

# 基于沙盒技术的行为分析系统研究

陈珂<sup>1</sup>, 柯文德<sup>1</sup>, 王爱国<sup>1</sup>, 郑捷<sup>1</sup>, 张良均<sup>2</sup>

(1. 广东石油化工学院 计算机科学与技术系, 广东 茂名 525000;

2. 广州太普信息技术有限公司, 广东 广州 510663)

**摘要:**随着恶意程序的快速增多,常用的分析技术遇到了瓶颈。文中总结与分析了国内外现有主流的恶意程序检测方法,运用了底层的多种HOOK技术,在底层驱动中利用重定向技术从文件、注册表、网络、进程、线程、窗口消息等多个方面,设计与构造了改进的混合型沙盒和行为分析器。沙盒可保证程序在运行中不会破坏真实的系统,可提高分析效率,可连续分析,不需要还原环境。行为分析器通过记录程序的函数调用序列,使用风险等级来判断程序的风险程度,运用了独创的行为分析算法来计算程序的风险级别,通过自定义的规则自动判断程序的恶意程度,同时生成分析报告,达到自动化分析的目的。经过测试,说明该系统达到了预期功能,能有效地保护真实系统,同时也能准确获取到恶意程序的行为,其分析结果是有效的。

**关键词:**虚拟执行;恶意程序检测;沙箱;行为分析

中图分类号:TP309

文献标识码:A

文章编号:1673-629X(2015)08-0166-04

doi:10.3969/j.issn.1673-629X.2015.08.035

## Research on Behavior Analysis System Based on Sandbox Technology

CHEN Ke<sup>1</sup>, KE Wen-de<sup>1</sup>, WANG Ai-guo<sup>1</sup>, ZHENG Jie<sup>1</sup>, ZHANG Liang-jun<sup>2</sup>

(1. Department of Computer Science and Technology, Guangdong University of Petrochemical Technology,

Maoming 525000, China;

2. Guangzhou TipDM Information Technology Co., Ltd., Guangzhou 510663, China)

**Abstract:** With the rapid increase of malicious programs, common analysis technology has encountered bottleneck. In this paper, summarize and analyze the domestic and foreign existing mainstream malware detection method, using the underlying multiple HOOK technology, utilizing the redirection technology in the underlying driver from a file, registry, network, process, thread, window message and so on, design and construct an improved sandbox analyzer and behavior. Sandbox ensures application won't destroy the real system in operation, which can improve the efficiency of analysis, can be used to analyze continuously, do not need to restore the environment. Behavior analyzer by recording the program sequence of function calls, using risk level to judge the risk degree of the program, using the original behavior analysis algorithm to calculate the risk level of the program, through the custom rules automatically judge the malicious degree of the program, at the same time generate analysis report, to achieve the purpose of automatic analysis. After testing, the system runs stable and extensible, the analysis result is valid.

**Key words:** virtual execution; malicious executables detection; sandbox; behavior analysis

## 0 引言

在计算机和网络高度发达的今天,恶意程序的破坏力和影响力表现得尤为突出,它不仅对人们的正常生活和工作带来极大的干扰,还对经济、社会、军事的发展带来严重的破坏,因此,研究恶意程序的分析技术具有重大意义,为计算机系统的改进和发展提供重要依据。目前在该领域,传统的恶意程序分析技术已经

不能满足现今的安全需求,具体表现为以下三点<sup>[1-4]</sup>:

(1)扫描特征码的恶意程序检测技术,不能发现未知的恶意程序,病毒库要不断升级,随着恶意程序的爆炸式增长,特征库随之增大,导致检测率下降。

(2)启发式检测虽能发现未知的恶意程序,但其过高的误报率带来很多烦恼,即使该技术能正确检测恶意程序,但无法确定恶意程序的类型。

收稿日期:2014-09-26

修回日期:2014-12-29

网络出版时间:2015-07-21

基金项目:国家自然科学基金资助项目(61272382);广东省科技计划项目(2012B0101100037);广东省高等学校科技创新项目(2013kjcx0132)

作者简介:陈珂(1964-),男,硕士,副教授,研究方向为数据挖掘、信息安全。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20150721.1448.054.html>

(3)人工动态跟踪检测,虽能解决上述问题,但其效率很低,难以应付恶意程序大幅度增长。

为了提高恶意程序的检测率、正确率,解决传统恶意程序分析的局限性,文中设计并实现了一个利用沙盒技术的恶意程序检测与行为分析系统。

1 沙箱技术

沙箱<sup>[5-9]</sup>(SandBox)也叫沙盒,其原理是把程序生成和修改的资源重定向到沙箱中,程序操作的并不是真实的资源,而是虚拟的资源或者是一个副本。图 1 的网格代表硬盘,白色为空白区域,黑色为存储数据。如图 1(b)所示,在无沙箱下,向硬盘写入数据的位置是不固定的,或是修改系统文件,或是修改其他文件,或是在空白区添加文件。如图 1(c)所示,在有沙箱下,沙箱在硬盘中划分出一个固定区域(矩形框部分),并由沙箱管理。当有写操作时,沙箱将这些写操作重定向到这个固定区域,而不修改已被占用的空间。沙箱退出时,沙箱将此区域清空。依据沙箱不同层次上的实现方法,沙箱可分为应用层沙箱、内核层沙箱、混合型沙箱。混合型沙箱把应用层和内核层结合起来,避免了其他沙盒已存在的弊端,和内核层安全性相比,较大幅度地提升了安全性能。但在其实现过程中,内核层与应用层需要实时交互,因此降低了沙箱的运行效率,也加大了设计与开发难度。

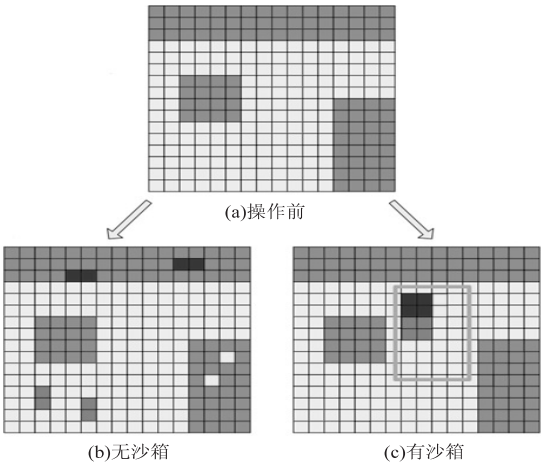


图 1 沙盒技术原理图

鉴于三种沙箱的特点,文中采用改进的混合型沙箱。通过 Hook KiFastCallEntry 函数在内核层监控全部经过 SSDT 的函数,来达到保护系统的目的。同时对应用层的特定函数进行 Hook 来监控更多的程序调用信息,从而得到更详细的分析结果。

2 系统设计方案

本系统主要由沙盒子系统、行为获取子系统、行为分析子系统、系统界面四部分组成。其中沙盒子系统包括 File、Registry、Thread、Netword、Process、Windows Message 六个模块。

系统整体框架如图 2 所示。

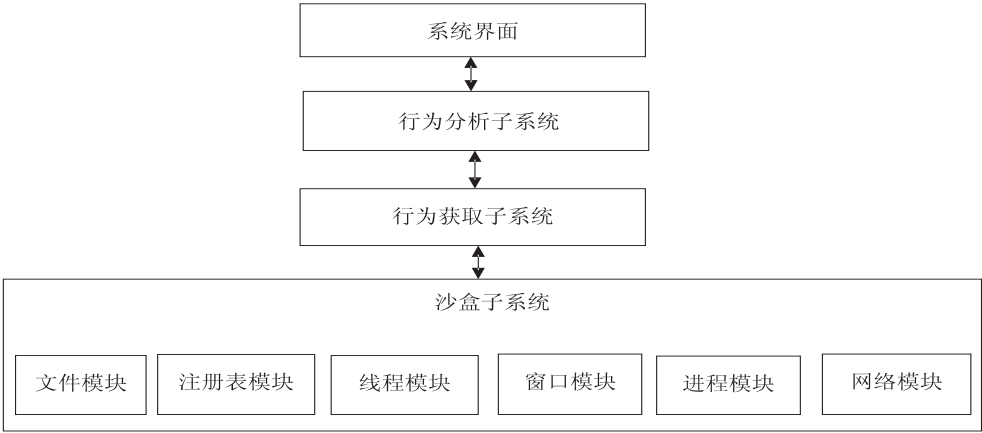


图 2 系统整体框架图

(1)沙盒子系统。

沙盒子系统是整个系统的基础,利用重定向技术,分别对真实系统的文件、注册表、线程、网络、进程和窗口实时保护。沙盒子系统可以有效地阻止恶意程序,防止被其破坏。

(2)行为获取子系统。

通过沙盒提供的回调功能来记录各个敏感函数和参数,同时进行记录和整理,发送到用户层进行查看和分析。

(3)行为分析子系统。

系统通过对程序所调用的序列进行分析,通过自定义的规则来准确识别出风险行为,同时根据风险行为的分数来加权求和,得出程序的风险等级。该等级能客观地反应程序的危险行为和危险程度,有效帮助专业人员进行辅助分析。

(4)系统界面。

主要接收用户请求,实现与用户交互,设置重定向路径、监控内容、防护级别等相关参数,还可以显示被

检测程序信息,敏感函数的操作行为和检测相似度等。

### 3 系统关键技术

SSDT HOOK 由于使用广泛,容易和其他软件引起冲突,而且由于监控函数众多,Inline Hook 实现起来非常繁琐,所以本系统采用一个改进的 Hook 方案。通过 Inline Hook KiFastCallEntry 函数来监控 SSDT 表,在系统通过 SSDT 查询地址时,都是通过 KiFastCallEntry 函数进行的,Hook 该函数可实现监控整个 SSDT 表和 Shadow SSDT 表,非常方便。要实现该方案,有以下几个步骤:

- (1) 找到 KiFastCallEntry 的函数地址,因为该函数未导出,所以不能直接获得;
- (2) 找到合适的位置 Hook;
- (3) 编写代理函数,用来过滤 SSDT;
- (4) 编写代理 SSDT 函数,过滤具体功能。

以下为 Hook KiFastCallEntry 后的函数调用流程,如图 3 所示。

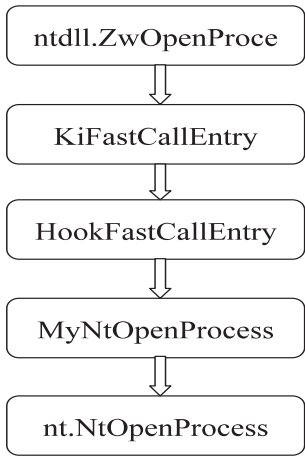


图 3 Hook KiFastCallEntry 流程图

MySSDTTable 是过滤的 SSDT 函数数组,通过一个设置该数组的数据,即可直接 Hook 对应的 SSDT 函数,Hook 成功后,EDX 被替换成代理函数。uBackKiFastCallEntry 是返回地址,执行 jmp uBackKiFastCallEntry 返回到原函数。

### 4 动态获取行为特征序列

函数或者函数序列的调用是程序行为特征的表现,获取函数调用序列(其中已包含每次调用函数时的参数信息)即可确定其行为<sup>[10-14]</sup>。文中研究重点是获取函数调用序列。利用行为序列进行恶意程序检测的关键,是能否准确地获取行为序列。因此,准确地获得函数调用序列是基于行为序列的恶意程序检测技术成败的关键。

本系统通过沙盒的回调来提供函数调用记录的功

能,因为截取函数调用可采用沙盒技术,而沙盒已经先 Hook 了,不能再 Hook,而且沙盒提供全方位的保护,所以编写一个回调系统,在沙盒的 Hook 函数里添加回调功能,通知行为记录模块,从而获取到程序的行为,这样就可以让行为记录模块和沙盒模块兼容。

表 1 是行为记录所需要监控的函数。

表 1 行为监控函数表

API 函数	可能行为
NtCreateFile	创建文件,打开修改系统关键文件
NtSetInformationFile	设置文件信息,删除文件
NtDeleteFile	删除文件
NtWriteFile	写文件,修改系统文件
NtReadFile	读文件,读取系统关键文件
NtCreateKey	创建注册表项
NtOpenKey	打开系统关键项,可能添加危险的键值
NtRenameKey	重命名关键系统项
NtDeleteKey	删除关键系统项
NtDeleteValueKey	删除关键系统键值
NtCreateUserProcess	创建新进程
NtDebugActiveProcess	启动调试功能
NtQueryInformationProcess	查询系统关键进程
NtSuspendProcess	暂停进程
NtResumeProcess	恢复进程
NtSetInformationProcess	设置进程信息
NtOpenProcess	打开进程
NtTerminateProcess	关闭进程
NtCreateThreadEx	创建远程线程,远程注入
NtOpenThread	打开线程
NtSuspendThread	暂停线程
NtResumeThread	恢复线程
NtTerminateThread	结束线程
NtSetContextThread	设置线程上下文可能破坏或者修改其他进程环境
NtGetContextThread	获取线程上下文
NtSetInformationThread	设置线程信息
NtUserSendInput	发送模拟鼠标键盘输入
NtUserSendMessage	以 Send 方式向其他窗口传递消息
NtUserPostMessage	以 Post 方式向其他窗口传递消息
NtUserFindWindowEx	查找制定窗口
NtUserSetWindowsHookEx	安装消息钩子

通过监控以上函数,就可以得到足够的程序行为,然后进行分析。以下为沙盒 NtCreateFile 调用回调函数的例子。

ULONG \_\_stdcall SafeBOX\_NtCreateFile ( OUT PHANDLE  
FileHandle,

```
IN ACCESS _ MASK DesiredAccess, IN POBJECT _ AT-
TRIBUTES ObjectAttributes,
OUT PIO _ STATUS _ BLOCK IoStatusBlock, IN PLARGE _ IN-
TEGER AllocationSize OPTIONAL,
IN ULONG FileAttributes, IN ULONG ShareAccess, IN UL-
ONG CreateDisposition,
IN ULONG CreateOptions, IN PVOID EaBuffer OPTIONAL,
IN ULONG EaLength)
{
//接下来调用回调函数 Mon_PostNtCreateFile,通知记录 Nt-
CreateFile 事件
Mon_PostNtCreateFile ( FileHandle, DesiredAccess, ObjectAt-
tributes, IoStatusBlock, AllocationSize,
FileAttributes, ShareAccess, CreateDisposition, CreateOp-
tions, EaBuffer, EaLength, LastStatus );
}
```

## 5 系统评测

针对沙箱子系统、行为获取子系统,对基于沙盒技术的程序行为分析系统进行评测。通过把函数的调用记录传输到界面上,然后显示出来,提供分析数据。这些函数是沙盒中的文件读写记录以及进程操作记录,通过对这些函数序列进行分析,就可以判断程序的恶意行为。

从检测到下载行为的测试结果,可以发现,下载者的三个行为都被发现并显示出来,先后调用了 UrlDownloadToFile, NtCreateFile, CreateProcess, 符合下载者的判断规则。

对行为检测系统的各个功能模块进行了针对性的实验,通过实验证实系统实现了预期功能,能有效保护真实系统,同时也能准确获取到恶意程序的行为。

## 6 结束语

文中设计并实现了一个基于沙盒技术的行为分析系统,系统包括沙箱子系统、行为获取子系统、行为分析子系统、系统界面四部分,前三部分是文中研究、设计和实现的重点。其创新点是设计轻量级沙箱,实现了高扩展性的行为判断规则。

Windows 是非常复杂的系统,沙盒保护若要全面,是非常困难的。目前沙盒很容易被一些特殊的恶意程

序穿透,对系统造成破坏,需进一步完善沙盒,添加更多规则,提高系统稳定性。

### 参考文献:

- [1] Schultz M G, Eskin E, Zadok E. Data mining methods for detection of new malicious executables [C]//Proceedings of 2001 IEEE symposium on security and privacy. [s. l.]: IEEE, 2001: 38-49.
- [2] Christodorescu M. Static analysis of executables to detect malicious patterns [C]//Proceedings of the 12th conference on USENIX security symposium. [s. l.]: [s. n.], 2003: 169-186.
- [3] Tony Abou-Assaleh, Cercone N, Sweidan R. Detection of new malicious code using n-grams signatures [C]//Proc of second annual conference on privacy, security and trust. [s. l.]: [s. n.], 2004: 193-196.
- [4] 卢浩, 胡华平, 刘波. 恶意软件分类方法研究 [J]. 计算机应用研究, 2006, 23(9): 4-7.
- [5] Moser A, Kruegel C, Kirda E. Limits of static analysis for malware detection [C]//Proceedings of twenty-third annual computer security applications conference. Miami Beach, FL: IEEE, 2007: 421-430.
- [6] 卓新建. 计算机病毒原理及防治 [M]. 北京: 北京邮电大学出版社, 2004.
- [7] 邢文利. 恶意代码动态分析系统的设计与实现 [D]. 北京: 清华大学, 2005.
- [8] 鲜明, 包卫东. 网络攻击效果评估 [M]. 长沙: 国防科技大学出版社, 2007.
- [9] 江雪, 朱永强. 基于分层权值的恶意程序仿真系统设计与实现 [J]. 计算机技术与发展, 2014, 24(4): 143-146.
- [10] 张波云, 殷建平, 篙敬波, 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.
- [11] 辛毅, 方滨兴, 贺龙涛, 等. 基于通信特征分析的蠕虫检测和特征提取方法的研究 [J]. 通信学报, 2007, 28(12): 1-7.
- [12] 唐彰国, 李焕洲, 钟明全, 等. 基于网络通信指纹的启发式木马识别系统 [J]. 计算机工程, 2011, 37(17): 119-121.
- [13] 樊震, 杨秋翔. 基于 PE 文件结构异常的未知病毒检测 [J]. 计算机技术与发展, 2009, 19(10): 160-163.
- [14] 单长虹, 张焕国, 孟庆树, 等. 一种启发式木马查杀模型的设计与分析 [J]. 计算机工程与应用, 2004, 40(20): 130-132.

## 2015 中国计算机大会 (CNCC2015)

2015 中国计算机大会 (CNCC2015) 将于 2015 年 10 月 22 ~ 24 日在安徽合肥召开, 诚挚邀请您参加 (报名网址: <http://cncc.ccf.org.cn/>)。预祝大会胜利圆满召开!

作者：[陈珂](#)，[柯文德](#)，[王爱国](#)，[郑捷](#)，[张良均](#)，[CHEN Ke](#)，[KE Wen-de](#)，[WANG Ai-guo](#)，[ZHENG Jie](#)，[ZHANG Liang-jun](#)

作者单位：[陈珂, 柯文德, 王爱国, 郑捷, CHEN Ke, KE Wen-de, WANG Ai-guo, ZHENG Jie \(广东石油化工学院 计算机科学与技术系, 广东 茂名, 525000\)](#)，[张良均, ZHANG Liang-jun \(广州太普信息技术有限公司, 广东 广州, 510663\)](#)

刊名：[计算机技术与发展](#)

英文刊名：[Computer Technology and Development](#)

年，卷(期)：2015(8)

引用本文格式：[陈珂](#).[柯文德](#).[王爱国](#).[郑捷](#).[张良均](#).[CHEN Ke](#).[KE Wen-de](#).[WANG Ai-guo](#).[ZHENG Jie](#).[ZHANG Liang-jun](#) [基于沙盒技术的行为分析系统研究](#)[期刊论文]-[计算机技术与发展](#) 2015(8)