ISSN 1673-9418　CODEN JKYTA8
Journal of Frontiers of Computer Science and Technology
1673-9418/2009/03(02)-0154-08
DOI:10.3778/j.issn.1673-9418.2009.02.004

E-mail:fcst@public2.bta.net.cn
http://www.ceaj.org
Tel:+86-10-51616056

Unknown Computer Virus Detection Inspired by Immunity*

ZHANG Yu+, LI Tao, QIN Renchao
College of Computer Science, Sichuan University, Chengdu 610065, China
+ Corresponding author: E-mail: bullzhangyu@yahoo.com.cn

受免疫启发的未知病毒检测技术*

张　瑜+,李　涛,覃仁超
四川大学 计算机学院,成都 610065

摘　要: 受免疫原理在入侵检测系统中成功应用的启发，提出了一种基于免疫的检测未知病毒的通用检测技术。由于病毒需要重定位模块来访问自己的资源,而这在正常程序中不常见,故可利用重定位模块来生成检测未知病毒的检测器。分析了计算机病毒的逻辑结构,建立了自体和非自体的演化方程、抗原提呈及抗体生成方法。实验表明,该技术不仅可检测已知病毒,还能有效检测未知病毒,且有自适应和自学习能力。

关键词: 计算机免疫系统;PE 病毒检测;重定位;病毒库

文献标识码: A　　**中图分类号:** TP309

ZHANG Yu, LI Tao, QIN Renchao. Unknown computer virus detection inspired by immunity. Journal of Frontiers of Computer Science and Technology, 2009,3(2):154-161.

Abstract: A novel Windows PE virus detection approach is presented that draws inspiration from artificial immune system and the structure of the relocation module of the virus. The structure of Windows PE virus is sufficiently analyzed. The dynamic evolution of self and nonself, the presentation of the antigen, and the generation of the antibody are proposed. The experiment is conducted and its results indicate that this approach not only has relatively higher detection rate of unknown Windows PE virus than the earlier known methods, but also has better capability of self-adaptive and self-learning.

Key words: computer immune system; PE virus detection; relocation module; virus gene pool

* The National Natural Science Foundation of China under Grant No.60573130 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z435 (国家高技术研究发展计划(863)); the New Century Excellent Expert Program of Ministry of Education of China under Grant No.NCET-04-0870 (国家教育部新世纪优秀人才计划).

Received 2008-11, Accepted 2009-01.

1 Introduction

The ever-increasing computer viruses have caused huge economic losses since the advent of computer viruses[1-2]. Most signature-based antivirus products are effective to detect known viruses but not unknown viruses or viruses' variants, which make them often lag behind viruses. In other words, antivirus often takes remedial measures to recuperate damage caused by viruses, not preventive measures to control viruses before they occur, and not strictly effective measures to block viruses' propagation during their epidemic.

Since most computer viruses are platform-dependent, they can operate only on a single operating system. With the operating system transforming from DOS to Windows, most previously DOS computer viruses cannot work in the new environment and Windows viruses including Windows macro viruses, Windows script viruses, and Windows PE viruses are becoming more and more popular. However, Windows macro viruses gradually decrease with the security enhancements of Microsoft's Office suite, and so do Windows script viruses because of the security enhancements of IE browser. Unlike them, the ever-growing PE (portable executable) viruses are easy to propagate between different platforms and are difficult to detect by antivirus because of their portable file format. In addition, PE viruses have become the favorite target of most virus writers who exhibit their techniques in the virus community. All these actions led to the development and upgrade of PE viruses, which make the antivirus more and more difficult to detect them, let alone to remove them.

As for Windows PE virus detection, antivirus researchers put forth different approaches in literature. Xu et al.[3] proposed an API sequence based scanner for polymorphic malicious executable. This approach rests on an analysis based on the Windows API calling sequence that reflects the behavior of a piece of particular code. Although it could achieve good experiment results, the construction of the API calling sequences and the similarity measurement between the two sequences take too much computational time. Reddy et al.[4] presented an n-gram based computer virus detection, which combined several classifiers using Dempster Shafer theory for better classification accuracy. But the training time is too much to apply to the antivirus applications. Tesauro et al.[5] developed a neural networks based method for computer virus recognition, which they deployed the neural network as a commercial product in IBM. Zhang et al.[6] proposed a Bayesian theory based method for unknown computer virus detection, which used the difference between the normal programs and suspicious programs to probably recognize unknown viruses. Wang et al.[7] proposed a support vector machines based approach for unknown virus detection. Zhang et al.[8] proposed a k-nearest neighbor algorithm based method for unknown virus detection. Chen et al.[9] presented a program behavior based method for unknown virus detection. Schultz et al.[10] proposed a data mining based approach for new virus detection. From the technical point of view, the approaches mentioned above are complex for two reasons. First, lots of malicious and benign codes as training dataset are difficult to collect. Second, they would consume lots of times when training the classifiers.

To improve the performance of the detector mentioned above and to effectively detect Windows PE viruses, a novel immune based approach for unknown Windows PE virus detection is proposed. Experiments are conducted and results show that this approach has better efficiency in the detection of known and unknown Windows PE viruses than the others.

In the following, we first describe the logical structure of Windows PE virus in section 2. Then introduce the theory of our detection model which includes the detection process, the evolution of self and nonself, the antigen presenting and the generation of the antibody in section 3. Section 4 shows the implementation and experiment results. We state our conclusion in section 5.

2 The Logical Structure of Windows PE Virus

PE (portable executable) is the native file format of Win32 and its specification is derived somewhat from the Unix COFF (common object file format). The meaning of "portable executable" is that the file format is universal across Win32 platform: the PE loader of every Win32 platform recognizes and uses this file format even when Windows is running on CPU platforms other than Intel. Windows PE virus takes advantages of the PE file format to spread themselves among different Win32 platforms. Generally speaking, a Windows PE virus must include the following modules[11] to better infect other host programs, namely, the relocation module, the module of obtaining API address, the module of searching target files, the module of mapping file to the memory, the module of adding new section to infected files, and the module of returning to the target file. The logical structure of Windo ws PE virus is shown in Fig.1. We will briefly introduce them below.

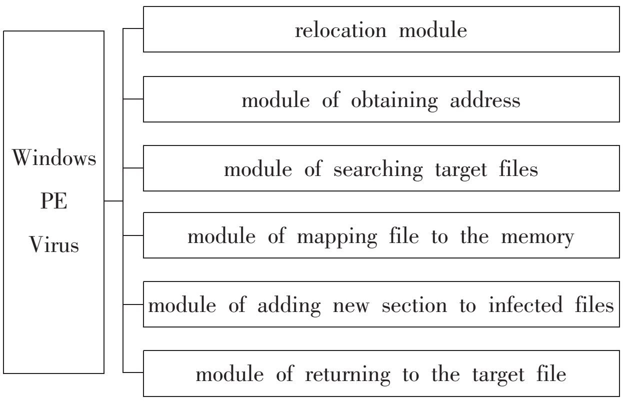
Windows
PE
Virus
relocation module
module of obtaining address
module of searching target files
module of mapping file to the memory
module of adding new section to infected files
module of returning to the target file

Fig.1 The logical structure of Windows PE virus

图1 Windows 病毒的逻辑结构

2.1 The Relocation Module

Normal programs do not concern the location of variables or constants, because their locations in the memory are well calculated when compiled by the compiler program. Therefore, when programs are loaded into the memory, they do not need to relocate the position of variables or constants used in them. The variables or constants are directly used by their names. Similarly, the virus programs also use variables and constants. But the locations of virus variables or constants vary with the infected host programs, because of their attachment to different host programs resulting in different positions of the virus variables or constants when loaded in the memory with the host programs. Since these variables or constants do not have fixed addresses, the virus must rely on itself to relocate these addresses to normally access to the relevant resources when executed in the memory. Therefore, the Windows PE virus must have an inherent relocation module, which is usually at the beginning of the virus program with less code and little change, so as to be correctly executed in the Windows platform.In this study, we extract the relocation module as a gene from the virus to produce antibodies to detect unknown Windows PE virus.

2.2 The Module of Obtaining API Address

Windows programs generally run in Ring 3, the protection mode in the Windows operating system. The system API calls achieve through the dynamic link library in the Windows.Generally, normal programs have an import address table, inside which the actual addresses of API functions are. Thus, when being called by the program, the corresponding API functions addresses can be found in the import address table of the Windows PE file.

However, the Windows PE virus has only a code section, which does not include the import address table so as to reduce the virus source code. Unlike the normal programs, the Windows PE virus program can not directly obtain the address of API functions, and must firstly identify these addresses in dynamic link library. Therefore, the Windows PE virus must have such module that can obtain the addresses of Windows API functions called by the virus.

2.3 The Module of Searching Target Files

In order to spread themselves, a virus must have to continuously search target documents to implement its infection to expand its influence.Therefore, the Windows PE virus needs a target files searching module.

2.4 The Module of Mapping File to the Memory

Memory-mapping-file provides a group of independent functions that are the association of a file's contents with a portion of the virtual address space of a process. Processes read from and write to the file view using pointers, just as they would with dynamically allocated memory. The use of file mapping improves efficiency because the file resides on disk, but the file view resides in memory. In this way, the computer virus can quickly infect target files to reduce the possession of system resources. Therefore, the Windows PE virus generally has a memory-mapping-file module.

2.5 The Module of Adding New Section to Infected Files

The most effective way to infect target files for the Windows PE virus is to add a new section to host programs. While adding a new section to the target file, the virus must modify the start code in the place of addressOfEntryPoint so as to firstly execute the virus code. Therefore, the Windows PE virus generally has the module of adding new section to infected files.

2.6 The Module of Returning to the Target File

In order to improve the viability, the virus should not destroy the infected target files. When infecting the target files, the virus should preserve the original value of AddressOfEntryPoint. After the execution, the virus should jump back to the original value of AddressOfEntryPoint to hand over control to the target files. Therefore, the Windows PE virus generally has the module of returning to the target file.

3 The Immune-Based Detection Theory

The main function of the immune system is how to distinguish between self and nonself [12], and thereafter protect self and kill nonself. Similarly, the main function of the computer virus detection system is how to discriminate viruses from benign programs. Given the similarity of the two systems, the computer virus detection system can draw inspirations from the immune system, so as to achieve better detection efficiency. We will elaborately introduce the immune-based approach below.

3.1 The Detection Process

The detection process of our approach consists of the following steps. Firstly, the relocation modules are extracted as virus genes from the viruses by antigen-presenting. Secondly, the qualified antibodies are generated from the virus genes through negative selection. Finally, the generated antibodies are used to detect the known and previously unknown viruses. The logic detection processes are shown in Fig.2.

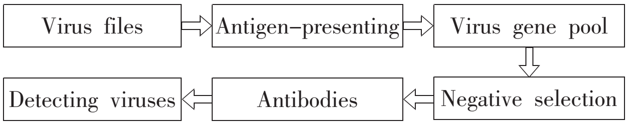

Fig.2　The detection process

图 2　病毒检测流程

3.2 The Dynamic Evolution of Self and Nonself

In this study, Self is defined as the protected files, and Nonself suspicious files. Suppose AG is a question domain, that is, $AG=\bigcup_{i=1}^{\infty} H^{i}$, and H=(0,1,2, ⋯,9,

A, B, C, D, E, F), a hexadecimal number set, i is the positive integer. *Self* and *Nonself* satisfy the following conditions, respectively, that is, $Self \subset AG$, $Nonself \subset AG$, $Self \cup Nonself = AG$, $Self \cap Nonself = \emptyset$.

In the immune systems, the self and nonself are dynamic changing with the interaction between them. Similarly, the self and nonself in the computer immune systems change with the infection and the effect. In other words, a self becomes a nonself with the infection, and a nonself becomes a self with the repair. The dynamic evolutional equations of *Self* and *Nonself* are as follows.

$$Self(t)=\begin{cases} S_{initial}, & t=0 \\ Self(t-1)-Self_{del}(t)+Self_{new}(t), & t\geqslant 1 \end{cases} \tag{1}$$

$$Self_{del}(t)=\{s|s\in Self(t-1)\wedge \exists y\in SA(t-1)\wedge <s,y>\in Match\} \tag{2}$$

$$Self_{new}(t)=\{s|s\in Nonself(t-1)\wedge \forall y\in SA(t-1)\wedge <s,y>\notin Match\} \tag{3}$$

$$Nonself(t)= \begin{cases} NS_{initial}, & t=0 \\ Nonself(t-1)-Nonself_{del}(t)+Nonself_{new}(t), & t\geqslant 1 \end{cases} \tag{4}$$

$$Nonself_{del}(t)=Self_{new}(t) \tag{5}$$

$$Nonself_{new}(t)=Self_{del}(t) \tag{6}$$

Where

$S_{initial}$ is the *Self* initial set that is composed of 500 normal Windows system files;

$Self_{del}$ is the *Nonself* set that has removed from the *Self* set;

$Self_{new}$ is the new generated *Self* set that is previously in the *Nonself* set, but is removed from it because of their non-matching with any antibodies;

$NS_{initial}$ is the *Nonself* initial set that is suspiciously infected by the viruses.

3.3 The Antigen-Presenting

The antigen-presenting is the process that extracts viruses' genes (virus relocation modules) from the suspicious files to form virus gene pool. The reasons why we choose relocation modules as virus genes are as follows. Firstly, the Windows PE virus relocation module is usually at the beginning of virus source code, and always small and little changed code easy to extract. Secondly, the other modules in the Windows PE virus such as the module of obtaining the API address, the module of searching target files and the module of memory-mapping file are also used in the normal programs. Thirdly, the module of adding a new section the infected files is extremely complex and difficult to analyze, though it is generally not found in the normal programs. The normally relocation module of the virus is shown in Fig.3. We extract the hex codes E8000000005B81EB2E1F4000 as the virus gene by anti-presenting.

The virus gene pool through the antigen-presenting is defined as follows.

$$V=\{v|v\in \bigcup_{i=8}^{32} H^{i} \wedge |v|=i\wedge v\}=Ap(x\in Nonself) \tag{7}$$

where

$Ap(x)$ is the antigen-presenting function;

v is the virus gene extracted from the virus relocation module, whose length is between eight and thirty-two hexadecimal codes.

```
.text: 00401F29   E8 00 00 00 00        call     $+5
.text: 00401F2E   loc_401F2E:
.text: 00401F2E   5B                    pop      ebx
.text: 00401F2F   81 EB 2E 1F 40 00     sub      ebx,offset loc_401F2E
```

Fig.3 The relocation module of the virus

图3 计算机病毒的重定位模块

3.4 The Generation of the Antibody

The acquired immune system that can protect against the specific viruses is generally acquired through vaccination to generate specific antibodies. The antibodies in this study are generated from the extraction of the vaccines in the virus gene pool, which are the virus genes obtained by antigen-presenting. Then, the detectors are generated from the antibodies to detect the viruses. For example, we will generate the detector E8000000005B from the virus gene pool. The detector set is defined as follows.

$$D=\{<d,affinity>|d\in\bigcup_{i=8}^{32}H^{i},affinity\in N\} \qquad (8)$$

Where

d is the antibody;

affinity is the match between the antigen and the antibody.

3.5 The Detection of Windows PE Virus

After the detectors are generated, they can effectively detect Windows PE viruses. During the detection, the antigen whose affinity with the antibody is larger than the threshold value will be regarded as a virus. The dynamic evolutional equations of the antibodies and the detection of Windows PE viruses are as follows.

$$SA(t)=\begin{cases}\emptyset,\ t=0\\ SA(t-1)+SA_{new}(t),t\geqslant 1\end{cases} \qquad (9)$$

$$SA_{new}(t)=\{v|v\in D,\ \forall y\in Self,<v,y>\notin Match\wedge v.affinity\geqslant\beta\} \qquad (10)$$

$$Match=\{<v,y>|v\in D,y\in AG,f_{match}(v.d,AP(y))=1\} \qquad (11)$$

$$f_{match}(v,y)=\begin{cases}1,\ f_{affinity}(v,y)/L_{v}\geqslant\alpha\\ 0,\ \text{otherwise}\end{cases} \qquad (12)$$

$$f_{affinity}(v,y)=\max(x_{1},x_{2},\cdots,x_{|L_{v}-L_{y}|+1}) \qquad (13)$$

$$x_{i}=\sum_{i=1}^{\min(L_{v},L_{y})}\theta_{ij} \qquad (14)$$

$$\theta_{ij}=\begin{cases}1,\ v_{i}=y_{i+j-1},\ 1\leqslant i\leqslant|L_{v}-L_{y}|+1,\ 1\leqslant j\leqslant L_{v}\\ 0,\ \text{otherwise}\end{cases} \qquad (15)$$

Where

SA is the specific antibody set;

SA_{new} is the new generated antibody set, whose affinity with self is greater than the threshold value β;

f_{match} is the matching function between antibody and antigen;

$f_{affinity}$ is the affinity function between antibody and antigen;

Match is the set consisting of the antigens and the antibodies matched by the antigens;

θ_{ij} is the affinity value; If there is a matching between an antibody and an antigen, the value is 1, otherwise 0;

L_{v} is the vaccine length;

L_{y} is the antigen length.

4 Experiment and Results

The experiment was conducted in the Computer Virus and Anti-virus Laboratory, Computer Network and Information Security Institute of Sichuan University. Since there is no benchmark data set available for the detection of computer viruses unlike intrusion detection, the data sets including 100 viruses and 500 benign executables were collected from the website VX Heavens[13] and from system32 folder in Windows, respectively. The main goal of the experiment is to test the detection rate of known and unknown viruses and false-positive rate of the normal files. The experimental results are shown in Fig.4.

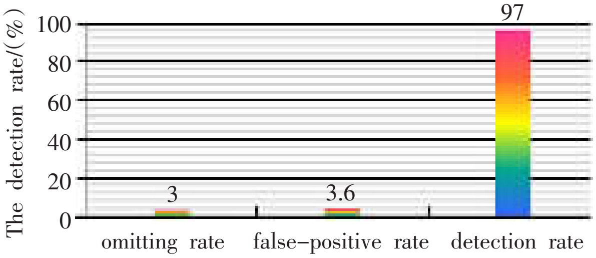

Fig.4 The experiment results of our approach

图 4　病毒实验结果

Fig.4 shows that the detection rate of the Windows PE viruses is 97%, the omitting rate is 3%, and the false-positive rate (misidentification of legitimate programs as viruses) is only 3.6%. The good experimental result can own to the detectors generated from the relocation module that is the indispensable part of Windows PE viruses, which make them accurately detect Windows PE viruses with higher detection rate. The detectors can detect most of Windows PE viruses except the shelled and encrypted ones. Since the shelled and encrypted viruses are protected by the shell and the inner binary codes including the relocation part are disturbed with encryption, the detection of them will result in the omitting rate. Meanwhile, some legitimate system programs comprise the relocation module that in turn leads to false-positive rate.

In order to test the performance of the proposed approach, we conducted the related comparison experiments with the currently most mature antivirus technologies including the Kingsoft 2008, Panda 2008, KV 2008, Eset NOD32 and Kaspersky 7.0.

The comparison experiments results are shown in Fig.5 that the detection rate of our proposed approach is 97%, Eset NOD32 is 94%, Kaspersky 7.0 is 88%, Panda 2008 is 67%, Jiangmin KV2008 is 55% and Kingsoft 2008 is 44%. Since most antivirus technologies are signature-based, they can only detect known computer viruses, and need to be updated frequently for its effectiveness. If their signature databases do not include a previously unknown virus signature, they cannot detect it. As a result, they have the higher detection rate of known viruses but lower detection rate of previously unknown viruses. Unlike them, our approach is non-signature-based technology, which can detect known and previously unknown viruses. The results indicate that the proposed approach has higher detection rate than the others, and efficiently testify the validity of our proposed approach.

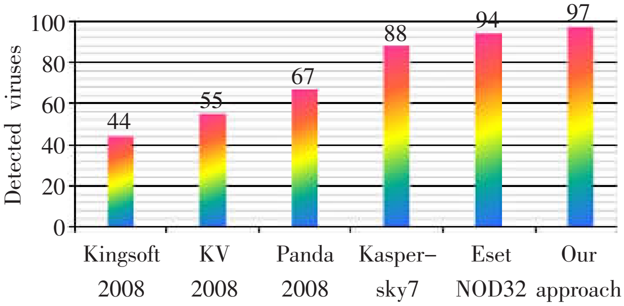

Fig.5 The comparison experiments results

图 5 病毒检测对比实验结果

5 Conclusions and Future Work

The Windows PE viruses have become an intriguing target of most virus writers, which leads to the continuously upgrade of the Windows PE viruses. At the same time, the currently antivirus is impossible to effectively detect unknown Windows PE viruses for most of them are signature-based. We draw inspirations from the immune system and the relocation module of the virus source code, and propose the immune based approach for detection of the Windows PE viruses. The experimental results show that the proposed approach which is non-signature-based not only has a high detection rate, low false-positive rate and low omitting rate, but also its efficiency is better than the currently mature antivirus products.

The future work will improve the detection to take into account shelled information, and extend the analysis to the DLLs and unknown worms, which are presently the most threat to the computer network systems.

References:

[1] Ford R, Spafford E H. Happy birthday, dear viruses[J]. Science, 2007,317:210-211.

[2] Balthrop J, Forrest S, Newman M E J, et al. Technological networks and the spread of computer viruses[J]. Science, 2004, 304:527-529.

[3] Xu J Y, Sung A H, Chavez P. Polymorphic malicious executable scanner by API sequence analysis[C]//Fourth International Conference on Hybrid Intelligent Systems, 2004:378-383.

[4] Reddy D K S, Pujari A K. N-gram analysis for computer virus detection[J]. Journal in Computer Virology, 2006,2:231-239.

[5] Tesauro G J, Kephart J O, Sorkin G B. Neural networks for computer virus recognition[J]. IEEE Expert, 1996,11(4):5-6.

[6] Zhang Boyun, Yin Jianping, Gao Jingbo, et al. Unknown computer virus detection based on multi-naive Bayes algorithm[J]. Computer Engineering, 2006,32(10):18-21.

[7] Wang Shuo, Zhou Jiliu, Peng Bo. Unknown virus detection based on API sequence and support vector machine[J]. Journal of Computer Applications, 2007,27(8):1942-1943.

[8] Zhang Boyun, Yin Jianping, Zhang Dingxing, et al. Unknown computer virus detection based on K-nearest neighbor algorithm[J]. Computer Engineering and Applications, 2005,41(6): 7-10.

[9] Chen Yueling, Jia Xiaozhu. Computer viruses detection method based on program behavior[J]. Journal of Qingdao University: Natural Science Edition, 2006,19(2):61-65.

[10] Schultz M G, Eskin E, Zadok E. Data mining methods for detection of new malicious executables[J]. IEEE Symposium on Security and Privacy, 2001.

[11] Guojpeng/CVC.GB. The analysis of Win32 PE viruses[EB/OL]. [2003]. http://www.hynubbs.cn/netstar/news_view.asp?id=61.

[12] Forrest S, Perelson A S. Self-nonself discrimination in a computer[J]. IEEE Symposium on Security and Privacy, 1994: 202-213.

[13] VX Heavens[EB/OL]. http://vx.netlux.org.

附中文参考文献:

[6] 张波云,殷建平,蒿敬波,等.基于多重朴素贝叶斯算法的未知病毒检测[J].计算机工程,2006,32(10):18-21.

[7] 王硕,周激流,彭博.基于API序列分析和支持向量机的未知病毒检测[J].计算机应用,2007,27(8):1942-1943.

[8] 张波云,殷建平,张鼎兴,等.基于K-最近邻算法的未知病毒检测[J].计算机工程与应用,2005,41(6):7-10.

[9] 陈月玲,贾小珠.基于程序行为的计算机病毒检测方法[J].青岛大学学报:自然科学版,2006,19(2):61-65.

ZHANG Yu was born in 1975. He is a Ph.D. candidate at Sichuan University. His research interests include network security, computational intelligence, etc.

张瑜(1975-),男,博士,讲师,主要研究方向为:网络安全,计算智能等。

LI Tao was born in 1965. He received the Ph.D. degree in Computer Science from University of Electronic Science and Technology of China in 1995. He is a professor and doctoral supervisor at Sichuan University. His research interests include network security, computational intelligence, intelligent information system, etc.

李涛(1965-),男,博士,教授,博士生导师,主要研究方向为:网络安全,计算智能,智能信息系统等。

QIN Renchao was born in 1978. He is a Ph.D. candidate at Sichuan University. His research interests include network security, computational intelligence, etc.

覃仁超(1978-),男,博士研究生,讲师,主要研究方向为:网络安全,计算智能等。