基于模糊识别和支持向量机的联合 Rootkit 动态检测技术研究

李 鹏[1,2,3],王汝传[1,2,3],高德华[1]

(1.南京邮电大学计算机学院,江苏南京 210003; 2.江苏省无线传感网高技术研究重点实验室,江苏南京 210003;
3.宽带无线通信与传感网技术教育部重点实验室,江苏南京 210003)

摘 要: 针对 Rootkit 恶意代码动态检测技术进行研究.总结出典型 Rootkit 恶意程序动态行为所调用的系统 API 函数.实时统计 API 调用序列生成元并形成特征向量,通过模糊隶属函数和模糊权向量,采用加权平均法得到模糊识别的评估结果;基于层次的多属性支持向量机分析法构建子任务;基于各个动态行为属性的汉明距离定位 Rootkit 的类型.提出的动态检测技术提高了自动检测 Rootkit 的准确率,也可以用于检测未知类型恶意代码.

关键词: 网络安全; 恶意代码; 模糊识别; 支持向量机; API 系统调用

中图分类号: TP393.08 **文献标识码:** A **文章编号:** 0372-2112 (2012) 01-0115-06

电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2012.01.019

Research on Rootkit Dynamic Detection Based on Fuzzy Pattern Recognition and Support Virtual Machine Technology

LI Peng[1,2,3], WANG Ru-chuan[1,2,3], GAO De-hua[1]

(1. *College of Computer, Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu* 210003, *China*;
2. *Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Jiangsu Province, Nanjing, Jiangsu* 210003, *China*;
3. *Key Lab of Broadband Wireless Communication and Sensor Network Technology* (*Nanjing University of Posts and Telecommunications*), *Ministry of Education, Jiangsu Province, Nanjing, Jiangsu* 210003, *China*)

Abstract: Dynamic detection technology of Rootkit malicious code has been studied. It summarizes typical dynamic system API functions which are called by Rootkit malicious codes. It extracts behavioural characters of the typical system API functional series accompany with the running of malicious code, forms feature vectors by counting up the generating elements important degree of system call series, uses fuzzy membership function and normalization fuzzy weights vector, and comes to the fuzzy pattern recognition conclusion with the use of weighted averaging method. It exactly locates the types of Rootkit malicious code based on the analysis method of layered multi-attributes support virtual machine, according to the subtasks constructed by the independent API system call behaviours, and with the calculation of hamming distance of dynamic behaviour properties. Experiments indicates the proposed dynamic detection method of combining fuzzy pattern recognition with support virtual machine technology not only improves the accuracy rate of Rootkit automatic detection but also has the ability of detecting the previous unknown type malicious code.

Key words: network security; malicious code; fuzzy pattern recognition; support virtual machine; application programming interface system call

1 引言

Rootkit 是攻击者在入侵系统后用来保持对系统的超级用户访问权限、创建后门和隐藏攻击痕迹等常采用的一种技术,能够窃取重要信息、提供隐蔽后门、隐藏攻击痕迹、欺骗检测工具、提供恶意代码植入手段,以及为进行其他非法活动提供跳板等功能,一直是危害比较大的恶意代码.

收稿日期:2011-04-19;修回日期:2011-07-18

基金项目:国家自然科学基金(No.60973139,No.61170065,No.61171053,No.61100199,No.60903181,No.61003039,No.61003236);江苏省科技支撑计划(工业)项目(No.BE2010197,No.BE2010198);省属高校自然科学研究重大项目(No.11KJA520001);江苏省高校自然科学基础研究项目(No.10KJB520013,No.10KJB520014);高校科研成果产业化推进工程项目(No.JH10-14);江苏高校科技创新计划项目(No.CX10B-196Z,No.CX10B-199Z);江苏省六大高峰人才项目(No.2008118);教育部高等学校博士学科点专项科研基金(No.20103223120007);江苏省计算机信息处理技术重点实验室基金(No.KJS1022)

近年来,Rootkit 恶意代码的发展非常迅速,并且朝着内核层发展.第 26 次《中国互联网络发展状况统计报告》指出,网络安全的问题仍然制约着中国网民深层次的网络应用发展[1,2].Black Hat 会议上,一名研究人员表示,采用 Rootkit 技术的黑客可以将恶意代码隐藏在计算机的 BIOS 闪存内,这一策略使得安全软件要发现、清除恶意软件变得更为困难[3].内核级 Rootkit 涉及到操作系统的底层内核,与应用级 Rootkit 相比,内核级 Rootkit 的破坏性更大,并且能够逃避任何应用层下的检测工具.

自动分析 Rootkit 恶意代码是当前的 Rootkit 检测技术中的难点,在分析现有的 Rootkit 检测技术的基础上,本文提出基于模糊识别和支持向量机联合的恶意代码动态检测方法对 Rootkit 进行检测,模糊识别用于初步判定可疑的检测程序,然后进一步使用支持向量机回归分析得出准确的恶意代码类型,对恶意代码的行为进行自动分类与共性特征分析并确认其行为和对系统的影响,能够大幅度提高防病毒软件的处理效率.

2 相关工作

针对 Rootkit 的检测比较困难,目前比较成熟的一些检测方法总结如下:Rootkit 签名检测法[4]先对已知的 Rootkit 进行分析,找出其特点,然后进行签名,Rootkit 检测工具可以利用一个签名对系统的关键区域进行模式匹配.如果存在已知的 Rootkit,这种检测是很快的,但对于未知的 Rootkit 无能为力.交叉视图检测法[5]对 Rootkit 的几个特定段通过获取任意两个点的信息,进行比较并判断两种信息是否一样,如果不一致认为遭到 Rootkit 的攻击,Sysinternal 公司的 RootkitReavealer 就是采用这个方法检测的.完整性检测法[6]针对文件感染型恶意 Rootkit 代码的检测技术,计算出系统标准文件的 Hash 值,放入安全的数据库,检测时将当前的文件与标准文件匹配,若匹配则为正常文件,否则断定系统被 Rootkit 攻击.虚拟机检测法[7]是一种新的 Rootkit 恶意代码检测手段,虚拟机检测技术可实现自动脱壳,虚拟机从文件入口点处一条一条的取指令执行,直至解密段指令执行完成,可以进行特征检测,其优点是可以有效检测加密变形的 Rootkit 攻击.

本文总结出各种的检测方法的特点,见表 1.基于以上分析,提出基于模糊识别和支持向量机的联合恶意代码动态检测技术,一方面可以提高检测的准确率,另一方面可以动态检测未知的恶意代码.

3 指标分析

同一类型的恶意代码及其变种因为开发平台、攻击系统平台、产生的攻击效果,以及实现模式存在相似性;恶意代码运行时,与正常程序相比较,具有其独特的程序行为,这些可以作为标识恶意代码的重要行为特征.动态特征包括以下几个方面:异常的文件访问、进程操作、注册表操作、系统服务和网络服务.

表 1 各种 Rootkit 检测算法的特点比较

	签名检测法	交叉视图检测法	完整性检测法	虚拟机检测法
执行环境	简单	简单	复杂	复杂
检测速度	快	快	较慢	慢
准确率	差	较差	较好	好
未知恶意代码检测能力	无	无	无	无
系统开销	小	小	较大	大

本文总结出典型 Rootkit 恶意程序动态行为所调用的系统 API 函数[8,9],如表 2 所示.正常的程序和恶意代码都需要调用这些 API,但是某一类型的恶意代码在调用这些 API 时存在特定的行为特征,本文通过对典型的 API 调用序列进行分析,力求客观反映出 Rootkit 恶意代码的动态特征.

表 2 恶意代码异常行为特征典型 API

属性	行为	API 函数
注册表	设置自启动项	RegCreateKeyEx, RegOpenKeyEx, RegQueryInfoKey
	修改文件关联	RegEnumKeyEx, RegEnumValue, RegSetValueEx
文件	修改系统配置	CreateFile, ReadFile, WriteFile, DeleteFile, MoveFile
	文件复制	CopyFile, CopyFileEx, WriteProcessMemory
进程	打开或结束进程	CreateProcess, OpenProcess, TerminateProcess
	远程线程注入	CteateRemoteThread
	键盘钩子	SetWindowsHookEx, LoadLibrary, GetProcAddress
系统服务	服务的设置	CreateService, DeleteService, OpenService
	服务的修改	ChangeServiceConfig, ControlService
网络	服务的设置	Socket, Listen, Setsockopt, Accept, Connect, Closesocket
	数据转换	ntohl, htonl, ntohs, htons
	数据传输	Send, Recv, Sendto, Recvfrom

4 检测方法和关键技术

4.1 检测方法思路

对系统调用序列进行匹配生成元扫描,统计函数库中的各个系统调用生成元的重要程度,根据生成元的距离得出可疑程序特定生成元的出现频率,采用加权平均法计算该待检测程序的估算结果 E,并给定判别阈值,对待检测程序进行判断.

对于判断出的可疑程序,进一步使用支持向量机

技术进行确认,采用基于层次的多属性支持向量机分析法,对量化的 API 系统调用序列进行属性分解,根据子任务求解每个属性的二次规划问题,最终依据多个属性动态行为属性的汉明距离,从而确认其恶意行为和恶意代码的所属类型.判定的结果作为 SVM 校正模型,提供给分析层作为以后的分析的修正.

4.2　特征提取

定义 1　系统调用序列.将恶意程序在执行时所调用的系统函数,按照调用时间顺序的排列集合 A.其子序列 S 为集合 A 中的删除若干元素后,组成的排列集合 $S=A_1A_2\cdots A_m$.

定义 2　生成元.若集合 $I=\{i_1,i_2,\cdots,i_p\}$,$1\leqslant i_1<i_2<\cdots\leqslant i_p\leqslant m$,使得 $S=A(I)$,其中 $A(I)=A_{i1}A_{i2}\cdots A_{ip}$,此时称 $A(I)$是系统调用序列的一个生成元,该生成元的距离为 i_p-i_1,记为 $D(I)$.各生成元在系统调用序列的重要程度随其距离增加而减少,选取参数 $\lambda\in(0,1)$,通过式(1)量化调用序列的重要程度.本文中,$\lambda=1/2$.

$$\phi_I=\sum_{I:S=A(I)}\lambda^{D(I)} \tag{1}$$

统计系统调用序列的各生成元,并且衡量每个生成元的重要程度,更能反映出系统调用的特征.将采集到的系统调用序列,按照每一类型系统调用序列取前 n 个生成元的统计调用排列,表示成识别向量,如式(2).

$$\boldsymbol{\phi}_S=(\phi_{S1},\phi_{S2},\phi_{S3}\cdots\phi_{Sn})^{\mathrm{T}} \tag{2}$$

所有的属性按照各自生成元分解成为 m 个不相交的子集 ϕ,如式(3),本文中 $m=5$,表 2 中所列出的五种属性.用于后面进行模糊识别[10]和支持向量机识别.

$$\phi=\bigcup_{S=1}^{m}\boldsymbol{\phi}_S \tag{3}$$

4.3　模糊识别

从实验的数据分析看,程序的 API 调用比较符合正态分布的概率,故本文选取正态分布模糊隶属函数 R.设某 API 系统调用生成元在待测文件中出现的频率统计值为 $\boldsymbol{\phi}_S$,构造程序系统调用序列的隶属函数如式(4)所示,i 为特征向量的个数.

$$R(\phi_{Si})=\begin{cases}0, & \phi_{Si}<0\\ 1-\mathrm{e}^{-(\phi_{Si})^{2/\sigma^2}}, & \phi_{Si}\geqslant 0\end{cases} \tag{4}$$

其中,

$$\sigma=\max\{\phi_{S1},\phi_{S2},\phi_{S3}\cdots\phi_{Sn}\}/3 \tag{5}$$

对每一种属性根据其重要程度,赋予归一化模糊权向量 $\boldsymbol{\omega}$ 为:

$$\boldsymbol{\omega}_S=(\omega_{S1},\omega_{S2},\omega_{S3},\cdots,\omega_{Sn}),\ \sum_{i=1}^{n}\omega_{Si}=1,\omega_{Si}\geqslant 0 \tag{6}$$

通过模糊隶属函数 R 和归一化模糊权向量 $\boldsymbol{\omega}$,可以得到模糊识别的评估结果 E.

$$E_S=\omega_S\circ R(\boldsymbol{\phi}_S) \tag{7}$$

多个特征向量的距离向量采用加权平均法进行单值化处理,本文中 m 表示向量的维度,E_S 是对应的分量,k 主要用于调节较大的 E_S 产生的影响,本文中 $k=2$.

$$E=\frac{\sum_{S=1}^{m}S*E_S^k}{\sum_{S=1}^{m}E_S^k} \tag{8}$$

4.4　基于层次的多属性支持向量机分析

在模糊识别的基础上,可以通过支持向量机[11,12]确认出恶意代码的类型.

由于各类型 API 调用的动态行为彼此独立,为了提高对于恶意代码类型识别的分析精度,必须研究新的支持向量机实现结构,适用于复杂样本的训练,并具有理想的学习速度和精度.本文提出基于层次的多属性支持向量机分析法,其示意图如图 1 所示.

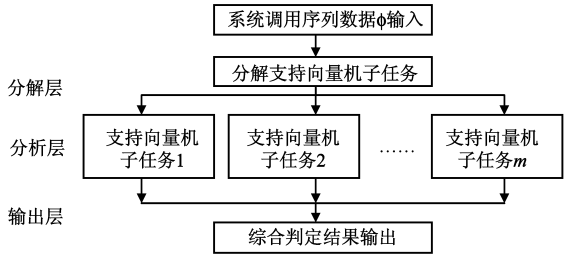

图1　基于层次的多属性支持向量机分析法示意图

分解层基于本文的特征选取结果,有系统调用的动态行为属性样本为 ϕ,将待分析数据按照属性分解为多个支持向量机子任务集合,满足关系如式(9).

$$\phi=\phi_1\cup\phi_2\cup\cdots\cup\phi_m \tag{9}$$

本文中采集的恶意代码特征向量的维度较高,分解层删除关系不大的特征,关注恶意代码的主要属性特征,采用局部线性方法[11].

通过

$$\min_{w_i\in R^n}\left\|x_i-\sum_{j=1}^{m}\omega_{ij}x_j\right\|^2 \tag{10}$$

约束条件 $\sum_{j=1}^{m}\omega_{ij}=1,\omega_{ii}=0$

构造矩阵 $\boldsymbol{W}^*=(\omega_1^*,\cdots,\omega_m^*)$,并通过如下方法进行特征降维,$\overline{\boldsymbol{X}}=(\overline{x_1},\cdots,\overline{x_m})\in R^{d\times m}$.从而缩减维数优化后续的检索,并且可以降低计算量,提高准确率.

$$\min_{\bar{X}}\|(I-\boldsymbol{W}^{*\mathrm{T}})\overline{\boldsymbol{X}}^{\mathrm{T}}\|^2 \tag{11}$$

约束条件 $\sum_{i=1}^{m}\overline{\boldsymbol{x}_i}=0,\frac{1}{m}\sum_{i=1}^{m}\overline{\boldsymbol{x}_i}\,\overline{\boldsymbol{x}_i}^{\mathrm{T}}=\overline{I}$

分析层根据各个子任务,采用支持向量机方法,选取合适的参数 C 和 γ,对各个属性进行分析,得出子结论,提供给输出层分析.对于每个子任务,求解以下二次规划问题[11]:

$$\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j(x_i\cdot x_j)-\sum_{i=1}^{l}\alpha_j \tag{12}$$

$$\sum_{j=1}^{l} \alpha_i y_i = 0 \tag{13}$$

$$0 \leqslant \alpha_i \leqslant C, i = 1,2,\cdots,l \tag{14}$$

输出层基于各个动态行为属性的汉明距离准确定位恶意代码的类型.综合各个支持向量机子任务的输出结果向量,求解向量和模板向量之间的汉明距离,选择最小的汉明距离判断为输出的 Rootkit 类型,这样的选择是最优的.同时,也可以根据汉明距离发现潜在的恶意代码.

关于核函数 $K(x_i, x_j)$的选取,由于本文中的动态行为属性数远小于生成元的个数,所以,在 k-折交叉确认(k-fold cross-validation)中使用 RBF 函数作为核函数.

$$K(x,y) = \exp^{-\gamma \| x-y \|^2} \tag{15}$$

本文使用 k-折交叉确认方法,提高支持向量机子任务的判定准确度.将输入数据随机分成 k 个互不相交的训练集 $S_1, S_2, \cdots, S_K$,然后进行 k 次迭代,每次选择其中一个训练集 S_i 作为测试集与其他的训练集进行训练,根据训练集求出决策函数后,对测试集 S_i 进行测试,得到正确判定的概率 P_i. k 次迭代完成后,得到最优的决策函数.

4.5 根据汉明距离判断类型

基于层次的多属性支持向量机分析,可以得出各个支持向量机子任务的结果输出.通过输出的子结论和测试集进行比较得到的距离,最小的汉明距离,即可以精确判定恶意代码及其类型.

同时,通过确认未知的支持向量机子任务的结果,也可以根据汉明距离发现潜在的恶意代码.表 3 示例的是测试子任务的汉明距离的例子,子任务的输出结果是“10001”,则可以推断出未知的恶意代码的类型为“InstDriver”类型的概率最大.

表 3 测试子任务的汉明距离

Rootkit 的类型	基于层次的多属性支持向量机结果向量	实验数据最小汉明距离
badrkdemo	1 1 0 1 0	3
InstDriver	1 1 0 0 1	1
n00bkit	1 1 1 0 1	2
…	…	

5 实验分析

5.1 实验流程和样本

首先执行恶意代码程序,根据表 2 中的恶意代码异常行为特征典型系统调用序列以及相关参数,提取其实时运行时的 API 序列.提取的信息包括:文件(包括文件名、文件大小、文件类型等);进程(包括进程号、运行时间、CPU 占用、内存占用等);注册表(包括注册表调用号、调用时间、调用参数等);系统服务(包括服务名、启动时间、重要程度、关闭时间等);五元组(包括源 IP 地址、目的 IP 地址、源端口、目的端口、协议类型).Rootkit 动态检测的流程如图 2所示.

实验使用的样本来源于Rootkit网站(http://www.rootkit.com).

仿真实验中,待检测的程序在电脑虚拟机 VMWare(http://www.vmware.com)中运行,提取其在执行时的完整 API 函数调用序列,将待检测程序的系统函数调用序列进行预处理后,联合模糊识别和基于层次的多属性支持向量机分析法,对导入系统的数据进行检测实验.本文的检测流程如下:

(1)采用模糊识别的方法进行实验和数据分析;

(2)基于层次的多属性支持向量机分析法进行实验和数据分析;

(3)钩挂函数添加代码性能和系统性能进行分析.

5.2 实验和数据分析

(1)模糊识别的实验与分析

通过模糊识别的方法,对 Rootkit 恶意程序进行实验,通过调整模糊识别的归一化模糊权向量,可以得到不同的单值化 E 值,其检测结果如图 3 所示.

在单值化 E 分别取不同的值时,系统的报警个数、误报警个数和漏报警个数都不同.在单纯的模糊识别的方法中,报警率平均值只有 67.43%,识别率是偏低的.此外,在 E 值为 4 的情况下,检测系统的报警率最高,系统的漏报个数比较低,与此同时其误报的病毒个数又明显的减少.这就能使检测系统具有相对好的检测效率.能够在实际的病毒检测中最大限度的检测出可疑程序,并且减少系统的误报率,提高了系统的检测效率.

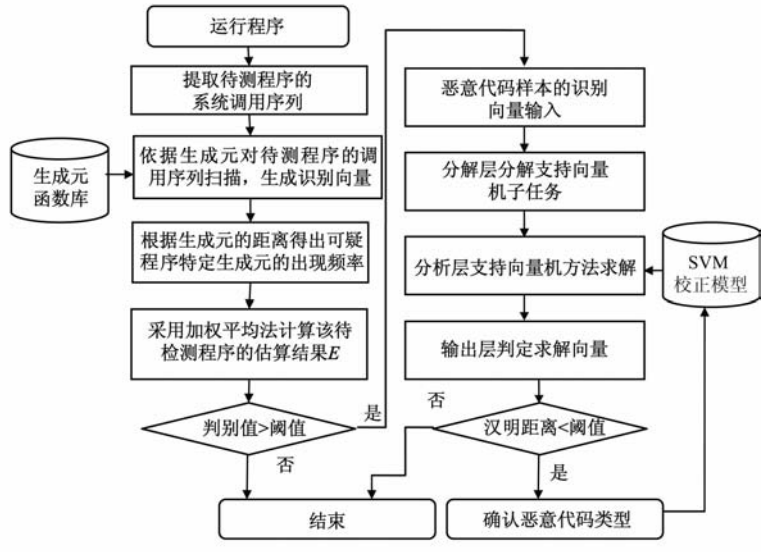

图2 Rootkit动态检测的流程图

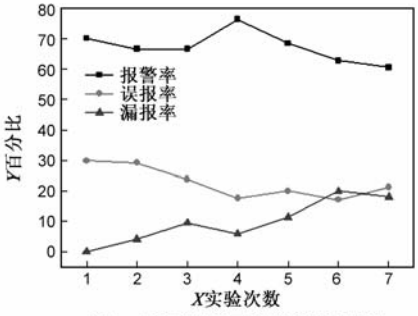

图3　不同阈值下的检测结果图

(2)支持向量机的实验与分析

结合模糊识别的结果,进一步采用支持向量机进行分析.基于层次的多属性支持向量机分析法,按照本文提出的五大属性作为输入的子任务,选取合适的参数 C 和 γ,对各个属性进行分析,得出子结论,最后综合各个子结论的结果向量和测试集的结果测量其汉明距离,最小的即为判定的 Rootkit 类型.表 4 给出的是基于层次支持向量机实验的子任务的参数优化和得到的 SVM 子任务的正确百分比.

表 4　支持向量机各个子任务的测试

属性(子任务)	参数 C	参数 γ	SVM 百分比
注册表	32.0	0.078	95.30
文件	32.0	0.05	96.52
进程	128.0	0.36	98.12
系统服务	64.0	0.11	97.96
网络	128.0	0.5	98.57

单独采用模糊识别的方法进行 Rootkit 恶意代码识别报警率平均值只有 67.43%,而单独采用支持向量机的检测方法进行识别的正确率大概是 75%,而联合模糊识别和基于层次的多属性支持向量机分析法,能够达到 97.30%的平均识别率,并且能够判定未知恶意代码的具体类型.其识别率要大大高于单独的检测方法,从而达到了精确的检测效果.

5.3　性能分析

(1)钩挂函数添加代码性能分析

本文检测方法需要对钩挂函数添加代码,这必然会对原系统调用增加一些开销,这些调用函数的开销反映了检测方法的性能.为了衡量这个开销,本文使用内核函数 KeQueryPerformanceCounter[13,14],该函数提供操作系统中最精确的运行计数,测试钩挂函数添加代码运行时间与原系统调用运行时间之间的对比,对本文提取的部分关键系统调用进行测试,测试结果显示,钩挂函数添加代码开销平均约 27%.测试结果如表 5 所示.

另一方面,对检测系统的整体性能进行对比测试,使用微软公司提供的工具 krview(http://www.microsoft.com/whdc/system/sysperf/krview.mspx),用来收集和分析驱动程序的性能数据,可以测试核心态和用户态时间的使用情况.由图 4 可以看出,本检测系统使用中,对系统的内核态的使用率增加约 2%,用户态的使用率增加约 3.8%,并没有对系统性能产生明显影响,而且性能比较平稳.

表 5　系统调用消耗时间和钩挂函数添加代码开销对比表

API	平均执行时间(ms)		负载百分比(I)
	API 时间(A)	负载时间(O)	$I = O/A * 100$
CreateFile	144.97	30.16	20.80
OpenProcess	48.21	14.29	29.64
RegEnumKeyEx	45.17	15.91	35.22
LoadLibrary	118.11	28.50	24.13
平均值	89.115	22.215	27.45

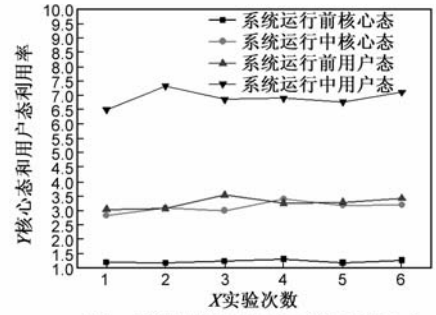

图4　检测技术对操作系统性能影响

6　结束语

本文将 Rootkit 恶意代码运行的系统调用序列数值化后,使用基于模糊识别进行初步识别,对于确认的可疑恶意代码,使用基于层次的多属性支持向量机分析法进行恶意行为检测序列的精确判定.两种方法的结合,一方面可以减少检测的工作量和对系统的影响,另一方面可以提高检测的效果.结果表明了实验的有效性.对多属性的 API 参数的细化分析,以及对两种方法的融合度是本文的进一步研究方向.

参考文献

[1] 中国互联网络信息中心.第 26 次中国互联网络发展状况统计报告[R].2010.24 - 34.
China Internet Network Information Center. China Internet Development Statistics Report[R]. 2010. 24 - 34. (in Chinese)

[2] 刘巍伟,石勇,郭煜,韩臻,沈昌祥.一种基于综合行为特征的恶意代码识别方法[J].电子学报,2009,37(4):696 - 700.
LIU Weiwei, SHI Yong, GUO Yu, HAN Zhen, SHEN Changxiang. A malicious code detection method based on integrated behavior characterization[J]. Acta Electronica Sinica, 2009, 37(4):696 - 700. (in Chinese)

[3] Bilby D. Low Down and Dirty: Anti-Forensic Rootkits[R].

Japan:2006 Blackhat Japan,2006.
[4] 梁升荣.Windows Rootkit 检测机制的研究与实现[D].成都:电子科技大学,2009.13 – 19.
LIANG Shenrong. Research and Implementation of Windows Rootkit Detection Mechanism [D]. Chengdu: University of Electronic Science and Technology of China,2009.13 – 19.(in Chinese)
[5] Vasisht V R. Architectural Support for Autonomic Protection Against Stealth by Rootkit[D]. Georgia Institute of Technology, 2008.8 – 9.
[6] Peinado M, Chen Y, Engl P, Manferdelli J. NGSCB: A trusted open system[A]. Proceedings of 9th Australasian Conference on Information Security and Privacy ACISP, 2004, Sydney, Australia[C]. Berlin, German: Springer, 2004.86 – 97.
[7] Bayer U, Moser A, Kruegel C, Kirda E. Dynamic analysis of malicious code[J]. Journal in Computer Virology, 2006, 2(1): 67 – 77.
[8] Barford P, Yegneswaran V. An inside look at botnets[J]. Advances in Information Security, 2007, 27(1): 171 – 191.
[9] Weber M, Schmid M, Geyer D, Schatz M. A toolkit for detecting and analyzing malicious software[A]. 18th Annual Computer Security Applications Conference[C]. Los Alamitos, CA, USA: IEEE Computer Society, 2007.423 – 431.
[10] 付文,魏博,赵荣彩,庞建民.基于模糊推理的程序恶意性分析模型研究[J].通信学报,2010,31(1):44 – 50.
FU Wen, WEI Bo, ZHAO Rong-cai, PANG Jian-min. Fuzzy reasoning model for analysis of program maliciousness[J]. Journal on Communications, 2010, 31(1): 44 – 50.(in Chinese)
[11] 邓乃扬,田英杰.支持向量机:理论、算法与拓展[M].北京:科学出版社,2009.81 – 111.
Deng Naiyang, TIAN Yingjie. Support Vector Machine: Theory, Algorithm & Continuation[M]. Beijing: Science Press, 2009.81 – 111.(in Chinese)
[12] C W Hsu, C C Chang, C J Lin. A Practical Guide to Support Vector Classification[R]. Department of Computer Science, National Taiwan University, 2003.1 – 16.
[13] Kruegel C. Detecting kernel-level rootkits through binary analysis[A]. Proceedings of the 20th Annual Computer Security Applications Conference[C]. Washington, DC, USA: IEEE Computer Society, 2004.2 – 5.
[14] Battistoni R, Gabrielli E, Mancini L V. A host intrusion prevention system for windows operating systems[A]. 9th European Symposium on Research in Computer Security Sophia Antipolis[C]. France: French Riviera, 2004.352 – 368.

作者简介

李 鹏 男,1979 年 6 月出生于福建长汀,南京邮电大学信息网络专业博士研究生,主要研究方向是计算机通信与网络和信息安全.
E-mail:lipeng@njupt.edu.cn

王汝传 男,1943 年 9 月出生于安徽合肥,南京邮电大学教授、博士生导师.主要研究方向是计算机软件、计算机网络和网格、对等计算、信息安全、无线传感器网络、移动代理等.
E-mail:wangrc@njupt.edu.cn

高德华 男,1985 年出生于江苏泰兴,南京邮电大学计算机应用技术专业硕士研究生.主要研究方向是 IP 网络技术和信息安全.
E-mail:gaodehua1985@163.com