

**AN ANTI VIRUS SCHEME USING DIGITAL SIGNATURE AND
ANOMALY DETECTION TECHNIQUES**

SURESH BABU SUBRAMANIAM

**A thesis submitted in fulfilment of the requirements for
the award of the degree of Master in Engineering (Electrical)**

**Faculty of Electrical Engineering
Universiti Teknologi Malaysia**

JUNE, 2003

ABSTRACT

Among all the computer security breaches, viruses are the most frequent and destructive. Current anti-virus solutions focus too much on virus recognition techniques, causing new viruses to escape detection. Thus, this work proposes an anti-virus scheme that simply defends the data in the computer regardless of the type and name of virus. The scheme comprises two layers of protection, where the first layer implements digital signature technique while the second layer implements anomaly detection technique. In the scheme, newly downloaded files that have been digitally signed using SHA-1 and RSA algorithms are verified at the first layer. Here the source and integrity of the files are determined and the executables with authentic and genuine signatures are accepted and logged into a watch list. At the second layer, the behaviour of the new executables; the ones in the watch list, are monitored closely at the lowest level for any anomalies. These anomalies are either blocked or ignored depending on the configurations set by user. One of the main ideas of the proposed scheme is to focus on new executables alone, as viruses originate only from newly downloaded files, either from email attachments, shared files and folders or new software installation. To realize the proposed scheme a prototype has been developed for Microsoft Windows 98. Meanwhile, to verify the functionality of the prototype, a test program that simulates most of the virus behaviour is also devised. Test results have proven that the proposed scheme can offer users the desired protection against all kinds of malicious programs.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	DECLARATION	ii
	DEDICATION	iii
	ACKNOWLEDGEMENTS	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENTS	vii
	LIST OF TABLES	xii
	LIST OF FIGURES	xiii
	LIST OF ACRONYMS	xv
	LIST OF APPENDICES	xvi

PART ONE

THESIS CONTENT

CHAPTER 1	INTRODUCTION	1
	1.1 Background	1
	1.2 Problem Statement	3
	1.3 Research Objectives	4
	1.4 Scope of Research	5
	1.5 Contributions of Research	6
	1.6 Thesis Outline	7
 CHAPTER II	 LITERATURE REVIEW	 9
	2.1 Viruses and Anti Viruses: Current Issues	 9
	2.2 Research based on Cryptography	11
	2.3 The Use of Neural Networks to Fight	12

	Viruses	
2.4	Research on Automatic Signature Extraction	13
2.5.	Current Commercial Anti-Virus Techniques	13
2.5.1	Signature Scanning	14
	2.5.1.1 Trend Micro PcCillin	15
2.5.2	Heuristic Scanning	16
2.5.3	CRC Scanning	17
2.5.4	Behaviour Blockers	18
	2.5.4.1 Achilles Shield	19
	2.5.4.2 Finjan's SurfinSheild	19
2.5.5	Data Recovering	20
2.5.6	Digital Signature	20
	2.5.6.1 Microsoft Office 2000	21
2.6	Commercial Products Pertinent to the Proposed System: Lotus Notes	21
2.7	Summary	22

CHAPTER III	THE PROPOSED VIRUS PROTECTION - CONCEPTS AND METHODS	23
3.1	The Basic Idea and Concept of the Proposed Scheme	20
3.2	Anomaly Detection Technique	24
3.2.1	Viruses in Windows 98 Environment	26
3.2.2	Windows 98 General System Architecture	27
3.2.3	Windows 98 File and Registry System Architecture	27
3.2.4	Virtual Device Drivers	31
3.3	Digital Signature	32
3.4	Tools Utilized in this Work	34

3.5	The Issue of Configurability vs. User-Friendliness	35
3.6	Summary	36
CHAPTER IV	THE PROPOSED VIRUS PROTECTION SCHEME – DESIGN	37
4.1	Operational Flow	37
4.2	Structural View	38
4.3	Behavioural Description of the Proposed Protection System	40
4.4	Anomalies covered by the Proposed Protection Scheme	42
4.4.1	A Study on Virus Behaviours	42
4.4.2	The File and Registry Anomalies Covered by the Proposed Scheme	44
4.5	Summary	46
CHAPTER V	IMPLEMENTATION OF THE VIRUS PROTECTION SCHEME	47
5.1	Interfacing Low Level Components with the API	49
5.1.1	Loading and Unloading the VxDs	49
5.1.2	Sending and Receiving Data	50
5.2	Installing the File System API Hook	52
5.2.1	Dealing with File Names in the Hook	54
5.3	Installing the Registry Service Hook	56
5.3.1	Dealing with Registry	57

Paths

5.4	New Files Detector: Detecting New Executables	60
5.5	Logger: Logging New Executables and Extracting Processes	60
5.6	Digital Signature: Verifying and Signing Executables	63
5.7	Anomaly Detection: Monitoring and Filtering Activities	66
5.7.1	Monitoring and Filtering File Activities	66
5.7.1.1	Detecting Execution of New Executables and Processes	69
5.7.1.2	Opening of Address Book to Send Mail	70
5.7.1.3	Deletion and Modification of Files and Directories	71
5.7.1.4	Formatting and Writing Random Data to Hard Disk	73
5.7.1.5	Attempts to Install and IFS Hook	73
5.7.2	Monitoring Registry Activities	74
5.8	Summary	75

CHAPTER VI	THE PROTOTYPE – USER INTERFACING, TESTING AND BENCHMARKING	76
6.1	The GUI of the Prototype	76
6.2	Testing the Prototype	79
6.3	Benchmarking	82

6.3.1	Products Used for Comparison	83
6.3.2	The Environment	84
6.3.3	The Tools and Performance Tests	84
6.3.3	Benchmarking - Tests And Results	87
6.4	Summary	88
CHAPTER VII	CONCLUSIONS	90
7.1	Capabilities of the Proposed Protection Scheme	91
7.2	Limitations of the Proposed Protection Scheme	92
7.3	Recommendations	93
7.4	Concluding Remarks	94
	REFERENCES	96

PART TWO

APPENDICES

A	TYPES OF MALICIOUS CODE AND 10 MOST COMMON VIRUSES	100
B	LIST OF 90 ANALYZED VIRUSES	104
C	IOREQ STRUCTURE AND THE LOW LEVEL FILE AND REGISTRY REQUESTS	105
D	SPECIFIC TARGETS OF VIRUSES	110
E	VIRUS BEHAVIOURS	113
F	SKELETON FOR A DYNAMIC VXD	116
G	SOURCE CODES FOR THE PROTOTYPE	120

CHAPTER I

INTRODUCTION

1.1 Background

It has been more than 20 years since the emergence of the risks posed by viruses. The growth of malicious programs, generally known as viruses, has steadily increased from annoyance to a major security threat to all communities of Internet users and businesses. We cannot stop virus writers from creating viruses and neither can we fully avoid viruses unless we provide absolute isolation, which is not an option in this connected world.

The first computer virus problem was described in 1984 [1], by Dr Frederick Cohen, a renowned man in the virus field. Until today there is still no standard definition of a computer virus, despite many attempts to give one, as virtually all the features of a virus may be found in other non-viral programs, or there exist some viruses which are free from those features (except from their spreading capabilities). Dr Frederick Cohen defined viruses as follows:

"A computer virus is a program that can infect other programs by modifying them to include a possibly evolved version of itself"

However, in the context of this work, viruses are defined as any code or program that displays any undesired behaviour, including propagating. There are currently more than 60 thousand¹ viruses in wild. Appendix A describes in brief the

¹ The figure is obtained from reliable anti virus vendors web pages; i.e. Trend and Symantec

major types of malicious codes, which are classified according to their method of infection and technology used.

The worst cases of computer disasters are usually caused by viruses. The recent I Love You virus incident, for instance, which according to ICSA (International Computer Security Association) is, by far, the most expensive, pervasive and damaging virus in history has caused a loss of more than one billion dollars. According to ICSA Labs 7th annual Virus Prevalence survey (ICSA Labs, 2001), there have been nearly 1.2 million incidents involving destructive computer code on approximately 6666,327 machines during the 20 months from January 2000 through August 2001, and it is learnt that computer virus attacks have increased despite all the extra money companies have invested in protection tools [2]. According to a different survey by CSI (Computer Security Institute) and the FBI (Federal Bureau of Investigations) [3], despite that 90% of the surveyed companies used anti virus software, 85% were still hit by viruses and worms.

Viruses present four types of information security compromises; namely disclosure, unavailability, loss of integrity and repudiation of origin, which can eventually bring negative impacts to companies or individuals, such as loss of competitive advantage, operational disruption, denial of service, lower customer perception, legal transgression and direct financial loss [4], without taking into account emotional breakdowns and frustrations. The details on these threats can be found in Appendix A. Also included in the appendix is a list of ten most common viruses today.

The main reason for the widespread of viruses is the Internet. Any node in the world connected to the Internet is inevitably susceptible to virus attacks. Email applications, especially, have become the number one entry point for viruses, either through file attachments or by exploiting the bugs in the applications themselves. Today, it is known that statistically 1 in every 1500 emails will, on average, contain a virus [5]. Another entry point quite exploited by viruses is the web browser, which if not configured correctly will allow execution of harmful Java Applets and Active-X applications. Other Internet applications that contributes to the spreading of viruses are chat programs (e.g. MIRC) and Bulletin Board Software. With the Internet

applications seeing advancement in the future, more holes and entry points are foreseen.

1.2 Problem Statements

Numerous types of anti virus tools are available in the market from software scanners through behaviour blockers², CRC checkers, heuristic scanners and data recovering hardware. The most vastly used protection tool is the software scanner because of its capability to recognize and remove known viruses. One of its major drawbacks often highlighted is its inability to detect new viruses as they contain new signatures that are not in the scanners' database. By the time their signatures are extracted and distributed, the new viruses would have already propagated widely and accomplished their intended damage. In spite of this, users still prefer scanners to other virus protection tools, as they get comfort knowing the name and type of virus detected in their computers. Another problem with virus scanners is the increasing size of its signature database. As the number of viruses increases exponentially every year, it would be infeasible to store all the signatures in its database.

Most of the other protection tools available are very inconvenient as they generate a lot of false positives; where non-viral behaviour or programs recognized as malicious. Additionally, anti virus vendors tend to make their products as user-friendly as possible for the purpose of commercialization, despite the common tradeoff that stands between user-friendliness and user-configurability; i.e. as the more user-friendly a product is, the less configurable it would be, resulting in the user not being able to tune the product to the exact desired protection level.

Furthermore, new viruses with new techniques emerge almost every month rendering them elusive to current anti virus tools. Even though scanners are able to recognize and remove viruses they are totally ineffectual when it comes to new viruses, which cause the virus incidents. No matter what new algorithm is devised to fight viruses, virus writers will always surface with new counter algorithms; hence it

² Anomaly detection technique is similar to behaviour blocking

would be more appropriate now for anti virus developers to try to protect the data instead of identifying and removing the viruses

1.3 Research Objectives

Motivated by the problems stated above, this research work proposes an anti-virus scheme that incorporates two layers of protection to protect the data regardless of the origin of the attack (viral or non-viral). Two common data security techniques are used here; where the first layer employs digital signature technique to authenticate incoming executables meanwhile the second layer uses anomaly detection technique to block malicious behaviour by the new executables in the system.

The main objective of this research project is to devise a virus protection scheme that can protect data from all kinds of malicious programs; known or unknown. The protection scheme, which is intended to act as a supplementary or a replacement to signature scanners that miss new viruses, is developed based on the following specifications:-

- i. The protection scheme comprises two layers of protection, where the first layer delimits the number of potential viruses by blocking downloading of executables that are not authentic; while the second layer blocks any malicious attempts by the newly downloaded executables towards the file system and the registry.
- ii. The first layer of the scheme utilizes digital signature technique to authenticate the origin and integrity of downloaded executables.
- iii. The second layer employs anomaly detection technique to monitor all file and registry requests made by newly executables to block any malicious attempts towards the file system and the registry.

- iv. In order to diminish the high number of false positives that are usually produced by products using anomaly detection techniques, the second layer focuses on newly downloaded executables alone.
- v. The proposed scheme can stop all viruses that have direct impact towards the file system and the registry. Nonetheless, the proposed idea of protection here is rather general, where the anomaly detection technique can be applied to prevent all types of viruses.
- vi. To validate the viability of the proposed methods, a fully functional prototype is devised for a specific platform. Meanwhile, to test the functionality of the prototype a virus behaviour simulator is developed.

1.4 Scope of Research

The end result of this research project is a software prototype anti-virus scheme, which is designed according to the following specifications:-

1. The proposed anti virus scheme should stop new viruses from causing any harm to the file system or the registry. The proposed method *cannot recognize or remove* previously known or unknown viruses. With the anti virus active, the data in the computer system will stay unharmed, regardless of the existence of viruses in the computer. No scanning technique has been employed, thus no signature updating is required.
2. The prototype for the proposed scheme is designed for personal computers running on Windows 98™ and compatible operating systems. Operating systems like Windows NT™, Windows 2000™ and UNIX™ cannot support the system as they have dissimilar underlying file architectures.
3. The digital signature scheme applied here is the RSA (with SHA-1 hashing algorithm) standard, thus any signatures produced using any other schemes

cannot be verified by the system. The public key management problem for public key cryptography is not part of the scope of this work, thus will not be elucidated

1.5 Contributions of Research

The proposed idea of integrating both digital signature and anomaly detection techniques is the main contribution of this research. The closest work found to be similar to the proposed idea was introduced in Lotus Notes [6] where, basically, access controls to its databases are set according to the digital signatures; i.e. trusted person gets higher privilege of access and so on.

Furthermore, unlike other anti viruses, the proposed system emphasizes on configurability in both the protection layers where a user can tune the system to their desired protection level. Other contributions are listed below:-

1. An original approach to monitor and filter malicious behaviours has been implemented in the anomaly detection layer of the proposed scheme. Anti-virus vendors would not disclose the approaches they use and detailed information is hardly available in the World Wide Web and in published materials.
2. A virus behaviour analysis has been performed on a number of 90 current and old viruses (up to date till the end of this research). From this study, virus behaviours have been extracted and classified into general groups; as can be found later in this documentation. No official documents or publications that give such study have been found. The list of the 90 sampled viruses can be found in Appendix B.
3. A fully functional prototype has been developed for personal computers running Windows 98™ and other compatible operating systems. To test the

prototype a virus behaviour simulator program is also created

4. The high number of false positives usually produced by anti-virus programs with anomaly detection techniques is reduced hugely, as only new executables are monitored for malicious behaviours. However, the level of reduction cannot be quantified as it depends on the number of executables resident in the computer and the number of newly downloaded executables.
5. The basic idea of initially filtering out potential viruses through authenticity before filtering the behaviours of the accepted executables is rather general and can be applied to all operating systems.

1.6 Thesis Outline

This thesis is organized in 7 chapters. The first chapter is the introductory chapter that puts in the picture the motivation, scope and objectives of this work. Also included is the outline of the thesis itself.

Chapter II summarizes the literature that has been reviewed throughout the research work, which include the issues on viruses and anti viruses, current related works and commercial products pertinent to the one proposed here.

The next chapter; Chapter III, discusses generally about the method, concepts and algorithms that are used to realize the proposed scheme. These include the digital signature algorithms and the anomaly detection techniques implemented in each of the layers. The tools that are used to accommodate implementation are also revealed in this chapter.

Chapter IV serves to give an overview of the proposed system, by giving the basic approach used to protect the data from viruses, the pictorial view and a general view of what the protection system offers. The results of the virus study are revealed here.

Chapter V is about the implementation of the proposed system. This chapter explains how each task and the functional modules of each of the layer in the system are implemented. Many flow charts and pseudo codes have been provided to give a better understanding of the implementation.

The end result of this work is a fully functional software prototype. The user interfacing, testing and benchmarking of the developed prototype is delineated in Chapter VI. In this chapter, the prototype is compared with some other commercial products in terms of some standard performance evaluation tests; disk access speed, memory usage and CPU usage.

The final chapter summarizes the research findings, the capabilities and the limitations of the proposed scheme and gives some recommendations for the future expansion of the work. This chapter precedes the list of references used throughout this work.

There are 7 appendices given at the end of this thesis. Appendix A describes the major types of malicious codes and lists the 10 most common viruses today. Appendix B lists the 90 viruses that have been used to extract virus behaviours through the virus behaviours study. Meanwhile, Appendix C gives the details of the IOREQ structure; a structure that contains the necessary parameters for a file request. The results of the virus behaviours study are revealed in Appendices D and E; where Appendix D delineates the specific targets of viruses meanwhile Appendix E lists the specific virus behaviours. In Appendix F, a skeleton code for a dynamically loaded VxD is given. This is to aid the understanding of VxDs; which are used to implement the anomaly detection layer of the proposed scheme. Finally, the source codes for the software prototype devised for the proposed scheme are given in Appendix G.

REFERENCES

1. Cohen F. (1991). "Current Best practices against computer viruses" IEEE International Carnahan Conference on, 261- 270
2. Bridwell L. and Tippet P. (2001). "7th Annual Computer Virus Prevalence Survey ICSA Labs." ICSA Labs
3. Power R. (2002) "CSI/FBI Computer Crime and Security Survey" Computer Security Institute. USA..
4. Flynn H. and Smith C. (2001). "Maintaining Integrity and catching viruses: Effective malicious code management", Trend Micro, Tokyo
5. Sunner M. (2000). "ISP Virus Scanning – The Why and How"; Virus Bulletin Conference, Orlando, U.S
6. Overton M. (1999) "Viruses and Lotus Notes: - Have the Virus Writers Finally Met Their Match"; Virus Bulletin Conference, Vancouver, Canada 149-174
7. Cass S. (2001). "Anatomy of Malice" IEEE Spectrum
8. Chess, D.M and White, S.R. (2000): "An undetectable Computer virus"; Virus Bulletin Conference, Orlando, U.S
9. Cohen. F. (1987). "Computer Viruses, Theory and Experiments" Computers and Security, 6 22—35

10. Cohen F. (1987). "A Cryptographic Checksum for Integrating Protection";
Computers and Security, 6 505-510
11. Davida, G., Desmett, Y. and Matt, B. (1989),: "Defending Systems against Viruses
through Cryptographic Authentication"; IEEE Symposium, 312- 318
12. Bellare M., Goldreich O and Goldwasser S. (1995). "Incremental Cryptography and
Application to Virus Protection"; Proceedings of the 27th CAN Symposium on
the Theory of Computing, ACM Press, 45-56
13. Whe-Dar LIN and Jinn-ke JAN (2000) "An Automatic Signature Scheme using a
Compiler in Distributed System." IECE/IEEE Joint Special Issue on
Autonomous Decentralized Systems
14. Tesauro, G., Kephart, J.O. and Sorkin, G.B (1996). "Neural Networks for Computer
Virus Recognition", IEEE Expert, 11, 4
15. Luke, J. (1999). "The Application of CMAC based Intelligent Agents in the
Detection of Previously Unseen Computer Viruses", IEEE International
Conference on Information Intelligence and Systems, Washington, 662-667
16. Kephart, J.O and Arnold, W.C. (1994). "Automatic Extraction of Computer Virus
Signatures"; 4th Virus Bulletin International Conference, Abingdon, England
178- 184
17. Kephart J. O., Sorkin G.B. Swimmer M. and White S.R. (1997). "Blueprint for a
Computer Immune System." Virus Bulletin International Conference. San
Francisco, California
18. R&D Computer System. (1999). " AVC 500 User Manual", R&D Computer System
Co. Bangkok. Thailand.

19. Calluna (1999). "PC Bodyguard User Guide" Calluna Technology Ltd, Scotland
20. Overton M. (1999). "Implementing Anti-Virus Controls in the Corporate Arena",
16th Compsec International Conference, London, U.K. 575-586
21. Kephart J.O.(1994). "A Biologically Inspired Immune System for Computers"
Artificial Life IV, Proceedings of the Fourth International Workshop on
Synthesis and Simulation of Living Systems, MIT Press, Cambridge,
Massachusetts, pp. 130-139
22. Symantec White Paper Series (1998) "Understanding Heuristics: Symantec's
Bloodhound Technology", USA: Symantec
23. Wismer K. (1999), "The Anti-Virus Cook Book". BockLabs, Wisconsin, US
24. Ducklin P. and Hannay P. (2000). "Microsoft Office 2000 and digital macro
signatures", Sophos White Papers, Sophos
25. Oney W. (1996). "Systems Programming for Windows 95". Washington. Microsoft
Press
26. Mitchell, S. (1997). "Windows 95 File System"; O'Reilly & Associates, Inc
27. Russinovich M. and Cogswell B. (1996) "Examining the Windows 95 Registry,"
Windows Developer's Journal
28. FIPS PUB 180-1. (1993). "Secure Hash Standard". USA. National Institute of
Standards and Technology

29. Rivest R., Shamir A. and Adleman L. (1978). "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" Communications of the ACM. 21:120-126
30. Posch, R. (2001). "Will Internet Be Secure?"; Journal of universal Computer Science, 7, 5 447-456
31. Messmer E. (2002). "Behavior Blocking Repels New Viruses", Network World Fusion
32. Microsoft (1998), "Microsft 98 DDK Documentation", Microsoft. USA.
- 33 NAI's PGP (1998). "PGP Software Developer's Kit Reference Guide" Network Associates Inc. Santa Clara.
34. Chess, D.M.and Morar, J.F. (2000). "Can cryptography prevent computer viruses?" Virus Bulletin Conference, Orlando, Florida
35. Hadzic, I., Udani, S. and Smith, J.M. (1999), "FPGA Viruses"; Field-Programmable Logic and Applications, 9th International Workshop, FPL'99, Glasgow, UK 291-300
36. Hypponen, M. (1994) "Retroviruses-How viruses fight back"; Virus Bulletin Conference, New Jersey, U.S.
37. Wang, Y. (2000). "Using Mobile Agent Results to Create Hard to Detect Computer Viruses", 16th IFIP SEC, Beijing, China 161-170.