一种融合用户级和内核级拦截的主动防御方案*

许方恒[1]，陈 暄[1]，唐科萍[1]，龙 丹[2]
(1. 浙江工业职业技术学院，浙江 绍兴 312000；2. 浙江大学，杭州 310058)

摘 要：通过对 Windows 环境下程序机理的分析研究，探索采用用户级别拦截、内核级别拦截相结合的主动防御技术实现对恶意入侵行为自动精确检测和自动识别，保障系统和网络的安全。实验结果表明，该技术对于未知恶意入侵及其变种的检测能力均优于单一方法和其他传统检测方法。

关键词：恶意入侵；主动防御；行为分析；API HOOK
中图分类号：TP393.08 **文献标志码**：A **文章编号**：1001-3695(2013)06-1828-03
doi:10.3969/j.issn.1001-3695.2013.06.59

Active defense scheme of fusion of user-level and kernel level interception

XU Fang-heng[1], CHEN Xuan[1], TANG Ke-ping[1], LONG Dan[2]
(1. *Zhejiang Industry Polytechnic College*, *Shaoxing Zhejiang* 312000, *China*; 2. *Zhejiang University*, *Hangzhou* 310058, *China*)

Abstract: Based on the Windows environment procedures mechanism analysis, this paper explored the use of user level intercept, the kernel level intercept which combined with active defense technology of malicious code behavior accurately automatic detection and automatic identification, and proved security system and security of network. The experimental results show that the technology for the unknown malicious code and its variants of the detection ability is better than single method and other traditional methods of detection.

Key words: malicious intrusion; active defense; behavior analysis; API HOOK

随着网络的进一步发展，安全防御和恶意入侵两者在不断的攻防过程中不停地改进和升级，系统的数据信息时刻都受到恶意入侵的威胁。现代的恶意入侵常常采用多态、加壳等技术来隐藏自己，从而绕过反病毒软件的检测。传统的恶意入侵检测技术主要是基于特征码扫描的检测技术[1]，这本质上就决定了只能检测已知恶意入侵，对已知恶意入侵的变种或者是未知的恶意入侵的检测能力极其微弱。为此，本文提出一种基于行为分析的理念，融合用户级和内核级拦截以实现恶意入侵检测的主动防御系统研究方案。该方案设计实现的恶意入侵检测系统针对未知恶意入侵或者恶意入侵的变种都具有一定的检测能力。

1 典型恶意入侵分析方法

传统的恶意入侵分析方法有多种类型，一般可以将恶意入侵分析方法分成基于代码特征的分析方法、基于代码语义的分析方法、基于代码行为的分析方法三种[2]。常见的反病毒软件主要采取的恶意入侵检测方法有：

a)基于特征码检测方法。这主要是通过在网络中部署蜜罐系统获得大量的恶意入侵样本通过反汇编等技术分析采集它们的唯一特征指令序列，将这些特征放入反病毒软件的特征库，当反病毒软件对文件进行扫描时，将当前扫描文件的特征码与病毒特征库进行比较，判断文件代码片段与已知特征码是否一致，从而实现对恶意入侵的判断。这也是被传统杀毒软件厂商广泛采用的反病毒策略。

b)完整性验证方法。主要是检测一些与系统相关的关键程序文件，通过比较它们的 CRC 或 MD5 值是否与正常值有差别，从而作出判断。这种方法的缺点是只有当文件被感染之后才能检测出来，而且相对误报率较高。

c)基于行为的检测方法。主要通过一些技术引导病毒运行，并在运行过程中监控恶意入侵行为。通过对大量恶意入侵行为的观察、研究总结出病毒的共同特征，将检测到的行为特征与已知定义的恶意入侵行为特征进行对比，从而判断出病毒。这种技术的优点是克服了基于特征码检测技术的缺点，可以实现对未知恶意入侵的检测，但是误报率比较高，且实现难度较大。

综合考虑以上典型恶意入侵检测技术优缺点，本文采用基于行为分析技术，融合用户级和内核级拦截以实现恶意入侵检测的主动防御系统研究方案。该方案可以实现对未知恶意入侵及其变种的检测，同时该方案在恶意入侵某些行为方面进行规避处理，这样可以在一定程度上降低对恶意入侵检测的漏报率和误报率。

2 主动防御方案的设计与实现

2.1 整体设计

基于行为分析的恶意入侵检测程序主要是由系统前端界

收稿日期：2012-10-11；**修回日期**：2012-11-29 **基金项目**：国家自然科学基金资助项目(30900358/C100701)；浙江省教育厅科研资助项目(Y2011122724)；浙江省新世纪高等教育教学改革研究项目(yb09138)

作者简介：许方恒(1980-)，男，讲师，硕士研究生，主要研究方向为数据库与信息系统、信息安全(xufangheng321@163.com)；陈暄(1979-) 男，讲师，系统分析师，信息系统项目管理师，硕士研究生，主要研究方向为计算机体系结构、软件方法学、算法设计；唐科萍(1979-)，女，讲师，硕士，主要研究方向为软件工程、计算机应用、网络通信；龙丹(1975-)，男，博士，主要研究方向为图像处理和分析、算法设计.

面、行为获取模块、信息处理模块和行为分析模块四部分组成。整体设计如图 1 所示。行为获取模块主要负责获取恶意入侵的行为信息;信息处理模块主要是将获取的行为信息转换为与行为分析模块中的规则库一致的记录信息;行为分析模块主要负责将经过语义转换的行为信息进行比对处理得到恶意入侵的行为分析报告,反馈给前端界面。

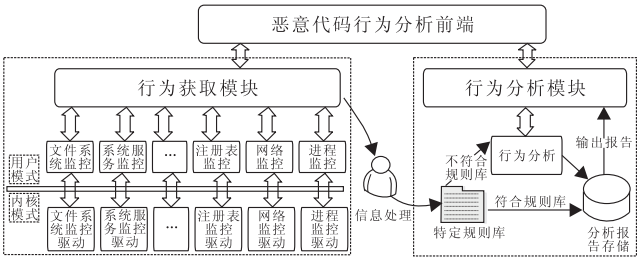

图1　全局架构

2.2　行为检测和捕获

恶意入侵一般的行为特征主要包括文件系统操作行为、系统服务操作行为、注册表操作行为、网络操作行为、进程行为等。基于行为分析的恶意入侵检测程序的行为获取模块主要通过设置不同模块的监控程序密切跟踪恶意入侵的行为,将这些行为结果根据不同的调用模块进行分类,将这些信息提交信息处理模块进行格式化处理,然后将经过处理的恶意入侵行为特征与特定规则库的规则进行逐项比较,当样本行为的特征与特定规则库一致时,就可以判断样本程序为恶意入侵并得到分析报告;否则,将样本行为特征提交分析模块进行分析,从而判断样本程序是恶意入侵的可能性,并将分析报告放入存储模块进行存储,最后输出分析报告反馈给用户。

HOOK 是很多恶意入侵实现其功能的重要手段之一,扫描 HOOK 的结果中就包含了大量关于恶意入侵的行为遗留痕迹,从中分析即可获得恶意入侵的行为特征。基于 Windows 系统的 API 拦截机制主要包括两个方面,即用户级别拦截和内核级别拦截[3]。

截获系统服务调用可以捕获用户模式代码的恶意入侵行为,也能捕获核心态恶意入侵的部分行为。行为捕获模块通过 HOOK 来获取所有的系统调用并收集所有有针对性的信息。

通过驱动之间通信截获内核驱动之间的数据流,主要包括:

a)挂钩系统内核驱动通信有关函数,一般上层驱动向下层驱动发出请求使用一些系统提供的内核函数,因此可以替换函数指针或间接 Inline 挂钩相关函数来截获数据通信。

b)挂钩驱动对象分派例程,NT 架构系统通常在加载一个驱动后为其分配一个驱动对象结构,驱动程序初始化时填充对象结构中的多个分派例程,之后操作系统和其他驱动程序就可以使用这些分派例程与其通信。因此,替代这些分派例程可以截获驱动程序之间的通信,如截获磁盘、文件等访问请求。

c)通过 NDIS 挂钩技术可以截获网络的访问请求数据。

另外,通过关键操作系统的内核扫描能更有效地发现恶意入侵遗留下的行为痕迹,这些关键内核结构包括进程控制块、内核线程控制块、内核模块等。

2.2.1　用户级别 API 截获

用户级别的 API 截获技术主要包括 DLL 注入、拦截内联函数和拦截进程导入地址表[4,5],缺点主要是当恶意入侵直接调用内核函数或者避开调用 Win 32 API 逃避检测[6]。

Microsoft Windows 操作系统上的恶意入侵主要是以二进制可执行文件(PE 文件格式)存在的[7]。Windows 操作系统结构特性决定了恶意入侵的各种行为都是通过调用 API 函数或者系统调用接口来实现的。API 函数是通过系统动态链接库(DLL)进行导出的,这些 DLL 主要包括 Kernel32. dll、ntdll. dll、ws2 32. dll 和 user32. dll。

API HOOK 的基本原理是在函数的入口地址处加一条转移指令(Call 或 JMP),当程序调用原函数时,无条件地实现跳转到用户指定的替代函数去执行,当替代函数完成之后才继续执行原执行体后面的部分,这样就可以截获应用程序对系统 API 的调用行为[8,9]。但是 Windows 系统的 API 是极其庞大的,不可能实现对所有的 API 监控,这样既浪费系统资源也是毫无意义的。对恶意入侵的分析必须有针对性地选取一些特定的 API 进行监控。

为了监控这些特定 API 调用,可以利用 Windows 提供的 DLL 注入技术来实现。Windows 提供了创建远程线程的技术实现对 DLL 的注入。其基本原理是首先将需要监控的 API 调用函数对应的处理函数进行预编译从而形成一个 DLL,然后通过创建远程技术手段将该 DLL 注入到需要监控的应用程序的地址空间中,当应用程序启动时就同时启动监控程序,这样就能实现对恶意入侵执行的 API 调用行为截获。

2.2.2　内核级别 API 截获

用户级别的 API HOOK 技术可以截获对系统 API 的调用行为只能运行在用户态,现在恶意入侵不仅仅是调用用户态的 API 进行破坏,更多的是直接请求系统内核进行服务调用从而造成更大的破坏而且避免了 API 检测模块。因此必须采用 Windows 内核及驱动级相关技术分别实现用户级(Ring 3)的行为截获及内核级(Ring 0)的行为截获,只有获取恶意入侵的所有行为才能真正实现对恶意入侵的全面准确性分析。

内核级别的 API 截获不是通过将代码注入到应用程序,而是注入到内核自身,就可以提供对系统状态的全局监控,恶意入侵很难逃避检测。但是这种内核监控技术受操作系统接口的限制,必须根据不同的操作系统进行更改,移植性较差。

Windows 行为获取体系如图 2 所示。

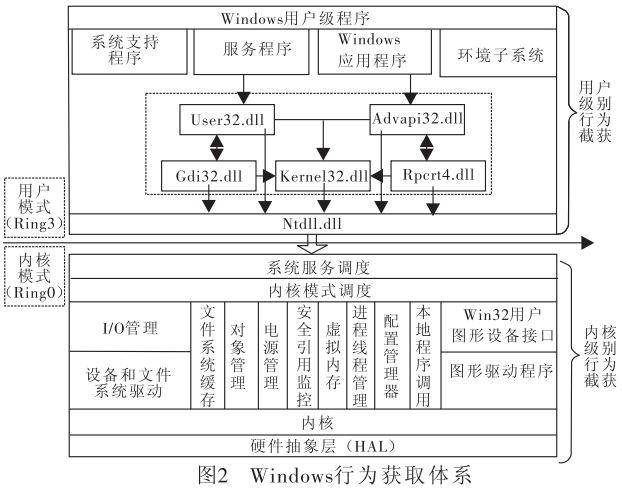

图2　Windows行为获取体系

结合以上两种 Windows 环境下恶意入侵行为获取技术的优缺点,该方案采用将用户级别行为拦截技术和内核级别行为拦截技术相结合的方式进行检测程序的设计与实现,从而真正意义地实现对恶意入侵行为的全面性截获。

2.3　行为处理和融合

为了将获取的行为信息提交给行为分析模块处理就需要提取一些重要的信息并进行格式化处理。信息处理模块主要

负责将获取的行为信息处理成〈API_name + parameter〉的格式,其中 parameter 定义为〈input:输入参数|output:输出参数〉。并将每个截获的 API 处理之后的信息用链表的一个节点存储,形成〈API_1,API_2,API_3,…,API_n〉格式的链表,以便行为分析模块进行遍历分析。其信息链表如图 3 所示。

图3 API信息链表

2.4 方案分析和设计

行为分析是实现基于行为分析的恶意入侵检测的关键环节,该环节的设计好坏直接影响到整个行为分析系统的准确性,因此是一项极具挑战性的工作[10,11]。

基于行为分析的恶意入侵检测技术存在的问题是误报率(即将正常的程序判断为恶意程序)比较高。主要原因是很难对恶意入侵行为进行恰当的定义。基于行为分析的恶意入侵检测需要采用一些手段对某些行为进行回避以降低误报率。行为分析模块包括特定规则库、行为分析、报告存储三部分。

特定规则库主要是定义一些确定的恶意入侵行为特征,当检测到样本程序有特定规则库定义的特征时就直接判断为恶意入侵,这样可以提高检测模块的效率。

为了提高程序检测的准确性,行为分析部分主要从两个方面进行设计:a)恶意入侵的一些特定行为;b)恶意入侵"通常不会做什么",比如针对正常软件的安装及卸载过程中的行为与恶意入侵一些行为非常相似,为了降低误报率就必须考虑"恶意入侵通常不会做,但安装和卸载程序会做"的情况。图 4 为针对正常程序安装卸载或恶意程序运行的处理流程。

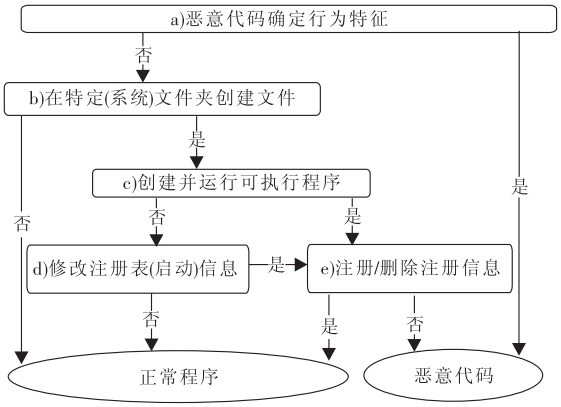

图4 行为分析流程

a)样本行为是否与特定规则匹配,若是则判定为恶意入侵,否则转 b)。

b)样本是否在系统文件夹新建文件,若是则转 c),否则判定为正常程序。

c)样本是否创立并运行可执行程序,若是则转 e),否则转 d)。

d)样本是否修改系统启动注册表项,若是则转 e),否则判定为正常程序。

e)样本在安装和卸载是否进行相关注册表的注册和删除,若是则判定为正常程序,否则判定为恶意入侵。

报告存储模块主要是将行为分析结果写入分析报告,存储于报告存储模块,最后从分析报告存储区取出分析报告反馈给用户,分析报告主要包括样本程序的关键系统行为记录以及系统的判断结果。

3 方案实例化和验证

通过在 VMware Workstation 安装 Windows XP Professional SP3 操作系统的虚拟机中,将样本程序引入恶意入侵检测系统,引导样本运行通过检测系统对样本程序进行检测分析,得到样本程序的分析报告。

样本程序 EuTe Amo. exe 的分析报告如下:

```
[Load DLL]
  C:\WINDOWS\system32\ntdll. dll
  C:\WINDOWS\system32\kernel32. dll
  C:\WINDOWS\system32\advapi32. dll
  C:\WINDOWS\system32\RPCRT4. dll
[Process]
  Creates Process-Filename{|CommandLine:|C:\WINDOWS\system32\ntsds.
exe|As User
  Creates Process-Filename{|CommandLine:|C:\WINDOWS\system32\javas.
exe|As User
[Thread]
  Create Thread-Target PID(1340)Thread ID(1392)Thread ID{ $ 77DC9981)
Parameter Address( $ 00000000)
  [File System]
  Create File:C:\WINDOWS\system32\wab. txt
  Create/Open File:\Device\RasAcd(OPEN_ALWAYS)
Create File:C:\WINDOWS\system32\javas. exe
////////////////////////////////////////////////////
Malware
////////////////////////////////////////////////////
```

其测试结果为恶意入侵,用户可以方便地查看该样本程序对系统进行的操作。

为了验证该方案对恶意入侵的检测能力,测试选取了 10 个木马、10 个蠕虫、10 个病毒、10 个正常程序作为样本程序,同时利用加壳或多态工具 ACProtect、EXECryptor、PE-Armor 进行加壳或多态处理,从而得到样本及其变种共 160 个程序(其中 120 个恶意程序、40 个正常程序)形成的样本集合。然后利用设计的恶意入侵检测程序、基于特征码扫描的瑞星杀毒软件、基于行为分析的微点杀毒软件分别对样本集合进行扫描,检测结果如表 1 所示。

表1 恶意入侵样本检测结果

检测目标	瑞星	微点	本系统
扫描总数	160	160	160
正确检测数	40	115	99
误报检测数	12	20	3
漏报检测数	10	6	20

通过瑞星扫描结果与本系统对照说明:基于行为分析的恶意入侵检测系统对于未知的恶意入侵或者恶意入侵的变种都有很好的检测效果。通过微点扫描结果与本系统的扫描结果数据对比发现,本系统的误报率比较微点的低很多,考虑到微点的行为分析技术的完善性,本系统的正确检测数偏低、漏报检测数偏高也是可以合理的。

4 结束语

本文设计的基融合用户级和内核级的主动防御方案可以对基于 Windows 平台的恶意入侵进行检测,对于未知病毒或同一种病毒及其相应的变种有很好的识别能力,同时在一定程度上降低了误报率。分析检测得到相应的恶意入侵行为分析报告,该报告提供的相关信息可以为恶意入侵分析人员或者用户提供参考,从而在一定程度上提高恶意入侵的分析效率。

(下转第 1839 页)

$(1-q_1^{-1})^{q_5+q_6+q_8}$,在时间 $t' \leqslant t+23t\rho\varepsilon^{-1}+(q_1+q_2+q_3+q_4+q_5+q_6+q_7+q_8+q_9)t_{\max}$ 内解决 CDH 问题。

证明过程类似定理 1,这里不再赘述。

3.2　可识别性

第 $j+1$ 级代理签名验证涉及到其公钥,从公钥中就可识别出第 $j+1$ 级代理人的身份。

3.3　不可否认性

因为多级代理签名具有不可伪造性,如果第 $j+1$ 级代理者产生了有效的多级代理签名,在签名验证过程中必将使用到第 $j+1$ 级代理者的公钥,那么第 $j+1$ 级代理者就无法否认他的代理签名。

3.4　可验证性

在第 $j+1$ 级代理签名验证过程中涉及到第 j 级及以前的各级代理人的公钥,验证者相信第 j 级及以前的各级代理人都同意该消息—签名对。

3.5　防止签名滥用

因为方案中各级代理签名是不可伪造的,委托证书 m_w 中规定了各级代理人的权限是无法被私自修改的,从而有效地阻止签名被滥用。

3.6　效率

衡量方案效率时,仅考虑代价较大的运算,用 e 表示双线性映射操作,H 指 $\{0,1\}^* \to G_1^*$ 上的 hash 运算,S 表示 G_1 上的标量乘运算。

$H_1(\mathrm{ID})$,$H_2(m_w)$,$e(W,P)$,$e(Q_i,P_{\mathrm{pub}})$,$e(P_i,U)$ 在方案中是可预运算的。表 1 数据显示方案在签名阶段无须 e,在验证阶段需 1e。各级所需计算的 e 个数不随代理级数的增大而增加。因此,方案具有很高的效率。

表 1　第 $j+1$ 级运算代价

统计方式	计预运算	不计预运算
部分代理钥生成	$3S+1H$	$3S$
部分代理钥验证	$6e+2(j+2)S+(j+2)H$	$1e$
代理签名	$3S+1H$	$3S$
代理签名验证	$6e+2(j+3)S+(j+3)H$	$1e$

4　结束语

随着电子商务的快速发展,人们之间的代理关系日趋复杂。本文针对不断拉长的代理链,研究了无证书密码系统下的多级代理签名和相应的安全模型。分析显示,该方案满足代理签名的各种安全要求,有很高的效率。同时,该方案还具有前向可追踪性和后向可扩展的灵活性等优点。虽然方案存在签名长度随着代理级数的增加而变大的不足,但是考虑到实际生活中的代理链长度不十分长,所以该方案具有实用价值,可广泛地应用于电子商务、电子政务、电子拍卖等场合。

参考文献:

[1] AL-RIYAMI S, PATERSON K. Certificateless public key cryptography [C]//LNCS, vol 2894. Berlin: Springer-Verlag, 2003: 452-473.

[2] 张福泰,孙银霞,张磊,等. 无证书公钥密码体制研究[J]. 软件学报,2011,22(6):1316-1332.

[3] HUANG Qing, WANG Shi. Generic certificateless encryption secure against malicious-but-passive KGC attacks in the standard model[J]. Journal of Computer Science and Technology, 2010, 25(4): 807-826.

[4] SUN Yin-xia, ZHANG Fu-tai. Secure certificateless encryption with short ciphertext[J]. Chinese Journal of Electronics, 2010, 19(2): 313-318.

[5] ZHANG Lei, ZHANG Fu-tai, WU Wei. A provably secure ring signature scheme in certificateless cryptography [C]//LNCS, vol 4784. Berlin: Springer-Verlag, 2007: 103-121.

[6] 陈虎,朱昌杰,宋如顺. 高效的无证书签名和群签名方案[J]. 计算机研究与发展,2010,47(2):231-237.

[7] 陈虎,张福泰,宋如顺. 可证安全的无证书代理签名方案[J]. 软件学报,2009,20(3):692-701.

[8] 陈虎,朱昌杰,宋永生,等. 有效的无证书代理签名方案[J]. 计算机工程与应用,2012,48(18):89-94.

[9] 冯蕾,彭长根,彭延国. 一种高效的无证书多重签名方案[J]. 计算机应用研究,2012,29(2):415-416.

[10] POINTCHEVAL D, STERN J. Security arguments for digital signatures and blind signatures [J]. Journal of Cryptology, 2000, 13(3): 361-396.

(上接第 1830 页)

下一步的研究主要是全面地获取恶意入侵的行为特征及对行为分析模块算法进行优化,在降低恶意入侵检测误报率和漏报率的同时提高检测正确率。

参考文献:

[1] IDIKA N, MATHUR A P. A survey of Malware detection techniques, Tec Report 268[R]. Muncie, Indiana: Software Engineering Research Center, 2007.

[2] 郑海洋. 普通恶意入侵技术分析与检测[J]. 现代经济信息,2008(10):79-80.

[3] MIAO Qi-guang, WANG Yun, CAO Ying, *et al*. APICapture: a tool for monitoring the behavior of Malware[C]//Proc of the 3rd International Conference on Advanced Computer Theory and Engineering. 2011: 390-394.

[4] 陈培,高维. 恶意代码行为获取的研究与实现[J]. 计算机应用,2009,29(B12):76-78,82.

[5] 袁键,周学思,黄永峰. HOOK 技术在网络隐藏通信中应用[J]. 四川大学学报:自然科学版,2011,48(3):539-545.

[6] BAYER U, HABIBI I, BALZAROTTI D. A view on current Malware behaviors[C]//Proc of the 2nd USENIX Workshop on Large-scale Exploits and Emergent Threats. 2010.

[7] 庞立会. PE 文件动态加壳技术的研究与实现[J]. 计算机工程,2008,34(19):160-162,166.

[8] 管云涛,段海新. 自动的恶意代码动态分析系统的设计与实现[J]. 小型微型计算机系统,2009,30(7):1326-1330.

[9] AHMED F, HAMEED H, SHAFIQ M Z, *et al*. Using spatio temporal information in API calls with machine learning algorithms for Malware detection[C]//Proc of ACM Workshop on Security and Artificial Intelligence. 2009: 55-62.

[10] XU J, SUNG A H, CHAVEZ P, *et al*. Polymorphic malicious executable scanner by API sequence analysis[C]//Proc of the 4th IEEE Symposium of International Conference on Hybrid Intelligent Systems. 2004.

[11] ALAZAB M, VENKATARAMAN S, WATTERS P. Towards understanding Malware behavior by the extraction of API calls[C]//Proc of the 2nd Cybercrime and Trustworthy Computing Workshop. 2010: 52-59.