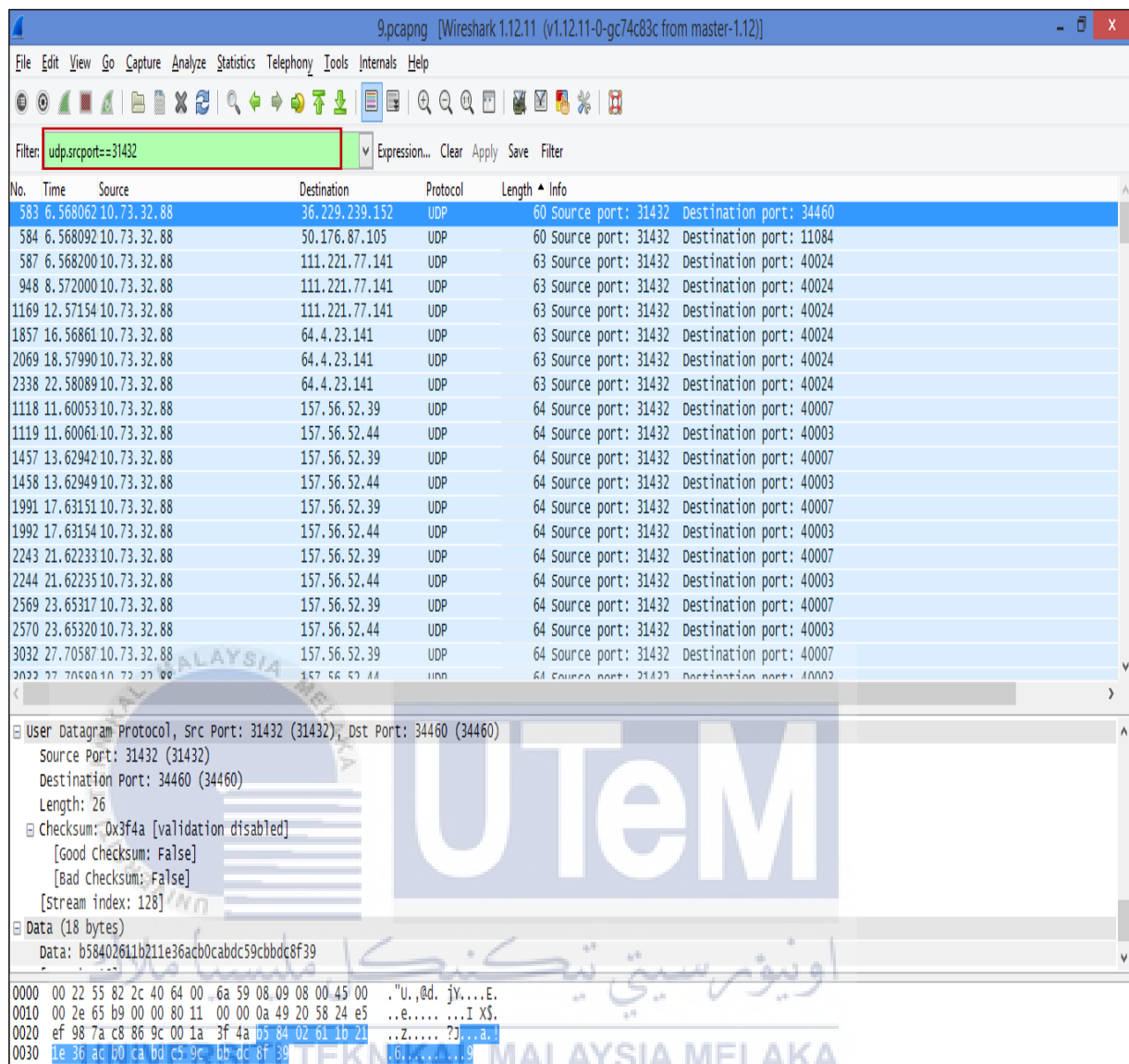


Figure 4. 5: The Skype is running in the Wireshark

From figure 4.6 above demonstrated the Skype is running in the Wireshark while being connected with another Skype. After that, to realize that the Skype is running in the Wireshark, the protocol of Skype is utilized to be filter in the Wireshark, which is filter using `udp.srcport==31432`.

4.5.2 TCP

TCP was characterized in RFC 793 (Postel J., 1981) is a transport protocol that is in charge of data transferring which is starting with one system then onto the next and it divides the data into pieces and marks them with grouping numbers for appropriate request upon delivery. All TCP connections are set up utilizing a three way handshake SYN, SYN/ACK and lastly ACK.

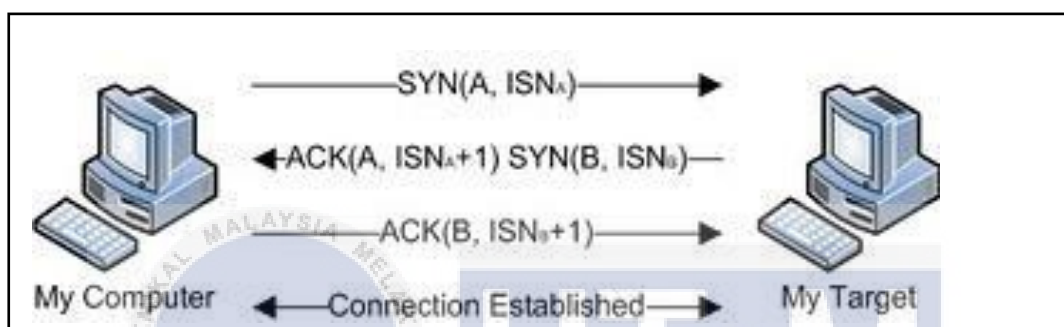


Figure 4. 6: The TCP States

The screenshot shows a Wireshark capture of a TCP connection. The filter is set to `tcp and ip.src==10.73.32.88`. The packet list pane shows the following key packets:

No.	Time	Source	Destination	Protocol	Length	Info
1914	16.7710	10.73.32.88	125.56.199.90	TCP	54	59460-443 [ACK] Seq=951 Ack=28405 win=262144 Len=0
1919	16.7810	10.73.32.88	103.243.222.32	TCP	54	59459-443 [ACK] Seq=110 Ack=2852 win=66048 Len=0
1920	16.7810	10.73.32.88	31.9.48.7	TCP	66	[TCP spurious retransmission] 59462-4443 [SYN] Seq=0 win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1922	16.7810	10.73.32.88	103.243.222.32	TLSv1	364	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
1926	16.8110	10.73.32.88	202.58.9.201	TCP	54	59457-443 [ACK] Seq=1533 Ack=4348 win=64860 Len=0
1930	16.8110	10.73.32.88	202.58.9.201	TCP	54	59457-443 [ACK] Seq=1533 Ack=3263 win=63943 Len=0
1933	16.8710	10.73.32.88	103.243.222.53	TCP	54	59461-443 [ACK] Seq=362 Ack=106 win=261888 Len=0
1934	16.8710	10.73.32.88	103.243.222.53	TLSv1.2	105	change Cipher Spec, Encrypted Handshake Message
1935	16.8710	10.73.32.88	103.243.222.53	TLSv1.2	2550	Application Data
1947	17.0710	10.73.32.88	103.243.222.32	TCP	1514	[TCP segment of a reassembled PDU]
1952	17.1610	10.73.32.88	103.243.222.32	TLSv1	615	Application Data
1954	17.1610	10.73.32.88	103.243.222.32	TCP	1514	[TCP segment of a reassembled PDU]
1958	17.2410	10.73.32.88	103.243.222.53	TCP	54	59461-443 [ACK] Seq=2909 Ack=822 win=261120 Len=0
1960	17.2710	10.73.32.88	103.243.222.53	TLSv1.2	1140	Application Data
1962	17.2510	10.73.32.88	31.9.48.7	TCP	62	[TCP spurious retransmission] 59462-4443 [SYN] Seq=0 win=65535 Len=0 MSS=1460 SACK_PERM=1
1970	17.4310	10.73.32.88	103.243.222.32	TLSv1	72	Application Data
1971	17.4510	10.73.32.88	31.9.48.7	TCP	66	59463-4443 [SYN] Seq=0 win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
1975	17.5410	10.73.32.88	103.243.222.32	TCP	54	59458-443 [ACK] Seq=981 Ack=3752 win=65280 Len=0
1978	17.5610	10.73.32.88	103.243.222.32	TLSv1	228	Application Data
1979	17.5710	10.73.32.88	111.221.05.12	TLSv1	70	Application Data

The packet details pane for packet 1926 shows:

- Frame 1926: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
- Ethernet II, Src: Dell_59:08:09 (64:00:6a:59:08:09), Dst: ciscoInc_82:2c:40 (00:22:55:82:2c:40)
- Internet Protocol Version 4, Src: 10.73.32.88 (10.73.32.88), Dst: 202.58.9.201 (202.58.9.201)
- Transmission Control Protocol, Src Port: 59457 (59457), Dst Port: 443 (443), Seq: 1533, Ack: 4348, Len: 0

Figure 4. 7: Normal TCP connection setup in Wireshark view

4.5.3 Characteristic Analysis

According to Clarke G. E. (2011), they are brought up that there are 8 flags; FIN, SYN, RST, PSH, ACK, URG, ECE, CWR in TCP. Fundamentally, these flags have decimal numbers and description as well assigned to them as Table 4.1 below:

Table 4. 1: Relationship between TCP Flag & Control Section

TCP Flags Bit	Control Sections	Corresponding Decimal	Description
8	CWR	128	Indicate that the congestion window has been reduced
7	ECE	64	Indicate that a CE notification was received
6	URG	32	Indicates that urgent pointer is valid that often caused by an interrupt
5	ACK	16	Indicates the value in acknowledgement is valid
4	PSH	8	Tells the receiver to pass on the data as soon as possible
3	RST	4	Immediately end a TCP connection
2	SYN	2	Initiate a TCP connection
1	FIN	1	Gracefully end a TCP connection

```

... 0000 0000 0010 = Flags: 0x002 (syn)
000. .... = Reserved: Not set
...0 .... = Nonce: Not set
...0... .... = Congestion Window Reduced (CWR): Not set
... .0. .... = ECN-Echo: Not set
... .0. .... = Urgent: Not set
... ..0 .... = Acknowledgment: Not set
... ..0... = Push: Not set
... ..0. .... = Reset: Not set
... ..1. = Syn: Set

```

Figure 4. 8: TCP Flag at packet details view in Wireshark

This study will be discussed about the TCP Flag combination which is likely attack the network traffic by an attacker. Below are the TCP Flag combination that will be analysed in this study:

i. TCP SYN Half Open ($\text{tcp.flags} = 2$)

Half open TCP or TCP SYN Scan is a tiny bit stealthier than the past scan since it utilizes a different method. The attacker sends a SYN to the targets, if the target's port is open and it reacted with a SYN/ACK, then the attacker will promptly tear down the association utilizing the RST flag. So this kind of traffic might appear to be normal but we have to notice on the number of the Half Open connection. If the SYN packets are greater than the SYN/ACK packets, then there is something wrong which is the scanning process was happen by attacker.

ii. TCP SYN/ACK Scan ($\text{tcp.flags} = 18$)

Flags is use to check if a TCP Connect Scan happened on this network. Furthermore If the SYN packets are greater than the SYN/ACK packets, then there is something incorrectly which is the filtering process was happen by attacker.

iii. TCP FIN Scan ($\text{tcp.flags} = 1$)

The FIN Scan breaks the rule of TCP connection establishment since it sends an unexpected packet toward begin of the connection, which is the FIN flag. Implies that, the FIN Scan begins with a FIN packet.

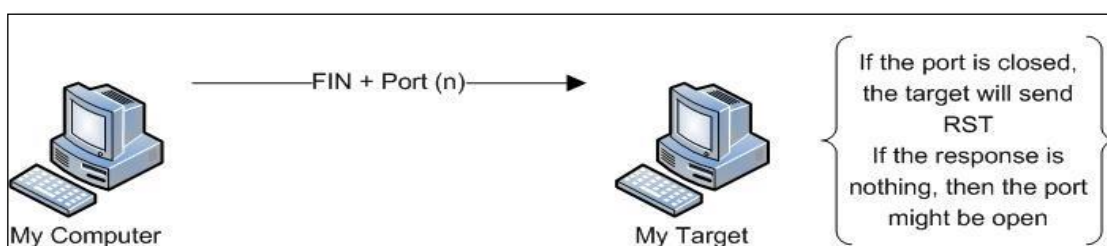


Figure 4. 9: TCP Fin Scan (Ezzeldin H., 2010)

FIN flags are a piece of any communication between two hosts since this communication must be finished at a minute. Nonetheless, if have an explosive number of FIN flagged packets without a past established connection, then deal with that. The reason is to confuse the targets on the grounds that every targets expects a SYN packet for connection establishment.

iv. TCP NULL Scan ($\text{tcp.flags} = = 0$)

The NULL Scan breaks the guideline of TCP association foundation since it sends an unexpected packets toward the starting of the connection, by all flags from the packets. The NULL Scan begins with a packet that has every one of the Flags set to 0. We ought to never at any point see a NULL packet on system for any reason since it is unlawful to have a bundle without any flags set.

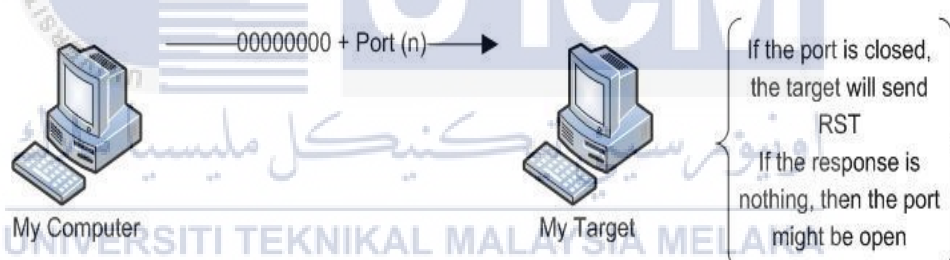


Figure 4. 10: TCP NULL Scan (Ezzeldin H., 2010)

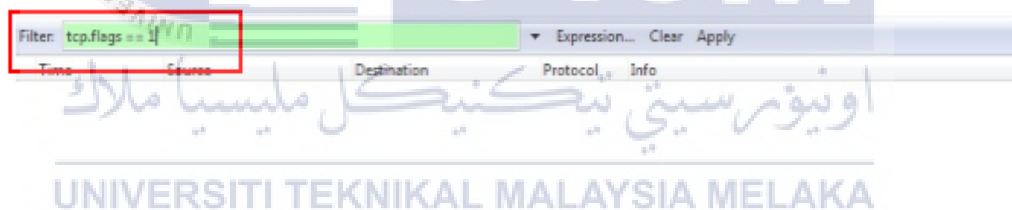
4.6 Comparison of P2P Normal and P2P Botnets

The characteristics have been analysed to distinguish between P2P normal and P2P botnets in the Skype Application which is focus on TCP flags combination are shown below:

i. TCP Flag Characteristic: TCP FIN Scan ($\text{tcp.flags} = = 1$)

Table 4. 2: TCP FIN Scan Comparison

Comparison on TCP FIN Scan (tcp.flags == 1)	
P2P Normal	P2P Botnets
In P2P normal, there is not have any TCP FIN Scan (tcp.flags==1). The P2P normal are captured as Figure 4.13 in Wireshark.	In P2P botnet, it found the TCP FIN Scan (tcp.flags == 1). The TCP FIN Scan will make the target confuse which is the target expectations of a SYN packet for the connection establishment. Next, the time for the attack in TCP FIN Scan is each about 2 seconds.

**Figure 4. 11: TCP Fin Scan in P2P normal**

ii. **TCP Flag Characteristic: TCP FIN Scan (tcp.flags == 2)**

Table 4. 3: TCP SYN Scan or known as Half Open (tcp.flags == 2)

Comparison on TCP SYN Scan (tcp.flags == 2)	
P2P Normal	P2P Botnets

<p>In P2P Normal, does not have TCP SYN Scan or known as Half Open (tcp.flags = = 2).</p>	<p>Have a TCP SYN Scan or known as Half Open (tcp.flags = 2) which is half open is a tiny bit stealthier than the past sweep since it utilizes an alternate strategy. The attacker sends a SYN to the targets, and if the target's port is open and it reacted with a SYN/ACK, then the attacker will instantly tear down the association utilizing the RST flag. Next, the Real captured packet in P2P botnets as Figure 4.1.4 in Wireshark view.</p>
---	--

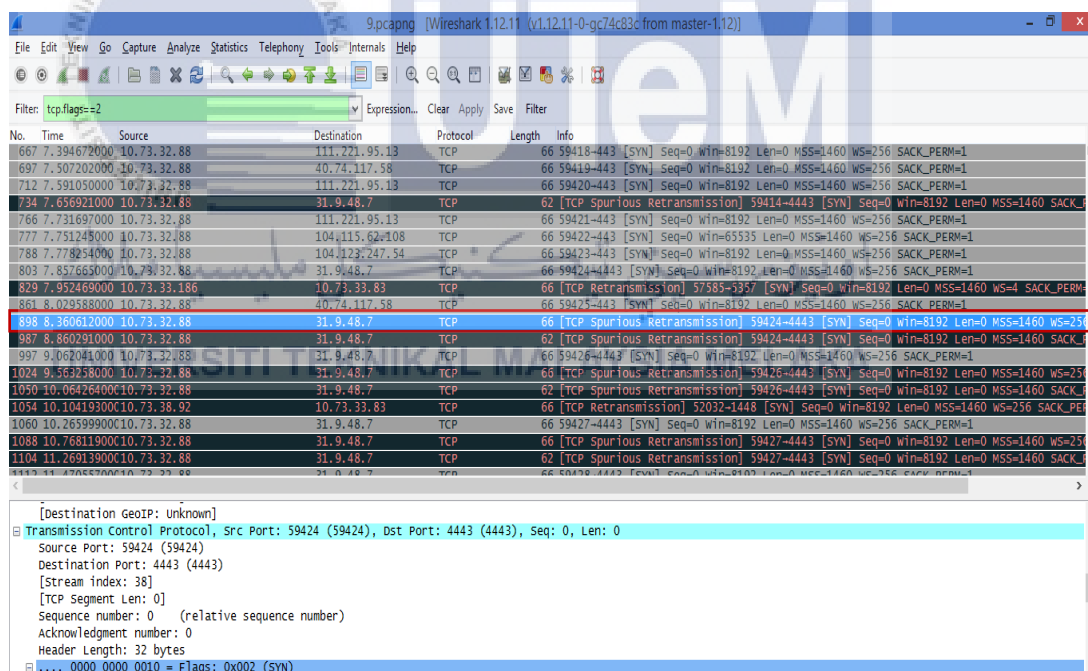


Figure 4. 12: Shown the abnormal traffic (tcp.flags = = 2).

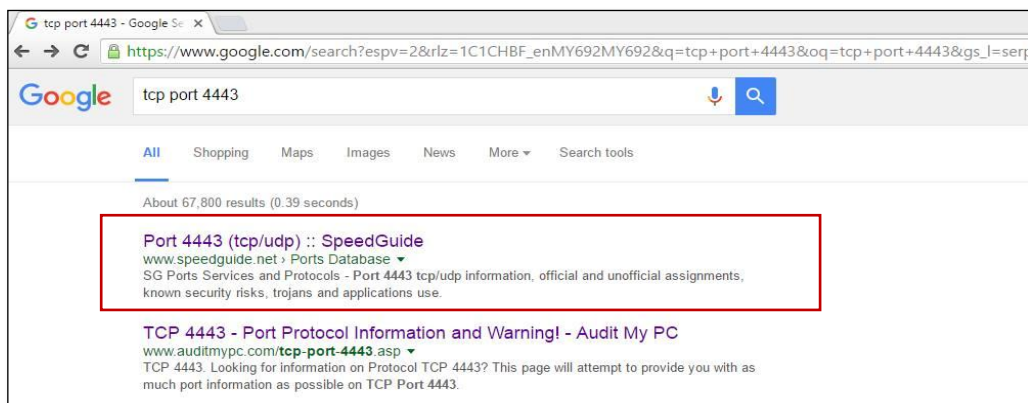


Figure 4. 13: Shown the abnormal port number found in Wireshark.

The abnormal port number (Port: 4443) found in the Wireshark are proved that the Skype Application have been attacked by the malware.



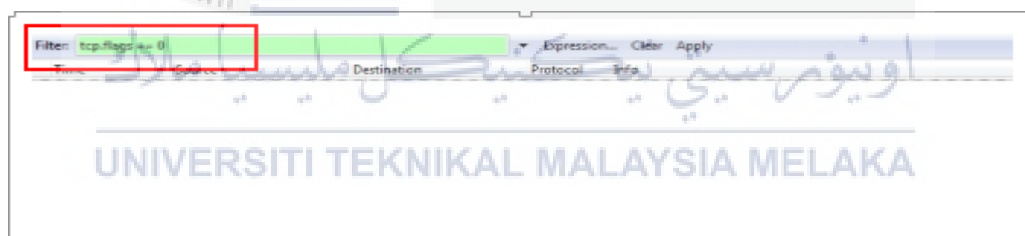
Figure 4. 14: Shown the abnormal IP address which found in the Wireshark

The abnormal IP address (31.9.8.7) found in the Wireshark are proved that the Skype Application have been attacked by the malware.

iii. TCP Flag Characteristic: TCP NULL Scan (tcp.flags = 0)

Table 4. 4: TCP NULL Scan Comparison

Comparison on TCP NULL Scan	
P2P Normal	P2P Botnets
<p>Does not have TCP NULL Scan</p> <p>Real captured packet in P2P normal as Figure 4.1.7 in Wireshark view.</p>	<p>Have a TCP NULL Scan.</p> <p>We should never ever see an NULL packet on a normal network for any reason because it is illegal to have a packet with no flags set. As usual, the attacker will do scanning process and network mapping before attack.</p> <p>Real captured packet in P2P normal as Figure 4.1.8 in Wireshark view.</p>

**Figure 4. 15: TCP NULL Scan in P2P Normal**

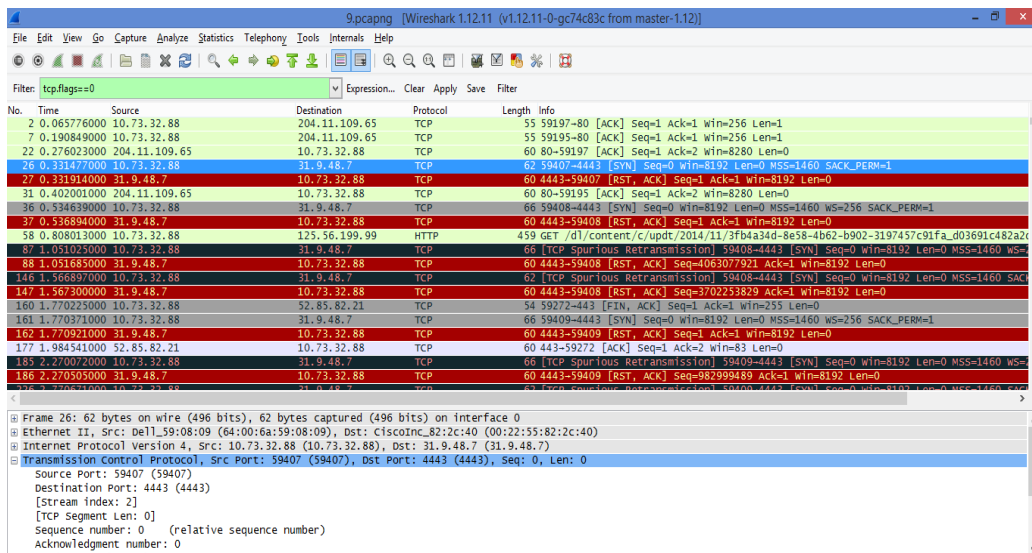


Figure 4. 16: TCP NULL Scan in P2P Abnormal

4.7 Conclusion

As for the conclusion, the Characteristic of Analysis which is the TCP Flags is the major part for the result of Analysis of P2P botnet in the Skype Application. The abnormal TCP Flags have been proved found in the Wireshark and its mean the data set that have been run is effected the Skype and it make the Skype Application being affected by the malware.

After that, for the next chapter will discuss about the implementation involved in the research. Next, it will be conduct based on the design from the analysis phase. After that, the testing phase will be discuss about the process that will done to the dataset which is Data Mining.

CHAPTER V

IMPLEMENTATION

5.1 Introduction

In this chapter, the implementation part is using data mining technique and the dataset used is continued data from the previous chapter. After that, all parameters must be apply to the data and the data must be clean to avoid noise and to get better accuracy. The parameters applied is important to have a good results. In this implementation chapter, the assumptions must be implement to get the good results. Next, when this stage is done, each one of the parameters and assumptions will be used for testing to get the results.

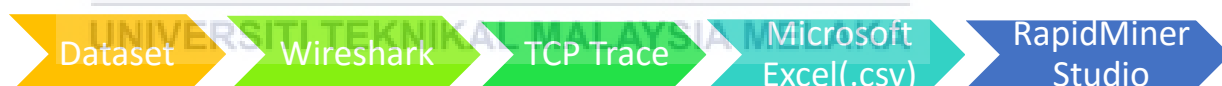


Figure 5.1: The Outline Diagram for Implementation Phase

5.2 Environment Setup

This project will run the dataset in the Wireshark and the dataset will continued to use in data mining techniques classification to get the accuracy results. The RapidMiner Studio version 7.2 is using to get the results for the dataset.

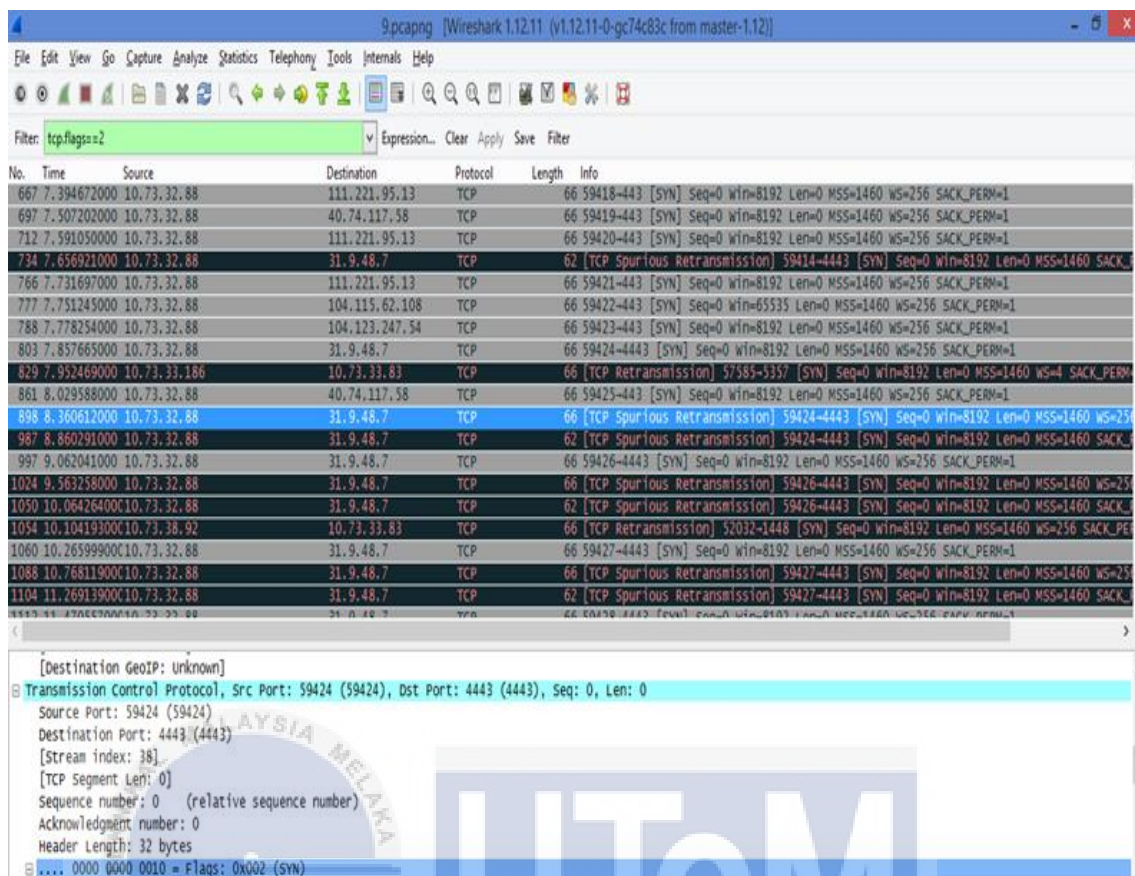


Figure 5.2: The malicious traffic that have been run in Wireshark

The Wireshark Network Analyser is a free and open source software which is used to run the dataset to get the detection results. After that, when the detection results of dataset that have been filtered was successful achieved, the dataset was continued being used in the next stage; to convert the dataset to Comma Spread Values(.csv) format using Tcp trace. Next, the malicious traffic must be combined with the benign traffic before converting the dataset to the Comma Spread Values (.csv) format. The benign traffic have been collected by connecting the legitimate Skype user with the other skype user. The communication activities consists of video calls, voice calls and chat that been captured it traffic using Wireshark.

	port_a	port_b	interdepa	total_pack	total_packs	resets_se	resets_s	ack_pkts	ack_pkts	pure_acks	pure_acks	sack_pkts	dsack_pkt	max_sack	u0ique_b	u0ique_b	actual_da	actual_da	actual_da
2	80	54772	-1.16E-05	4	2	0	0	4	2	2	1	0	0	0	667	0	1	0	667
3	54771	443	0.001227	7	5	1	0	7	5	3	1	0	0	0	467	498	2	3	467
4	54775	4443	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0
5	54776	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
6	54777	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
7	54767	80	0.000324	3	0	1	0	3	0	0	0	0	0	0	397	0	2	0	794
8	54778	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
9	443	51261	0.001076	4	15	0	1	4	15	1	3	0	0	0	335	271	3	10	335
10	54779	443	0.003565	26	21	0	1	25	21	12	5	2	0	1	2132	5420	13	14	2132
11	54780	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
12	54781	4443	2.31E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
13	54782	443	0	13	9	0	0	12	9	7	2	3	0	1	2205	5604	4	6	2205
14	54783	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
15	54762	443	1.16E-05	5	4	0	0	5	4	2	2	0	0	0	1498	506	2	2	1498
16	50961	443	0.013056	200	203	0	0	200	203	125	75	0	0	0	4441	1191	75	128	4441
17	54784	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
18	54785	4443	1.16E-05	3	3	0	3	0	3	0	0	0	0	0	0	0	0	0	0
19	54749	443	0	2	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0
20	54733	443	0	2	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0
21	54728	443	0.000498	7	0	1	0	7	0	0	0	0	0	0	0	0	0	0	0
22	54719	443	0.000498	7	0	1	0	7	0	0	0	0	0	0	0	0	0	0	0
23	54718	443	0	2	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0

Figure 5.3: Dataset in Comma Separated Values File (.csv) in Microsoft Excel

The dataset must be combined which is the malicious dataset traffic and the benign dataset traffic. After that, the combined data is labelled by 1 which is malicious and 0 is benign data. The labelled dataset is first must be cleaning to get the good results and to avoid noise. Next, the clean dataset are proceed to the next stage which is the dataset is processed based on parameters to get the accurate results with RapidMiner Studio.

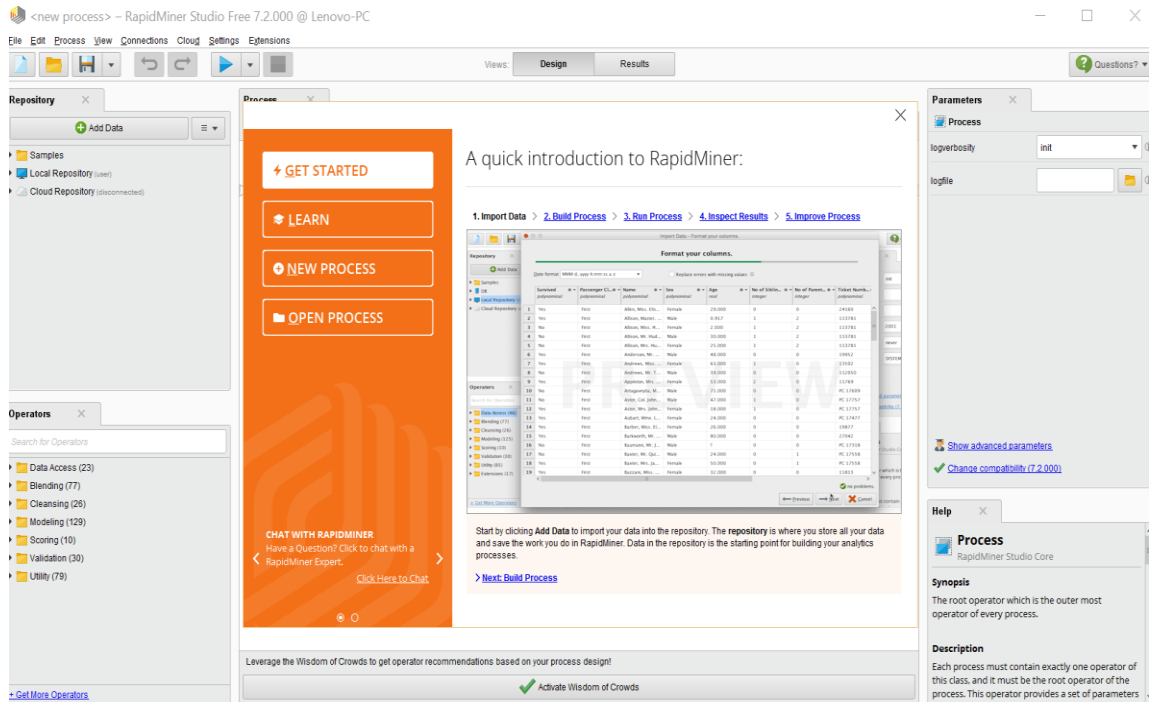


Figure 5.4: The interface of RapidMiner Studio version 7.2

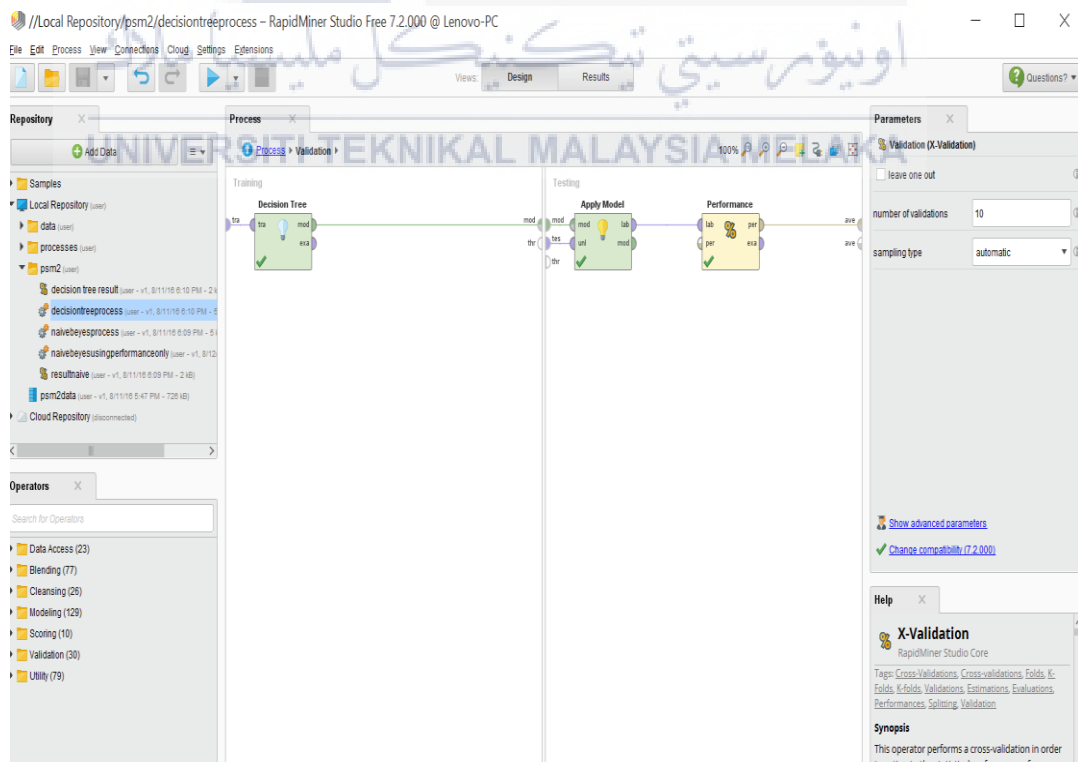


Figure 5.5: Design Of the Decision Tree process

Decision Tree Process gives an advantage which is easier to explain and this process are much meaningful compared to other process. Next, the goal of the decision tree process is to create a classification model that can predicts the estimation of an objective attribute (regularly called class or label) which taking input of a few information characteristics of the ExampleSet. After that, the Decision Tree process must be connected to the Apply Model and Performance to get the results.

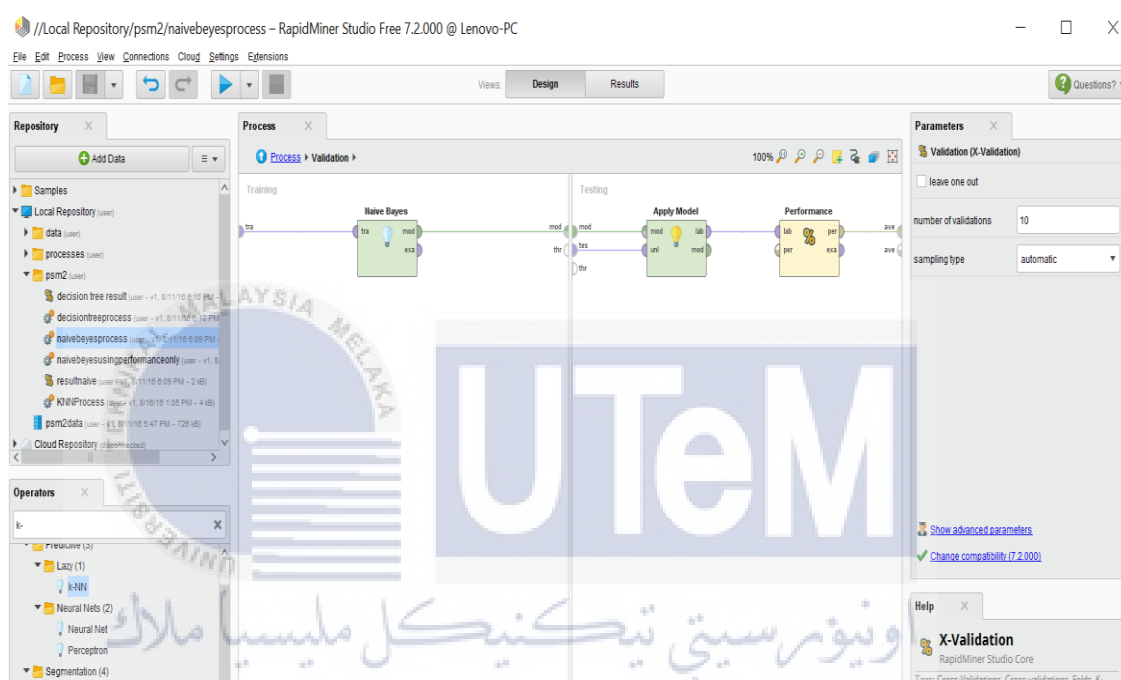


Figure 5.6: Design of Naive Bayes process

Naive Bayes classifier expect that the presence (or absence) of a specific element of a class (i.e. attribute) is disconnected to the presence (or absence) of some other feature. After that, the Naive Bayes Process must be connected to the Apply Model and Performance to get the results.

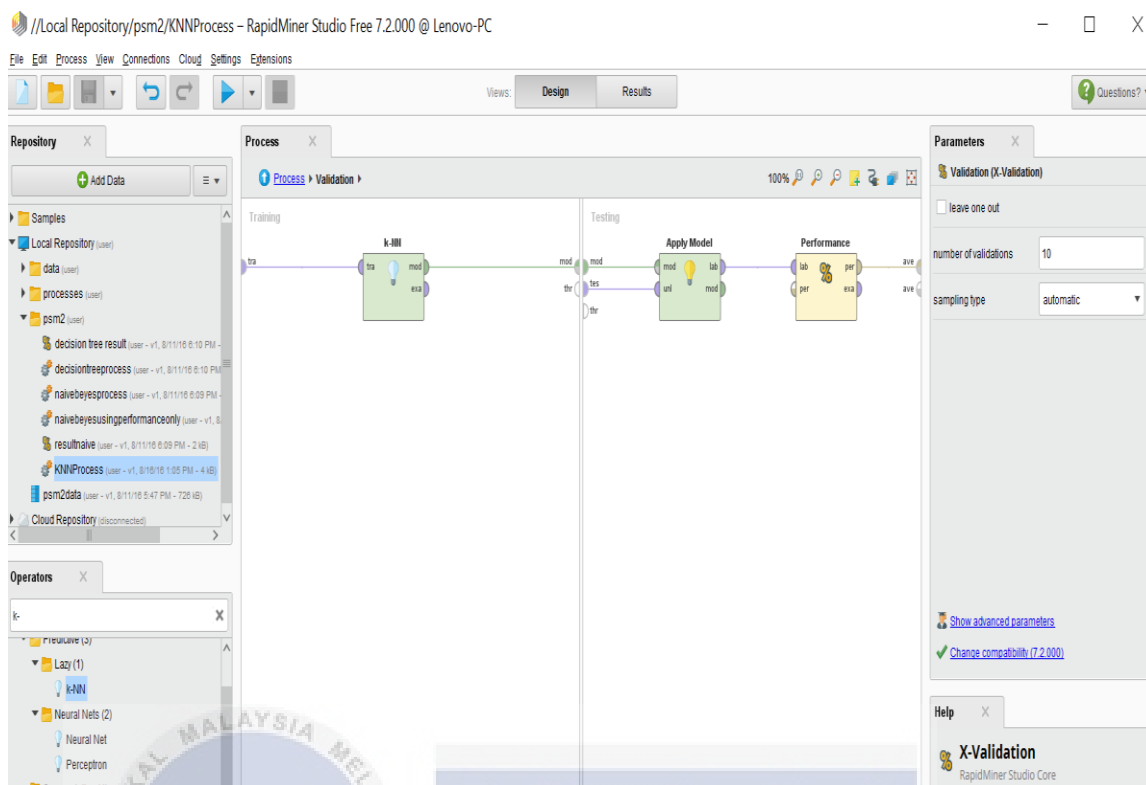


Figure 5.7: Design of K-Nearest Neighbor (KNN) Process

The K-Nearest Neighbor (KNN) algorithm is very simple algorithm compared to other machine learning. Next, the K-Nearest Neighbor algorithm (KNN) will searching k training examples which are the closest one to the unknown example. Next, the “Closeness” is defined as a distance metric. After that, the KNN process must be connected to the Apply Model and Performance to get the results.

In conclusion, the RapidMiner Studio consist of classifier or classification algorithm such as Naive Beyes, Decision Tree and K-Nearest Neighbor (KNN) process which been choose as a process to get the accuracy results. Next, the Apply Model and Performance are selected and must be connected to have a successful result.

5.2.1 List of Parameters

In this study, the list parameters used is the port number, packet time arrival, time of the first packet release. The packet time arrival is captured when the packet is successfully arrived to the destination. The packet arrival is important because it used to get the exact time of the packet when it reached the destination.

Table 5.1: List of Parameters

Parameters	Description
1. Port A	The port that assigned in program application act like sender to send packet to its destination.
2. Port B	The port that assigned in program application act like receiver to receive the packet.
3. Interdeparture time	The successive departures of time.
4. total_packets_a2b	Total packets carried from port a to port b
5. total_packets_b2a	Total packets carried from port b to port a
6. actual_data_pkts_a2b	The actual data packet is deliver from a to b.
7. actual_data_pkts_b2a	The actual data packet is deliver from b to a
8. outoforder_pkts_b2a	The delivery of data packets in a different order which are from b to a
9. idletime_max_a2b	The low priority of time that not impact programs running from a to b

10. idletime_max_b2a	The low priority of time that not impact programs running from b to a
11. throughput_a2b	The amount of data that moved successfully from a to b in a given time period
12. throughput_b2a	The amount of data that moved successfully from b to a in a given time period
13. class	The category that have been assign to the dataset.

5.2.2 Assumptions

The benign traffic is the connection between legitimate skype users. Next, the skype users must have antivirus such as Kaspersky antivirus, which is the use of antivirus not triggered any detection alert. The combined dataset must be cleaning manually to avoid noise and to get a good result.

5.3 Conclusion

In conclusion, the dataset must be converted to Comma Spread Values File (.csv) and must be cleaning to get the good result and to avoid noise that will affect the results that been testing using RapidMiner Studio. After that, the next chapter will further discussed about testing results of the dataset.

CHAPTER VI

TESTING AND ANALYSIS

6.1 Introduction

From the previous chapter, analysis and implementation have been discussed. Testing planning will be discussed further in this testing chapter. This chapter include the result and analysis that used similar dataset from previous. Firstly, the data was run in the Wireshark and the TCP characteristic has been chosen for detection of malicious activity. Next, data mining technique was used to get the result of accuracy, precision and recall in each process in RapidMiner Studio which is in Decision Tree process, Naive Bayes process and K-Nearest Neighbor (KNN) Process.

6.2 Result and Analysis

In this section, the testing part is focused on accuracy, precision, and recall result in each process. The accuracy is about the closeness measurements between the dataset. Next, the precision is showing the same results under a fixed condition of the repeated measurements. After that, the recall, called as sensitivity and based on understanding about the measure of relevance.

6.2.1 Result of Accuracy in each process

accuracy: 99.71% +/- 0.32% (mikro: 99.71%)

	true 1	true 0	class precision
pred. 1	1430	4	99.72%
pred. 0	4	1292	99.69%
class recall	99.72%	99.69%	

Figure 6.1 Decision Tree Process Result (Accuracy)

From the result above, the Decision Tree process gives result which is 99.72% class recall of accuracy for malicious traffic (labelled as true 1) and 99.69% class recall for normal traffic (labelled as true 0). As stated by (Tabish, Shafiq, & Farooq, n.d.) 90% and above accuracy give high performance of malicious detection. It can be shown that Decision Tree process gives high accuracy result which is 99.71%.

accuracy: 80.81% +/- 1.61% (mikro: 80.81%)

	true 1	true 0	class precision
pred. 1	1383	473	74.52%
pred. 0	51	823	94.16%
class recall	96.44%	63.50%	

Figure 6.2: Naive Bayes Process Result (Accuracy)

From the result above, the Naive Bayes process give results accuracy which is 96.44% class recall for malicious traffic (labelled as true 1) and 63.50% class recall for normal traffic (labelled as true 0). As conclusion, the Naive Bayes process give 80.81% accuracy result.

accuracy: 98.94% +/- 0.66% (mikro: 98.94%)

	true 1	true 0	class precision
pred. 1	1424	19	98.68%
pred. 0	10	1277	99.22%
class recall	99.30%	98.53%	

Figure 6.3: K-Nearest Neighbor (KNN) Process Result (Accuracy)

From the result above, the K-Nearest Neighbor (KNN) process give accuracy results which is 99.30% class recall for malicious traffic (labelled as true 1) and 98.53% class recall for normal traffic (labelled as true 0). As conclusion, the K-Nearest Neighbor (KNN) process give 98.94% accuracy result.

6.2.2 Result of Precision in each process

precision: 99.69% +/- 0.38% (mikro: 99.69%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1430	4	99.72%
pred. 0	4	1292	99.69%
class recall	99.72%	99.69%	

Figure 6.4: Decision Tree Process Result (Precision)

From the result of precision above, the Decision Tree process give result with the class recall is 99.72% for malicious traffic (labelled as true 1) and 99.69% for normal traffic (labelled as true 0). As conclusion, the Decision Tree process give 99.69% precision result.

precision: 94.27% +/- 2.61% (mikro: 94.16%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1383	473	74.52%
pred. 0	51	823	94.16%
class recall	96.44%	63.50%	

Figure 6.5: Naive Bayes Process Result (Precision)

From the result of precision above, the Naive Bayes process give result with the class recall is 96.44% for malicious traffic (labelled as true 1) and 63.50% for normal traffic (labelled as true 0). As conclusion, the Naive Bayes process give 94.27% precision result.

precision: 99.23% +/- 0.59% (mikro: 99.22%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1424	19	98.68%
pred. 0	10	1277	99.22%
class recall	99.30%	98.53%	

Figure 6.6: K-Nearest Neighbor (KNN) Process Result (Precision)

From the result of precision above, the K-Nearest Neighbor (KNN) process give result with the class recall is 99.30% for malicious traffic (labelled as true 1) and 98.53% for normal traffic (labelled as true 0). As conclusion, the K-Nearest Neighbor (KNN) process give 99.23% precision result.

6.2.3 Result of Recall in each process

recall: 99.69% +/- 0.38% (mikro: 99.69%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1430	4	99.72%
pred. 0	4	1292	99.69%
class recall	99.72%	99.69%	

Figure 6.7: Decision Tree Process Result (Recall)

From the recall result above, the Decision Tree process give result for the class recall which is 99.72% for malicious traffic (labelled as true 1) and 99.69% for normal traffic (labelled as true 0). As conclusion, the Decision Tree process give 99.69% recall result.

recall: 63.51% +/- 3.28% (mikro: 63.50%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1383	473	74.52%
pred. 0	51	823	94.16%
class recall	96.44%	63.50%	

Figure 6.8: Naive Bayes Process Result (Recall)

From the recall result above, the Naive Bayes process give result for the class recall which is 96.44% for malicious traffic (labelled as true 1) and 63.50% for normal traffic (labelled as true 0). As conclusion, the Naive Bayes process give 63.51% recall result.

recall: 98.53% +/- 1.40% (mikro: 98.53%) (positive class: 0)

	true 1	true 0	class precision
pred. 1	1424	19	98.68%
pred. 0	10	1277	99.22%
class recall	99.30%	98.53%	

Figure 6.9: K-Nearest Neighbor (KNN) Process Result (Recall)

From the recall result above, the K-Nearest Neighbor (KNN) Process give result for the class recall which is 99.30% for malicious traffic (labelled as true 1) and 98.53% for normal traffic (labelled as true 0). As conclusion, the K-Nearest Neighbor (KNN) process give 98.53% recall result.

6.2.4 Comparison result of each process

Table 6.1: Comparison results in each process

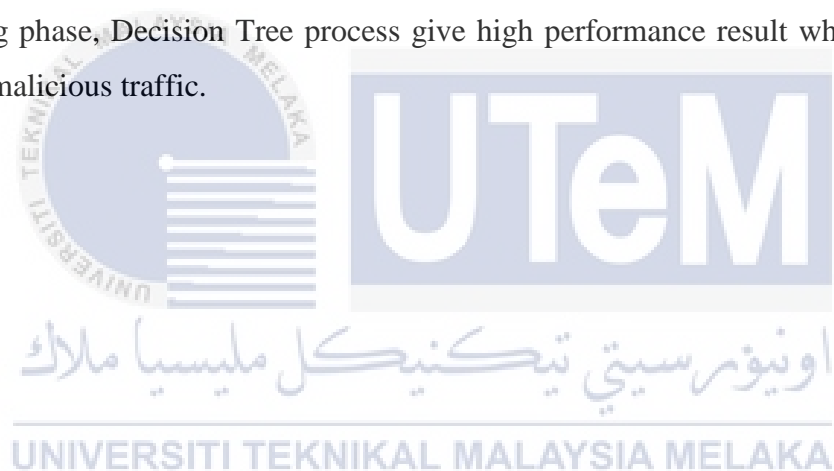
Process Name	Results		
	Accuracy Result	Precision Result	Recall Result
1. Decision Tree	99.71%	99.69%	99.69%
2. Naive Bayes	80.81%	94.27%	63.51%
3. K-Nearest Neighbor (KNN)	98.94%	99.23%	98.53%

From the table above, the highest and good accuracy is using Decision Tree process which give 99.71% accuracy result compared to Naive Bayes process which give 80.81% and K-Nearest Neighbor (KNN) which give 98.94% accuracy result. Next, the highest and good precision result is by using Decision Tree process which give 99.69% precision result compared to Naive Bayes process which give 94.27%

and K-Nearest Neighbor (KNN) which give 99.23% precision result. After that, the highest and good result for recall is using Decision Tree process which give 99.69% recall result compared to Naive Bayes process which give 63.51% and K-Nearest Neighbor (KNN) which give 98.53% recall result.

6.3 Conclusion

As a conclusion, the characteristic result between the normal traffic and the malicious traffic dataset in Skype Application can be evaluated by using data mining technique. The data mining technique have been done in RapidMiner Studio by using Naive Beyes, Decision Tree and K-Nearest Neighbor (KNN) process which can be used for analysing the malicious activity and to get the accuracy result. Lastly, in testing phase, Decision Tree process give high performance result which can detect high malicious traffic.



CHAPTER VII

PROJECT CONCLUSION

7.1 Introduction

The project conclusion is summarize of the testing result that have been done in previous chapter that is TCP characteristics for the P2P normal traffic and the abnormal traffic of the P2P in the Skype Application. Next, in this chapter the detailed of the discussion are including project summarization, project contribution, further work and the limitation of the project.

7.2 Project Summarization

The objective of this research are discussed early in Chapter 1 which is to investigate the characteristics of the botnet behaviour in peer-to-peer system which is have been done in the chapter 2. Next, the parameters, protocols, the dataset of normal and abnormal traffic in P2P Skype application was defined detailed for recognizing of the P2P botnet behaviour in Skype application.

P2P normal and P2P abnormal have main differences which is how the abnormal activity exists in selecting the parameter and protocol. Normally, the attacker will always scan the network, retrieve the data after they success attack the network. They will try to denial the services will repetitive request to the server, the attack is as denial of service. Next, the methodology of the research is an important chapter because it is a guide for analysis the P2P botnet. Hence, the flow is form to conduct the correct sequence.

7.3 Project Contribution

In this project contribution, the data mining technique will be used to get the performance detection result in identifying the behaviour of P2P botnet which have been done in Testing Result and Testing Analysis in the previous chapter.

7.4 Project Limitation

In completing the task in this study, there is some limitation which is the difficulties in cleaning the dataset to avoid noise which leads in the performance result of the detection. Next, the dataset that have been run become corrupted for this study. After that, there is a lack of library references and Internet source of the P2P botnet in Skype Application, so that it was hard to complete the task.

7.5 Further Project

In future, there are some suggestions which is develop a tool to clean the dataset instead in presence do it manually. Next, for further project, develop the tool which is dataset is easy to be trace and be analysed. After that, multi-references for library source and internet source for the P2P Botnet in Skype Application.

7.6 Summary

In conclusion, this project has been successfully done within the time given by Projek Sarjana Muda (PSM) committee and research objectives are achieved. This research (P2P botnet in skype application) can be useful in the future as reference, by researchers who are focused on P2P botnet detection. Moreover, with this research, skype users can be more alert about the malware detection in Skype application. Therefore the virus attack can be reduced.

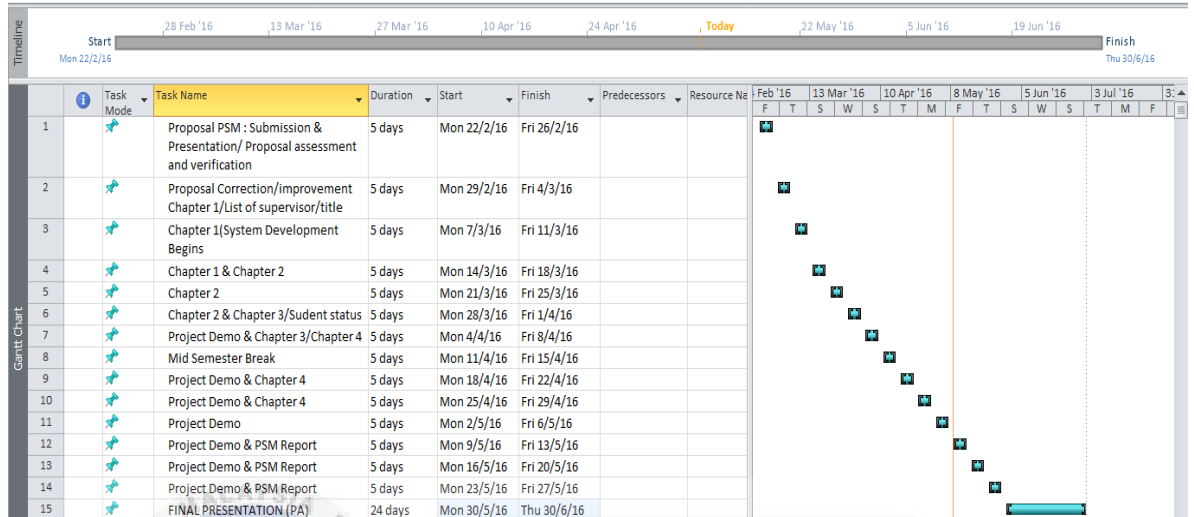


REFERENCES

- Abdullah, R., & Abdollah, M. (2013). Revealing the Criterion on Botnet Detection Technique. *IJCSI International Journal of Computer Science Issues*, 10(2), 208–215.
- Abdullah, R., & Ud, M. M. (2011). Recognizing P2P Botnets Characteristic Through TCP Distinctive Behaviour. *International ...*, 9(12), 12–16. Retrieved from <http://eprints2.utm.edu.my/3580/>
- Baset, S. A., & Schulzrinne, H. G. (2006). An analysis of the Skype peer-to-peer internet telephony protocol. *Proceedings - IEEE INFOCOM*. <http://doi.org/10.1109/INFOCOM.2006.312>
- Bayer, U., Moser, A., Kruegel, C., & Kirda, E. (2006). Dynamic analysis of malicious code, 67–77. <http://doi.org/10.1007/s11416-006-0012-2>
- Brands, E. H. T. B., & Karagiannis, G. (2009). Taxonomy of P2P Applications.
- Gadhiya, S. (2013). Techniques for Malware Analysis, 3(4), 972–975.
- Margaret Rouse. (2009). What is peer-to-peer? - Definition from WhatIs.com.
- Milojicic, D. S., Kalogeraki, V., Lukose, R., Nagaraja, K., Pruyne, J., Richard, B., ... Xu, Z. (2002). Peer-to-Peer Computing Peer-to-Peer Computing.
- Quang Hieu Vu, Mihai Lupu, B. C. O. (2010). Peer-to-Peer Computing.
- Raihana Syahirah, A. (2011). Recognizing P2P Botnets Characteristic Through TCP Distinctive Behaviour.

APPENDIX

APPENDIX 1



اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

