International Journal of Innovative Research in Science and Engineering
Vol. No.2, Issue 11, November 2016
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

# A MALWARE ANALYSIS BY CAPTURING THE PACKETS IN A NETWORK UNDER A FORENSIC INVESTIGATION

**Darien Obrien Fabian**

*Bsc (Hons) IT Specialism in Forensic Computing, Asia Pacific University, Kuala Lumpur, Malaysia*

**ABSTRACT**

*In this project, aim is to have malware to be analyze that are in the network. Hence, we need to find data about malware so that we can know what kind of malware is hiding behind the data. Data like how to obtain malware in the network by using existing tools is essentials for analyzing malware but it is useless without having any knowledge at all when it comes to analyzing. The information that store in the network are far too wide and malware can be hidden in any file that allow them to pass through a network without noticing or unnoticeable by antivirus software. Hence, an in-depth research on capturing packets in the network should be done in order to know and analyze malware in a network.*

## 1. INTRODUCTION

1. Malware is malicious code that have been widely spread around the world for the purpose of doing something irresponsible. In outside world where network is connected everywhere and anywhere, people are starting to lower their guard down about what can happen to them if their devices has been taken over by other devices. Also, in this modern world where technology has drastically became essential part in our daily life, some irresponsible people like cyber criminals using this advantage to get people personal information to do criminal. The advantage is that they send malware to a specific device and get the information from user inputs. Malware is malicious code that been made by irresponsible user for their own benefit. Malware can be transfer by any types of medium. For example, from a USB flash drive that we usually use to transfer files, or people hack into our devices, they can plant malware by using a secret passage that they have created it in the device at the first place. Besides from USB, malware is also transferable by using packets. A few malware like spyware, ransom ware, Trojan and etc. are cyber criminals commonly used when they are trying to steal information from another device. That is why using the internet, anyone from outside can just steal our personal information by installing malicious code by transferring in the packets to their devices. Since network is so huge, we cannot even track what coming in and out to our devices. With the usage of WIFI or mobile data, user sometimes do not notice when they surf the internet, malicious code or malware is already trying to get into their devices in the form of packets that transfer from one device to another. Hence in this proposal, we will create our own network environment which do not include real world and scan the network by using a few samples of malware like spyware so that we can capture malware in the packets by capturing it to make it detectable and research it to

ISSN (O) 2454-9665
ISSN (P) 2454-0663

find more information about the malware. By capturing the packet, we can know the fundamental of the malware detection. Report will be generated after everything so that it can be for future reference.

### 1.1 Problem Statement

Malware nowadays is a common thing that affect people devices. Malware does not care whether it is mobile devices, laptops, desktop, server and other internet of things (IOT), it just affect a device that has vulnerability inside it. So the vulnerability of a device allow malware to affect it because that is its chance to grow and take over a specific device. A problem about malware is that, it is very hard to be detect by an antivirus. This to be known due to malware can be created new and fresh and send to victim. Thus it is new, it is very hard to be detect by antivirus due to new characteristics. With new traits will bring by the malware, antivirus cannot recognize the characteristics as it not exists in antivirus database. Besides that, most of the network server do not expect executable code in their traffic. This shows that malware can be execute without it give a warning to the server. Also, scanning of a malware is not executed in the server. In other word, it is undetectable by the network only within a few minutes. Hence, in this proposal, we are going to capture the packets that contain malware so that it can be used for future reference when malware is attacking a device. The information gather allow us to think about the countermeasure on the malware sample that we are going to use in this project.

In addition, there is also limitation on tools when it comes to packet analyzing. Depends on tools that used for analyzing the packets, some tools might not give accurate information or the information that provided is not what we wanted. Hence, in this project while analyzing the packets, we can determine the better tools when it comes to packet analyzing. Effects on people about malware is quite a severe ones too. People only depend on their antivirus software to do this detection. Sometimes antivirus software, it is not so accurate because there are new malware keep producing. Thus, new malware means new signature. Antivirus cannot detect new malware. Thus, it is so easy for malware to be transferrable via network. Lack of knowledge about malware is also an advantage that malware can affect other devices via network because without knowledge, they do not know what is in and out from their computer and to their computer. Problem about malware may affect people severely as it is growing by the day. Most people will suffer from important files keep missing, their bank account will get empty without their noticing, got a different perspective from the society and other effects that malware can do. Hence, in this proposal we will provide the information how malware going to take place.

## II. LITERATURE REVIEW

### 2.1nEther:In-guest Detection of Out-of-the-guest Malware Analyzers

Malware analysis is very efficient way to fight against malware code. Hence, security experts is using heavy use of various virtualization techniques that provide a certain level of isolation between the host and the code that bound to be analyzed. Even so, it is so easy being detected and also easy to evade by the malware. Due to this, malware analysing experts has created a framework to demonstrate their concept in analysing malware. Their concept is an approached that allow hardware assisted virtualization platforms and out-of-the-guest malware analysis framework to be detected. They already created their first implementation of framework and it is installed on XEN and Intel VT.

ISSN (O) 2454-9665
ISSN (P) 2454-0663

The method that they are using in analysing malware by creating a new framework to demonstrate their concept is nEther [1]. It is also another framework that it allow hardware assisted virtualization platform and out-of-the guest malware analysis to be possible. Even though the first Ether had implemented but it cannot stop in-guest timing attack and also cannot prevent the attacked that was supposed to be blocked by Ether. The results from the method is that it capable of detecting the presence of the out-of-the-guest malware analysis Ether. They also found out that timing is a related problem to their method. At the same time they also found out about three different classes that is user interface is very slow, relative timing sources found and hardware related timing sources. Thus in their opinion, some case like involving in perfect transparency cannot be guaranteed neither theoretically nor practically.

### 2.2 The Detection of 8 Type Malware botnet using Hybrid Malware Analysis in Executable File Windows Operating Systems

In this article, the researchers are doing an analysis on 8 types of botnets which will be put in windows operating system and also which nowadays had infected at other device a lot on Windows. Botnets are being used for negative purposes like cybercrime. Most common attacks like Distributed denial of service or we usually called it DDoS or information stealing. Windows operating system is the main target for botnet and it will be executable easily as it comes in the form of .exe file. The reason they chose Windows it is because there a big amount of application that exists in executable file. This is why it is so difficult to differentiate whether the .exe file is a botnet or not. Hence, from the researchers perspective, there are two techniques that can be used in analyse the malware. First one is static analysis and the second one is dynamic analysis [2]. To test in static analysis, the researchers are using few types of botnets and then they will put it in VM malware analysis laboratory. For static, the botnets will be examined without any execution. There are few techniques that will be implement in this static analysis. The techniques are File Fingerprinting by Hashing, Extraction of Hard Coded Strings, Disassembly, Extract Linked Libraries and Functions, and Debugging. While for dynamic analysis with also using few types of malware, those malware samples will be performed by using admin's permission rights. There are also a few techniques that will be apply in this analysis. Those are Viewing Process Details, File System Monitoring Activities, Registry Monitoring Activities, and Network Traffic Monitoring.

In conclusion, by doing the testing and analysis of 8 malware botnet samples, first thing that can conclude is that the data recommendation from the outcome of hybrid malware analysis, characteristics data of botnet can be used in order to find the botnet and second conclusion, by looking at the malware botnet linked libraries, enable the detection of what is contain in the malware botnet's functionalities.

### 2.3Panorama: Capturing System-wide Information Flow for Malware Detection and Analysis

The researchers have been doing observations on malicious information access and processing behavior is the basic attributes that has a big amount of malware categories that penetrating into user's privacy. That is why the application are called malicious application from being a software. They thought the techniques that are already exists for detecting the malware and analyzing unknown code samples are not enough in terms of information gathering. With the use of Google Desktop as a scenario, they proposed a system that can detect but already gave them a result of false positive and analyse malware by getting the information behaviour and processing

ISSN (O) 2454-9665
ISSN (P) 2454-0663

attribute [3]. They called the system, Panorama.In this system Panorama, they have used several tools to put into together as an architecture for Panorama. Some part of the architecture has been designed just to fit into the system. Hence, in order to perform their automatic malware detection and analysis, they load the sample into the environment that already been set. Then they will run a few of automated tests. For each test, they will create a situation where the sensitive data that will be sent to trusted application in order to observe the malicious to react. From there, the information access and processing behavior will be record down in a report.

In conclusion, the researchers have successfully monitored and record the behaviour down. This means from Google Desktop being used a case study, Panorama precisely capture its information access and processing behaviour. It has capture the inherent characteristics of a wide-spectrum of malware like Keyloggers, packet sniffers and so on. From 42 malware sample that had been evaluated, they got the outcome of zero false negative. Hence, they hope that this system can help malware analyst to work better in the future.

### 2.4 Libtrace: A Packet Capture and Analysis Library

Those researchers has introduced an open-source software library for reading and writing network packet traces that is libtrace. When compare to other library, this libtrace software library can offer performance and the enhancement of usability [4]. They had described in this paper about the main features of libtrace and also they demonstrated how this software library helps user to analyze without considering the information of the format on what we will be capture.The method that they use in this library is that it has been created by using C programming. There are few features when they use libtrace like there is API installed that made to be streamlined. Besides that, when the network is being captured, with feature of capture format agnostic, it would not change the code, or making any format conversion unnecessary. There are other features like protocol decoding that allow direct access to the header, compression and performance that enhance the ability to capture the packet without alter the information.

In conclusion, they successfully implemented libtrace in the network sniffing that allow the reading, processing and writing packet traces. They have made an evaluation with applying two trace analysis tasks using a few types of analysis libraries like libpcap. Hence, from this, the outcome from using libtrace is that this library gives out the best performance for I/O bound analysis and also the second fastest when running a heavy task in the system.

### 2.5 In-Cloud Malware Analysis and Detection: State of the Art

Researchers in this article are doing a research on the Internet of Things. They found out due to limited resources, it is quite very challenging to detect those devices for malware since malware is active in affecting people's devices. Hence, in this paper is a research on the current system, discussion about pros and cons and also recommendation on how to improve in-cloud analysis and detection system[5]. Besides that, they introduce a new three layered hybrid system with a lightweight antimalware engine.The method that they use in this research is that in the cloud computing has already its own of malware detection but it cannot detect sophisticated malware due to insufficient resources. Hence, by implementing a hybrid system and put it into cloud computing, they have created an antimalware engine called *lightweight*. From this, there are two new features that make it more special than the others system. Those are LWE (lightweight antimalware engine) and

ISSN (O) 2454-9665
ISSN (P) 2454-0663

lightweight agent (LWA) [5]. Hence, with these system, it can detect either with a normal type malware analysis and also simple signature based techniques, or the sophisticated ones but all it need with just more resources.

In conclusion, they have successfully found the pros and cons about the system on how potential the cloud performance in malware detecting and they had already implemented the hybrid system. With the introduction of the 3 layered, these features can give a faster performance in detecting the malware, protect the client and lessen the bandwidth between the cloud and client. Hence, in the future, they will compare cloud and a stand-alone server of their specific performance.

### 2.6 Kindred Domains: Detecting and Clustering Botnet Domains Using DNS Traffic

In this article, researchers are detecting malware bots and grouping those cluster distinct grouping of domain names that are queried by a lot of infected machines. They analysed the domain name system traffic, like Non-Existent Domain (NXDomain) authoritative name that has the strongest connection of groups with malware related domains [6]. They observed these domain name system in a global scale and they make use of the inner to plot a system ability of scanning precisely on malware domains without the need of obtaining a malware sample. A results of the system will give a special point of view on the current situation, especially the one that malware are stuck onto a DNS.They have use a popular family of malware called Conficker that uses a technique called Domain Generation Algorithm (DGA) to evade detection. Before they go into in detection of malware, they did a research on the traffic on how DNS can be similar when it comes to infect by malware. They had sorted the traffic based on the similarities and use this opportunities to measure to cluster domains. Hence, to detect the malware in the domain name system, they have constructed a system that computes the comparison between DNS similarities measures and subsequently performs groupings.

In conclusion, they have successfully examined the special traits of traffic that been sent by name servers to the authoritative name servers. They have specifically use Conficker because it is widely used and powerful malware at one time. From the results, they found that not only can detect Conficker but other family like Flashback and other DGA family also can detect by using the same system.

### 2.7 MAIL: Malware Analysis Intermediate Language - A Step towards Automating and Optimizing Malware Detection

In this article, it explains about how malware can never keep the same order of opcode and that is called metamorphism. Hence, metamorphism is a technique that it keeps the malware behavioural to be changed constantly. The effect of this is that malware allow itself not to be detect even with the help of the tools. The researchers have made an automated and can optimise the detection of the malware.Hence, they have made a language called malware analysis intermediate language (MAIL) [7]. This language has built-in of detection tool and malware analysis. How this work is that they need to give this language a pattern that can be used to interpret a control flow graph in order to match a pattern and detect malware that changes its own signature. After a program has already translated to MAIL program, this where detection starts. When a program that contains part of the control flow of a training malware sample, is classified as a malware.

In conclusion, they have successfully developed MAIL to as an intermediate language. Disadvantage to this is that MAIL is not suitable to perform for dynamic analysis but only for static analysis. Not only focusing on this

ISSN (O) 2454-9665
ISSN (P) 2454-0663

project, they are also currently do a further research into enhancing the tool to increase its precision on detecting the malware.

### 2.8 MalwareVis: Entity based Visualization of Malware Network Traces

In this article, researchers have working on a utility that gives security researches about malware a way to browse, filter, view and see the difference of malware network traces as entities [8]. They proposed in this article where the viewing of the network traces are in cell-like visualization model during malware sample execution. They have encoded it and the design is taking into their concern which takes part in able to give them to give a clear global view of those malware sample's behaviour.Hence, for this type of function they have made a utility called MalwareVis where they will do a demonstration in the real world with malware sample and then show their individually activity patterns. The method to use this utility is that from the pcap file that the utility will read, it will capture the data and gather a few of malware sample to give a feedback on it. From the malware that had been captured will be extracted from the protocol information and arrange into an array. Hence, at last m a visualization module generates table views and cell views. So the viewing will produced from MalwareVis. This can organize the malware to be arranged to let people to see.

In conclusion, they found that malware network traces have important characteristics of a malware sample's behaviour. At the same time, it is also to address the need of a visual representation in the workplace of malware analysis. In the future, they will discover several parts on how to relate visualizing with a collection of malware network traces.

### 2.9 Characterizing Kernel Malware Behaviour with Kernel Data Access Patterns

There are a lot of ways to determine malware variants when doing malware analysis. In this article, the aim is to find out the characteristics of malware behaviour. Hence, the researchers have found out a new approach to differentiate kernel malware's behaviour. By using kernel data access patterns. Not like other research file where they use CFG that control flow graph in finding the matching pattern in order to find the similarity pattern between the graph and the malware. Hence, to do demonstration on malware analysis, the method that they use is that they first generated the signature of those three classic rootkits. They will do a further observation to similar data access patterns in the signatures of the tested rootkits and exposed popular rootkit attack operations by ranking common data behaviour across rootkits. With this examples, it can be used for the chosen research topic as a sample.

In conclusion, from the data that they have researched on about malware which they have created as a classic rootkit, they have successfully presented a new approach to differentiate by using a pattern collected for kernel data access unique to the malware. Hence, for my research topic, it is good to practice another analysis when it comes to malware. This is due to having a strong and solid prove about the malware that we are going to capture and having similarities out of it.

### 2.10Mal-ID: Automatic Malware Detection Using Common Segment Analysis and Meta-Features

In this article, their aim is to analyse malware without touching the unpacking or disassembling part in the

ISSN (O) 2454-9665
ISSN (P) 2454-0663

process [9]. Hence, it had introduced a methodology where it detects malware in executable file without pre-processing like unpacking the malware or disassembling the malware into pieces. So this methodology the researcher had created is only suitable for static analysis that uses common segment in order to detect malware files. Hence, this is suitable for my research as it is almost similar to what want to do for my research. The method that they use is implementing this methodology called Mal-ID. It is a basic method that only uses some part of malware to detect it whether is it a malware file or not. At the same time, they also introduced 2 Mal-ID extensions that can improve Mal-ID in various kind of situation. This is good to test with different kind environment. This is to ensure that malware cannot fight back or to be undetected by that specific methodology.

In conclusion, the mentioned method above has performs well in a mixed environment. By mixed, it means where the files are going under obfuscate and plain executable files. This method is good to analyse but it only limit to one type of malware analysis that is static. That means the malware cannot be executed while the testing is going on.

## III. RESEARCH DESIGN

### 3.1 Aim

The main aim of this project is to analyze malware in the network by capturing the packets so that it can be detected in order not to let it take over a specific device and take control of it. This also at the same time can improve the forensic investigation with the information provided by the malware.

### 3.2 Objectives

- To set up own network environment in order to capture the spyware
- Create own spyware with the existing algorithm so that antivirus would not detect
- To give out information on how is spyware attack attacked the system
- By using existing tools, we have to capture the packet in order to analyze the spyware
- To store and save the data of the spyware information in a log file
- Identify the spyware in the network from a notification that will pop out after it analyze
- To generate a report based on the results on what we have analyzed

### 3.3 Research Questions

- Identify the traits of a normal packets and the packets that affect with malware and compare the difference
- What are the types of framework should be used when we trying to capture spyware in the packets?
- Determine the process of capturing the packets within a network in order to get the information of a malware.
- Identify the type of techniques that going to use when capturing the packets while not disrupting the malware packets.

In this proposal, there are a few methods can be used to gather and collect data to answer the research questions that have mentioned. Methods are like sampling, observation, interview and distributing questionnaires. Hence in my opinion, methods that are suitable for gathering those data are by sampling and distributing questionnaires.

ISSN (O) 2454-9665
ISSN (P) 2454-0663

First method to gather data is by distributing questionnaires. Why questionnaires? This is due to amount of time that information that can be gathered in a large amount of people only for a short period of time and in a relatively cost effective way. The results that they produce is very fast and the outcome of those results can be calculated easily by using a software or by a researcher. Since questionnaires have a mix of closed and open questions, this give respondent some feedback or opinions on what are we working on and they also have a choice to just tick what they think is relevant. For this project, target audience will be students because they are the current generation who uses technology in their daily life. Questionnaires will be distributed to these audience and they answer based on their own perspective and relevancy. Hence, research question will be used to ask respondents about whether are they agree on the research. This shows that this is based quantitative research type where it requires the amount of respondent to get feedbacks about the research.

Another method that suitable to gather data is by using sampling method. Why sampling method? This is due to in a very large population of people, there will be so many with different kind of interest. Hence, by representing those with bigger population to a sample representative, it reflects on the same traits as the sample. If carefully choose a sample, might getting a high chance on similar result. Since the target audience is students, they are a very large population. To determine how much sample we should research on, there are few types of sampling that can be used like probability sampling, stratified sampling, cluster sampling. For this project, stratified sampling allows to divide a large group into a sub group and in a sub group, there are many elements. From the elements that they have, we can sort them out according to what we think is suitable for our research. The result is rating by majority. Hence, this research is based on qualitative research that requires comments and feedbacks about the research.

### 3.4 Work Plan / Timeline

The following are the work plan and Gantt chart for the work proposed schedule shown in "Table.1 and Fig.1".

Table.1 Work Plan

| | | | | |
|---|---|---|---|---|
| 1 | Design questionnaires | 3 days | Mon 10/31/16 | Wed 11/2/16 |
| 2 | Conduct survey questionnaires | 3 days | Thu 11/3/16 | Mon 11/7/16 |
| 3 | Analyze results questionnaires | 4 days | Tue 11/8/16 | Fri 11/11/16 |
| 4 | Determine an experiment | 2 days | Mon 11/14/16 | Tue 11/15/16 |
| 5 | Choose a sample | 2 days | Thu 11/17/16 | Fri 11/18/16 |
| 6 | Conduct the survey | 4 days | Mon 11/21/16 | Thu 11/24/16 |
| 7 | Analyze the results from the survey | 4 days | Mon 11/28/16 | Thu 12/1/16 |

International Journal of Innovative Research in Science and Engineering
Vol. No.2, Issue 11, November 2016
www.ijirse.com
ISSN (O) 2454-9665
ISSN (P) 2454-0663

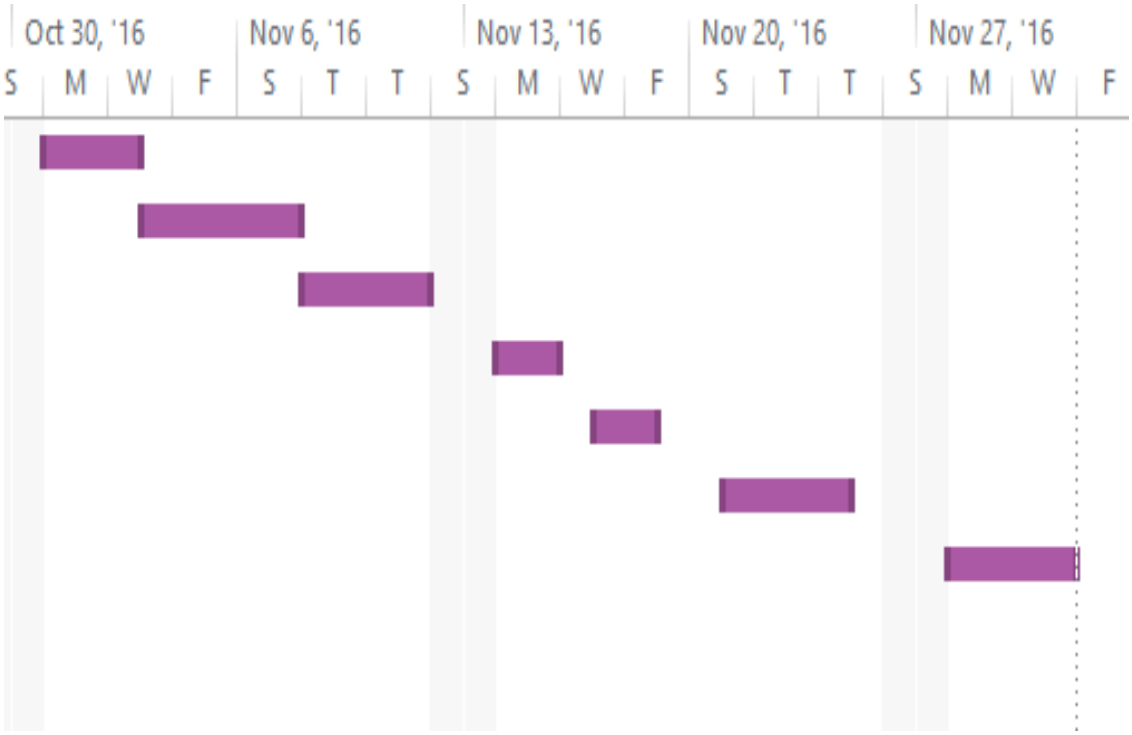


Figure.1 Gantt Chart

## IV. CONCLUSION

Besides knowing how to obtain the packets in the network, we have to know on what is the next step to analyse malware in the network. Without further step on analysing, we cannot proceed on how to determine the behaviour of the malware and what is its capabilities in affecting someone device. In analysing it, there are also a few types of analysis that need to conduct. Hence, throughout all these analysis, we need to compare whether all these analysis are having the same traits from the malware. This is why the limitation from here is that one type of analysing is not enough to proof a malware because sometimes malware can act differently in a different environment. Besides researching on how to capturing the malware in the packets, we need to study on the malware behaviour so that malware's ability we easily to identify. Hence the malware analysis done and discussed in detail about the packets in a network under a forensic investigation.

## V. ACKNOWLEDGMENT

The authors would like to share thanks to Mr Umapathy Eaganathan, Lecturer in Computing, Asia Pacific University, Malaysia for his constant support and motivation helped us to participate in this International Conference and also for journal publication.

ISSN (O) 2454-9665
ISSN (P) 2454-0663

**REFERENCES**

[1] Pek, G. (2016). nEther: In-guest Detection of Out-of-the-guest Malware Analyzers. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=1972554&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[2] Satrya, G. (2016). The Detection of 8 Type Malware botnet using Hybrid Malware Analysis in Executable File Windows Operating Systems. 1st ed. [ebook] p.1. Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2781567&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[3] Yin, H. (2016). Panorama: Capturing System-wide Information Flow for Malware Detection and Analysis. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=1315261&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[4] Alcock, S., Lorier, P. and Nelson, R. (2012). Libtrace: A Packet Capture and Analysis Library. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2185382&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[5] Alam, S., Sogukpinar, I., Traore, I. and Coady, Y. (2014). In-Cloud Malware Analysis and Detection: State of the Art. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2659730&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[6] Thomas, M. and Mohaisen, A. (2014). Kindred Domains: Detecting and Clustering Botnet Domains Using DNS Traffic. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2579359&CFID=788341515&CFTOKEN=88998673 [Accessed 21 May 2016].

[7] Alam, S. and Horspool, R. (2013). MAIL: Malware Analysis Intermediate Language - A Step Towards Automating and Optimizing Malware Detection. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2527006&CFID=788341515&CFTOKEN=88998673 [Accessed 22 May 2016].

[8] Zhuo, W. and Nadjin, Y. (2012). MalwareVis: Entity-based Visualization of Malware Network Traces. 1st ed. [pdf] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2379696&CFID=788341515&CFTOKEN=88998673 [Accessed 22 May 2016].

[9] Tahan, G. and Rokach, L. (2012). Mal-ID: Automatic Malware Detection Using Common Segment Analysis and Meta-Features. 1st ed. [ebook] Available at: http://dl.acm.org.ezproxy.apiit.edu.my:2048/citation.cfm?id=2343677&CFID=788801438&CFTOKEN=91560857 [Accessed 22 May 2016].