ABSTRACT

Computer viruses and other forms of malware have viewed as a threat to any software system. A computer virus is a piece of software which takes advantage of known weaknesses in a software system. It has the capability to deliver a malicious infection. A common technique that virus writers use to avoid detection is to enable the virus to change itself by having some kind of self-modifying code. This kind of virus is commonly known as a metamorphic virus, and can be particularly difficult to detect. As being discussed, metamorphic viruses have a potential to avoid any signature-based detection schemes by implementing code obfuscation techniques in an effort to defeat it. In metamorphic virus, if dead code is added and the control flow is changed sufficiently by inserting jump statements, the virus cannot be detected. In this project we first developed a code obfuscation engine. We then used this engine to create metamorphic variants of a seed virus and performed the validity of the statement about metamorphic viruses and signature based detectors. Last but not least, we have propose a profile which enclose the information about the existing metamorphic viruses infection.

ABSTRAK

Virus komputer dan lain-lain pepijat menjadi satu ancaman kepada mana-mana sistem perisian. Virus komputer didefinasikan sebagai satu program yg mengambil kesempatan ke atas kelemahan yang ada pada sesuatu sistem perisian.Ia mempunyai kebolehan unutk menyebarkan sebarang jangkitan pepijat yang telah diprogramkan. Bagi mengelak daripada dikesan, penulis atau pencipta virus sering menggunakan teknik pengubahan diri dengan megubahsuai kodnya sendiri. Jenis virus ini dikenali sebagai virus *metamorphic* dan ianya sukar untuk dikesan kehadirannya.Seperti yang diperbincangkan virus *metamorphic* yang menggunakan teknik pengeliruan kod mempunyai potensi bagi mengelak daripada dikesan oleh sebarang perisian antivirus yang menggunakan teknik pengesanan jenis paten. Jika *dead code* ditambah dan aliran kawalan diubah secukukupnya dengan memasukkan *jump statements*, virus tersebut tidak akan dapat dikesan kehadirannya. Di dalam projek ini, kami pada awalnya akan membangunkan enjin kod pengeliruan. Setelah itu, kami akan menggunakan enjin ini untuk membentuk pelbagai jenis virus *metamorphic* berasaskan daripada benih virus yang digunakan. Kemudian, kami akan mengesahkan penyataan tentang virus *metamorphic* dan cuba mengesan kehadirannya menggunakan pengesan jenis paten. Akhir sekali, kami mencadangkan satu profil tentang kewujudan penyebaran oleh virus *metamorphic*.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AV	-	AntiVirus
HMM	-	Hidden Markov Model
NGVCK	-	Next Generation Virus Creation Kit
MPCGEN	-	Mass Code Generator
CRC	-	Cyclic Redundancy Check
PS-MPC	-	Phalcon/Skism Mass Produced Code Generator
VM	-	Virtual Machine
CFG	-	Control Flow Graph Personal Computer
CPU	-	Control Processing Unit
PC	-	Personal Computer
RAM	-	Random Access Memory

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

Computer viruses have created a successful culture as they have the world's computer population. The also have become the subject of widespread urban legends and hoaxes, as popular television shows and movies. Yet they have not received much scientific analysis with in it.

Much of their popular presence is by attaching themselves to a host (a program or computer instead of a biological cell) and used the host's resources to make copies of them (Kephart et al., 1993).

Computer viruses can trace their pedigree to John von Neumann's studies of self replicating mathematical automata in the 1940s. Although the idea of programs that could infect computers appears in 1970s, the first well documented case of a computer virus spreading "in the wild" occurred in 1986 (Essam et al., 2008). It happened when a code snippet known as the "Brain" virus appeared on several dozen diskettes at the University of Delaware. Today viruses afflict at least a million computers every year. Users spend several hundred million dollars annually on antivirus products and services, and this figure is growing rapidly.

Over the past two decades, the number of viruses has been increasing rapidly. We have seen several attacks that caused great disruption to the Internet and brought huge damage to organizations and individuals. For example, in 1999, the infamous

Melissa virus infected thousands of computers and caused damage close to $80 million; while the Code Red worm outbreak in 2001 affected systems running Windows NT and Windows 2000 server and caused damage in excess of $2 billion (Washingtonpost, 2008). Computer virus attacks will continue to pose a serious security threat to every computer user. Today viruses afflict at least a million computers every year. Users spend several hundred million dollars annually on antivirus products and services, and this figure is growing rapidly.

Generally a computer virus causes damage to the host machine. The damage can be done to a number of different components of the computer's operating and file system. These include system sectors, files, macros, companion files and source code. The always connected world of internet is a soft target for viruses. Viruses use internet connectivity to spread across the world faster and create havoc. The early detection of viruses is imperative to minimize the damages caused by them.

Antivirus companies have updated and evolved their technologies to fight these viruses. Yet the virus writers too changed their tactics and mode of attack to create more complex and tough to avoid detection and the game of cat and mouse continued. Figure 1.1 shows number of new malicious threats rise from year 2002 until the end of year 2007 (Turner et al., 2008).

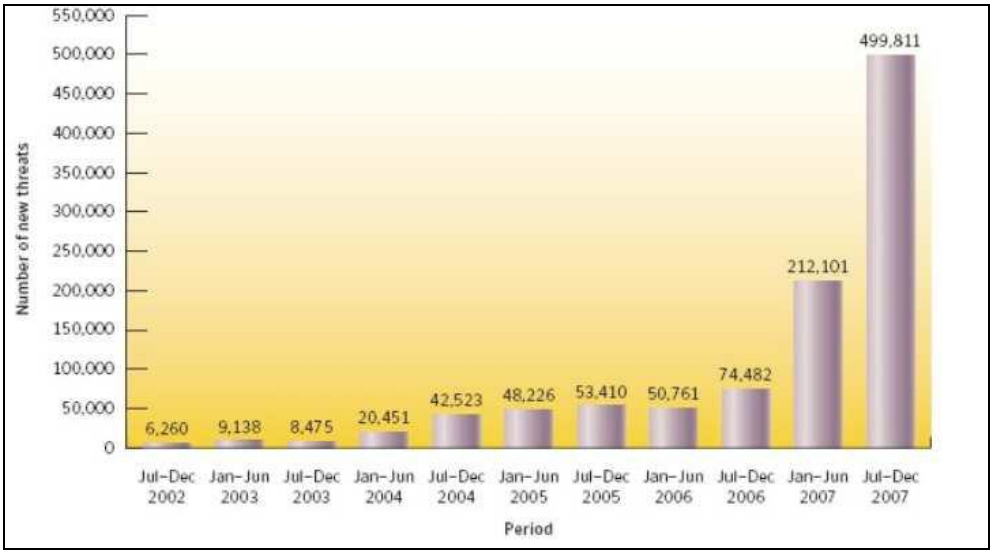

Figure 1.1 New Malicious threats rise (Turner et al., 2008)

1.2 Research Background

Computer viruses generally refer to programs that unintentionally get into computers, interrupt the normal operation, and cause damage to data and programs. However, not all programs that cause damage are real viruses. For instance, worms and trojan horses are two types that are not viruses. A worm finds its way into system registries and spreads as an independent program, while the trojan horse is run without knowing by the user. Thus, a virus is best identified not by what it does, but by how it spreads and infects other programs.

Viruses can be either unharmed or destructive. If they are unharmed , they cause no real damage, but may produce harmless changes like disrupting the display on a monitor, or making some sounds or false alarms. However, if they are destructive, the viruses can cause real damage to the system such as hogging disk space or main memory, using up CPU processing time and introducing the risk of less performances and conflicts. Over 40,000 different viruses have been identified so far. In recent years, the number of viruses unleashed has increased dramatically and the extent of the damage (in lost data, time and productivity) is estimated to be several billions of US dollars per year (Subramanya and Lakshminarasimhan , 2001).

As being discussed above, computer viruses and other malware have been in existence from the very early days of the personal computer and continue to pose a threat to home and enterprise users alike. As anti-virus technologies evolved to combat these viruses, the virus writers too changed their tactics and mode of operation to create more complex and harder to detect viruses and the game of cat and mouse continued.

Both viruses and virus detectors have gone through several generations of change since the first appearance of viruses. Basically, there are five methods of virus detection that have been used by existing anti-virus software: signatures based scanning, check summing, real time scanning, virtual machine simulation, and heuristics. The most popular virus detection technique employed today is signature

based static detection, which involves looking for a fingerprint-like sequence of bits (extracted from a known sample of the virus) in the suspect file.

One kind of virus which is generally believed to be difficult to detect is a metamorphic virus and this thesis is particularly concerned with a recent stage in virus evolution metamorphic viruses. These are viruses which employ code obfuscation techniques to hide and mutate their appearance in host programs as a means to avoid detection. They have the ability to change its internal structure ensuring any two instances of the same virus will likely be different from each other, even though they will behave the same. Metamorphic viruses are quite potent against this technique since they can create variants of themselves by code-morphing and the morphed variants do not necessarily have a common signature. Detecting metamorphic viruses is challenging. The problem with simple signature-based scanning is that even small changes in the viral code may cause a scanner to fail. In addition, the signature database requires constant updates to detect newly morphed variants.

People use a variety of methods to prevent, detect, and eliminate computer viruses to safeguard their computer information systems (Kephart et al.,1997). Unfortunately, the popularity of unknown and sophisticated virus like metamorphic makes analysis and defense increasingly difficult for these traditional anti-virus methods. Therefore, new approaches must be found to effectively detect or identify the unknown viruses.

1.3 Research Problem Statement

Attacks by different kinds of viruses lead to different types of problems. Lots of effort has been done to minimize these attacks, but as time goes on, it becomes an obvious fact that the attacking technology is always getting one step ahead than the preventing technology. The problems statements that lead to this topic proposal are as follows:

1. Computer virus attacks appear more frequent lately.
2. Lots of valuable information being stolen everyday because of computer virus attacks.
3. Harm to either a computer system's hosted data, functional performance, or networking throughput, when they are executed.
4. Current analysis detection that may not suitable to detect the evolved of today's virus.

1.4 Research Objectives

This analysis on viruses code will be done to fulfill several objectives. Stated below are all the objectives:

1. To study the metamorphic virus characteristics and its evolve.
2. To investigate the existing metamorphic virus detection methods.
3. To propose a virus profile for the existing metamorphic virus.
4. To conduct analysis that metamorphic viruses using obfuscation technique can bypass any signature-based detection.

1.5 Research Scopes

Project scope limits the analysis that will be conduct to specific types of malicious code and the platform for the application to runs on. The scope is as follows:

1. Analysis on metamorphic virus that running on Windows Xp platform.
2. Study on the existing detection method for the metamorphic viruses.
3. Analysis based on the method available to detect the present of the metamorphic virus.

REFERENCES

1. Wing Wong and Mark Stamp. *Hunting for metamorphic engines*. Journal in Computer Virology, 2(3):211{229, 2006.
2. Jean-Maries Borello and Ludovic Me. *Code Obfuscation Techniques for Metamorphic Viruses*. Feb 2008.
3. Kephart J. O, Chess D M and White S.R . *Computers and Epidemiology,* IEEE Spectrum. 1993.
4. Essam Al Daoud, Iqbal H. Jebril and Belal Zaqaibeh. *Computer Virus Strategies and Detection Methods*. Int. J. Open Problems Compt. Math, 2008
5. S.R.Subramanya and N. Lakshminarasimhan . *Computer viruses* . Potentials, IEEE. vol. 20, Issue 4, 2001, pp 16-19.
6. J. Kephart, G. Sorkin, D. Chess, et al . *Fighting computer viruses*. Scientific American, 1997.
7. P. Szor. *The Art of Computer Virus Research and Defense*. Addison-Wesley, 2005.
8. P. Szor, P. Ferrie. *Hunting for Metamorphic*. Symantec Security Response, 2002.
9. P. Szor. *Advanced code evolution techniques and computer virus generation toolkits*. March 2005 (b).
10. I.Muttik. *Silicon Implants*. Virus Bulletin, pp. 8-10, May 1997.
11. M. Stamp. *Information Security: Principles and Practice*. Wiley-Interscience August 2005.
12. P.Szor. *The Art of Computer Virus Defense and Research*. Symantec Press, 2005 (c).
13. Myles Jordan. *Anti-Virus Research - Dealing with Metamorphism*. Virus Bulletin, Oct. 2002.

14. Peter Szor and Peter Ferrie. *Hunting for metamorphic*. In Virus Bulletin Conference, September 2001.
15. A.Walenstein, R. Mathur, M.R. Chouchane and A. Lakhotia. *Normalizing Metamorphic Malware Using Term Rewriting*. Proc. Int'l Workshop on Source Code Analysis and Manipulation (SCAM), IEEE CS Press, Sept. 2006. Pages 75–84.
16. Anti-virus test center, University of Hamburg, Germany. *Profile of Phalcon/ Skism Mass Produced Code Generator*. January 1993.
17. Ferdinand Wagner, Ruedi Schmuki, Thomas Wagner, and Peter Wolstenholme. *Modeling Software with Finite State Machines: A Practical Approach.* Number 0-8493-8086-3. Taylor & Francis Group, LLC, 1edition, 2006.
18. Mohamed R. Chouchane and Arun Lakhotia. *Using engine signature to detect metamorphic malwar.*, In WORM '06: Proceedings of the 4th ACM workshop on Recurring malcode, pages 73{78, New York, NY, USA, 2006. ACM Press.
19. Wing Wong and Mark Stamp. *Hunting for metamorphic engines*. Journal in Computer Virology, 2(3):211{229, 2006.
20. Arun Lakhotia and Moinuddin Mohammed. *Imposing order on program statements to assist anti-virus scanners*. In WCRE '04: Proceedings of the 11th Working Conference on Reverse Engineering (WCRE'04), pages 161{170, Washington, DC, USA, 2004. IEEE Computer Society.
21. Ruo Ando, Nguyen Anh Quynh, and Yoshiyasu Takefuji. *Resolution based metamorphic computer virus detection using redundancy control strategy.* In WSEAS Conference, Tenerife, Canary Islands, Spain, December 2005.
22. P. Mishra. *A taxonomy of software uniqueness transformations*. master's thesis, San Jose State University, Dec. 2003.
23. Arun Lakhotia, Aditya Kapoor, and Eric Uday Kumar. *Are metamorphic computer viruses really invisible?* part 1. Virus Bulletin, pages 5-7, December 2004.
24. Arun Lakhotia and Prabhat K. Singh. *Challenges in getting 'formal' with viruses*.Virus Bulletin, pages 15-19, September 2003.
25. Turner,Symantec Global Internet Security Threat Report Trends for July–December 07, Volume XII, 2008.

26. Washingtonpost.com Staff Writer, "*A Short History of Computer Viruses and Attacks*", Feb. 2003.

27. Walenstein, R. Mathur, M. Chouchane R. Chouchane, and A. Lakhotia, "*The design space of metamorphic malware*," In Proceedings of the 2nd International Conference on Information Warfare, March 2007.

28. E. Konstantinou, "*Metamorphic Virus: Analysis and Detection*," January 2008.