基于行为的主机入侵防御系统设计

成 洁
(四川水利职业技术学院，四川 都江堰 611830)

摘 要：针对个人计算机面临的安全威胁提出一种基于行为的主机入侵防御系统方案(B-HIPS)。B-HIPS 系统针对应用程序、注册表、文件、自身保护等方面进行全方位的设计与功能模块划分，并针对进程创建、注册表写入、驱动加载等部分功能实现 B-HIPS Demo。对于 Demo 程序采用“灰鸽子”与“熊猫烧香”两种样本进行测试，成功阻止了样本对主机的破坏行为，完成防护的目的。

关键字：信息安全；主机入侵防御；行为
中图分类号：TP309.5 文献标识码：A

Design of behavior-based host intrusion prevention system

CHENG Jie
(Sichuan Water Conservancy Vocational College, Dujiangyan 611830, China)

Abstract: In the face of the security threats, this paper presents a solution named behavior-based intrusion prevention system(B-HIPS)with protects computer all-directional, including application defend, registry defend, file defend, self defend and so on. B-HIPS Demo is completed as a demo program with some function of system, which monitors all behaviors of creating process, writes registry and loads driver. The demo program is successfully in passing the test of two samples named “Worm. WhBoy” and “win32.hack.huigezi”. It prevents virus sample from destroying personal computer.

Key words: information security; host intrusion prevention; behavior

随着网络技术的发展及其应用范围的扩大，Internet 的迅速发展大大地改变了以单机为主的计算模式，便捷的网络提供了资源共享性的同时也加剧了网络安全的脆弱性和网络遭受攻击的可能性和风险性。通过互联网下载、浏览网站和电子邮件感染计算机病毒的用户数量明显增加，网络成为计算机病毒传播的重要途径。在人们不断采用新安全技术抵御攻击的同时，攻击者也在不断演练更复杂的攻击技术，攻与防之间处于此消彼长的态势。

CSI/FBI 的安全报告[1]显示，90% 的入侵行为可以绕过防火墙。在网络安全事件中，86% 的用户使用了防火墙，42% 的用户使用了入侵检测系统(IDS)。当前这些传统的安全防护手段已经显示不足，人们迫切需要找到一种更加主动的入侵防护解决方案，它不仅能监测攻击并报警，更重要的是能够采取主动防护手段，实时地阻止攻击，入侵防御系统 IPS(Intrusion Prevention System)就采用了这种安全理念[2]。本文针对个人计算机的安全防护，提出一种基于行为的主机入侵防御系统 B-HIPS (Behavior-based Host IPS)方案，并对其进行了全方位设计和部分实现。

1 基于行为的主机入侵防御系统的设计

B-HIPS 安全解决方案应付攻击的办法就是在操作系统的层面上对操作系统和应用程序进行加固。操作系统扮演着“漏斗”的角色，所有的服务请求都必须通过它执行。操作系统设计的一个最基本的原则就是对所有的系统资源进行隐藏。访问这些系统资源的唯一

方式是利用操作系统提供的预定义的系统服务例程，这些服务例程又被称之为系统调用。任何企图直接访问系统资源的行为都会被操作系统阻止。这些系统调用接口定义非常清晰，它只允许激活操作系统内核中相应的服务例程。

B-HIPS工作在个人计算机主机上，采用策略和基于可接受行为的访问控制规则进行入侵检测和阻止。参考多篇文献的思想[3-5]，本文设计了如图1所示的基于行为监控的主机安全防护系统B-HIPS。该系统主要分为以下几个模块：访问控制规则和策略、应用程序运行控制模块、文件保护模块、注册表保护模块以及自我保护模块。

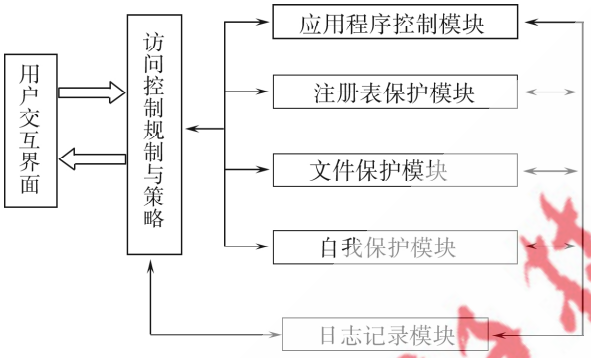

图1 B-HIPS模块组成

(1)访问控制规则和策略：这个模块主要是一个基于访问控制策略的文件。在程序启动的时候，加载规则到内存中，对应用程序、文件、注册表等操作的系统调用都将经过规则的检查，对禁止的予以终止，对允许的予以放行，对未定义的予以警告并等待用户裁决；

(2)文件保护模块：对主机所有文件的访问进行防护，根据事先定义的访问控制策略，可以对特定文件的操作予以禁止或允许。通过截获与文件操作密切相关的几个系统服务，从而实现对文件操作的访问控制，包括文件的复制、删除、剪切和移动保护、文件的重命名、文件的创建、打开以及文件的读写；

(3)注册表保护模块：通过截获与注册表访问相关的系统服务，利用自定义的访问控制规则对注册表的各种操作进行监视和保护。实现对注册表关键键值和键的创建、读写、隐藏、重命名和删除等操作进行保护；

(4)自我保护模块：主要防止HIPS系统的关键进程和系统中的重要进程被恶意进程攻击和终止，以及实现控制内核模块的加载和卸载，还能够有效阻止Rootkits[7]的加载和运行，防止了利用Rootkits进行攻击的企图；

(5)应用程序运行控制模块：这个模块主要是对不受系统信任的应用程序的运行进行控制，能够在一定程度上有效防范恶意进程运行；

(6)日志记录模块：记录所有日志信息。

2 B-HIPS程序模块的设计与实现

下面将介绍B-HIPS访问控制的公共模块及各子模块功能设计、实现原理和部分功能的具体实现。

2.1 公共模块

B-HIPS的主体流程如图2所示，将主体的初始化分为界面初始化、驱动程序初始化、控制规则和策略初始化3方面。

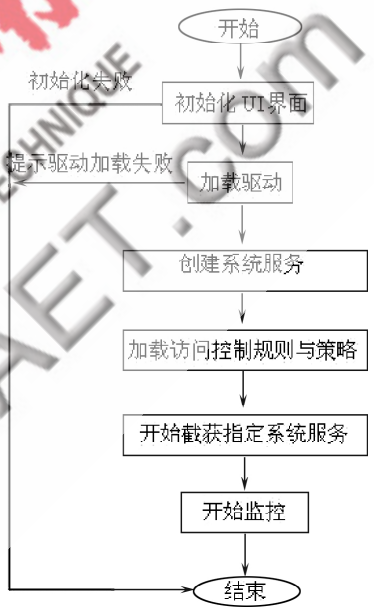

图2 B-HIPS主体流程

(1)界面初始化

优秀的UI界面有助于用户对B-HIPS进行控制使用，本系统采用Windows SDK进行界面化设计，在提高主机运行效率的同时也在B-HIPS启动的第一时间予以用户最友好的控制支持。UI主要包括规则和策略的查询与修改、日志的定位查询、B-HIPS各模块的启动与关闭等。

(2)驱动程序初始化

由于B-HIPS入侵防御系统深入Windows操作系统核心层，所以必须使用内核设备驱动程序来实现。但HIPS系统控制的驱动程序并不驱动任何实际的物理设备，实际上它是一个伪设备驱动程序。像其他类型的程序一样，驱动程序也有一个入口函数DriverEntry，通过它进入驱动程序，并完成相应的初始化工作，创建输

入/输出设备对象以与用户态的 GUI 进行通信。

使用 IoCreateDevice 创建一个虚拟设备，设置内核驱动函数数组指针，完成驱动与用户界面程序的通信初始化。

```
theDriverObject->MajorFunction[IRP_MJ_CLOSE] = DevCreateClose;
theDriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL]= DevDispatch;
theDriverObject->DriverUnload = OnUnload;
```

DevCreateClose 为驱动退出时的操作函数，DevDispatch 为驱动的控制函数，OnUnload 为驱动卸载时的卸载函数。用户在用户层程序中可以通过 IRP 与驱动程序进行交互 DevDispatch 中 case 1000 完成了对进程创建监控函数的调用，用户层程序(UI 界面等)通过 DeviceIoControl(device，1000，controlbuff，256，controlbuff，256，&dw，0)完成进程创建监控函数的加载，实现对进程创建的监控。

(3)控制规则与策略初始化

控制规则与策略作为一个独立模块能够为用户使用 B-HIPS 提供灵活性，用户可以定义符合自身的规则来减少 B-HIPS 警告提示。通过选择比较，XML 格式文件可以作为该模块的实现方式。

XML 是一种简单的数据存储语言，使用一系列简单的标记描述数据，而这些标记可以用方便的方式建立，简单易用是 XML 最主要的优点，也使其易于在任何应用程序中读写数据，这使 XML 成为数据交换的唯一公共语言。根据参考文献[3]的思想，设计如图 3 所示的规则流程，B-HIPS 在截获到系统调用时将遍历该 XML 规则，寻找符合该调用的规则，并按该规则执行操作，对于未找到规则的调用则直接提示用户警告，由用户选择执行方式。

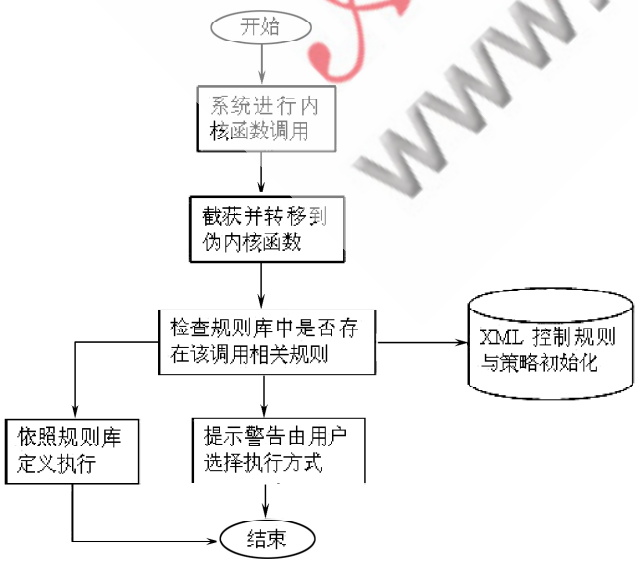

图 3 截获系统调用规则流程

2.2 应用程序控制模块

应用程序控制模块主要完成对应用程序进程的创建、打开、终止等行为的监控，具体需要进行监控的内核函数如表 1 所示。

表1 应用程序控制模块截获的内核函数表

Windows 原内核函数	相应截获函数	实现的功能
ZwCreateProcess	FakedZwCreateProcess	监控进程创建行为
ZwCreateSection	FakedZwCreateProcess	监控进程创建行为
ZwOpenProcess	FakedZwOpenProcess	监控打开新进程行为
ZwTerminateProcess	FakedZwTerminateProcess	监控终止进程行为

2.3 文件保护模块的设计

文件保护模块主要完成对文件、目录的创建、打开、读写、删除以及属性修改等行为的监控，设计重点在于对表 2 具体内核函数的调用截获。与前面应用程序控制模块类似，同样需要使用钩子技术“钩住”相关函数，用自定义的函数地址替换原有内核函数地址。而且同样需要获取与文件操作的相关进程路径、名称以及文件的名称，然后向用户发出提示警告。

表2 文件保护模块截获的内核函数表

Windows 原内核函数	相应截获函数	实现的功能
ZwCreateFile	FakedCreateFile	监控文件、目录创建、打开、复制、剪切等行为
ZwReadFile	FakedZwReadFile	监控读文件行为
ZwWriteFile	FakedZwWriteFile	监控写文件行为
ZwSetInformationFile	FakedZwSetInformationFile	监控文件属性控制操作，包括重命名、删除、以及其他相关属性修改行为
ZwQueryDirectoryFile	FakedZwQueryDirectoryFile	监控文件、目录隐藏行为

2.4 注册表保护模块

注册表保护模块主要完成对注册表键的创建、打开、删除以及键值的写入、删除等行为的监控，设计重点在于对表 3 具体内核函数的调用截获。与前面应用程序控制模块类似，同样需要使用钩子技术“钩住”相关函数，用自定义的函数地址替换原有内核函数地址。而且同样需要获取进程的路径、名称以及注册表项的键和相关键值。

2.5 自我保护模块

B-HIPS 作为保护个人计算机安全的系统，很可能

表3 注册表保护模块截获的内核函数表

Windows 原内核函数	相应截获函数	实现的功能
ZwCreateKey	FakedZwCreateKey	监控键创建行为
ZwOpenKey	FakedZwOpenKey	监控键打开行为
ZwSetValueKey	FakedZwSetValueKey	监控键值写入行为
ZwDeleteKey	FakedZwDeleteKey	监控键删除行为
ZwDeleteValueKey	FakedZwDeleteValueKey	监控键值删除行为

直接受到病毒木马等的直接攻击。如果 B-HIPS 瘫痪，将导致计算机系统瞬间失去保护罩，任由木马病毒肆虐。自我保护模块正是很大程度上杜绝这种情况发生的模块。终止进程内核函数 ZwTerminateProcess 已经在应用程序控制模块中予以实现，但在内核层上作为驱动的病毒木马同样能够获得与 B-HIPS 相同的 Ring0 权限，这样将有可能通过各种未知的 Windows 内核函数对系统甚至 B-HIPS 进行攻击，因此，自我保护模块应该对这一缺陷予以弥补，截获的相关函数如表 4 所示。

表4 自我保护模块截获的内核函数表

Windows 原内核函数	相应截获函数	实现的功能
ZwLoadDriver	FakedZwLoadDriver	监控驱动加载行为
ZwUnloadDriver	FakedZwUnloadDriver	监控驱动卸载行为

2.6 日志记录模块

该模块为用户提供程序运行的相关日志，包括启动、关闭、拦截信息、控制规则与策略查询等相关信息。通过 DailyRecord 函数完成对 dailyrecord.txt 文件的写入，其他模块通过 DailyRecord 函数进行写入日志，属于用户层操作，实现比较简单。

2.7 B-HIPS 的部分实现与测试

本段仅仅演示部分函数的截获，程序定名为 B-HIPS Demo，能够较好地保护系统的安全。如图 4 所示，主要截获了(1)ZwCreateProcess，针对进程创建，通过 ZwCreateProcess 函数替换；(2)ZwSetValueKey，针对注册表键值写入，通过 ZwSetValueKey 函数替换；(3)ZwLoadDriver，针对驱动加载，通过 ZwLoadDriver 函数替换。

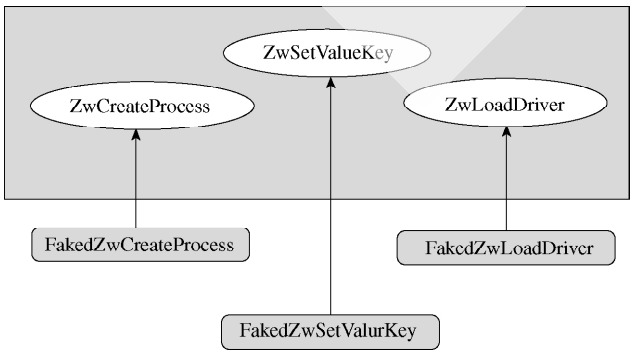

图 4 实现截获函数

对于 B-HIPS Demo 程序，通过熊猫烧香、灰鸽子对其进行测试，测试环境为虚拟机环境 VmWare 6.0，安装 Microsoft Windows XP professional with Service Pack 2(原版，未安装任何补丁)，运行 B-HIPS Demo，打开全部 3 项监控选项。对于每个样本的测试完毕后使用 VmWare 快照功能恢复系统，确保测试结果的准确性。

不论攻击者采用何种攻击技术，最终都是通过操作系统提供必要的支持进行。也就是说，发生在个人计算机主机上的一切活动，最终都需要使用操作系统来完成。这些活动在操作系统内核表现为内核函数的调用，即本文所说的行为。基于行为的主机入侵防御系统有助于强化对主机应用程序、文件、注册表等各方面资源的保护。作为当今个人计算机的最后一道防线，B-HIPS 有能力抵御病毒木马的攻击，由于其针对的是行为，因此摆脱了病毒库的 B-HIPS 能够对任意新型病毒木马或变种进行拦截，从而保护个人计算机的安全。实验表明，本系统能够对“熊猫烧香”样本的破坏行为予以完全阻止、对“灰鸽子”的大部分破坏行为予以阻止，总体上达到了该程序的设计目的，同时也证明了完整的 B-HIPS 思想的可行性与对系统保护的安全性。

鉴于 B-HIPS 自身基于行为的特点，因此对任何行为不分善恶都一律拦截，这势必造成弹出窗口过多的问题，除了通过有效的规则库予以解决外，行为自动化智能分析将成为研究的可行方向。同时，还需强化实现规则与策略控制模块，避免重复操作，并对规则进行层次化分类，通过有效算法提高搜索效率。值得说明的是，道高一尺，魔高一丈，技术往往是一把双刃剑，保持对病毒木马的与时俱进将有助于 B-HIPS 的进一步加强。

参考文献

[1] 中国互联网络信息中心. 中国互联网络发展统计报告[EB/OL]. http://www.cnnic.net.cn/uploadfiles/doc/2008/1/17/104126.doc.2008.

[2] 贾永刚. 基于SNORT的分布式入侵检测体系结构的研究[D]. 杭州：浙江大学，2007.

[3] Xianghe L，Liancheng Z，Shuo L. Kernel rootkits implement and detection[J]. 武汉大学学报：(自然科学英文版)，2006，11(6)：1473-1476.

[4] BARRY B. The intel microprocessors(7th ed)[M]. Prentice Hall，2006.

[5] 高岩，蒋若江. 主机防护系统中系统调用截获机制的实现[J]. 计算机工程与设计，2003，24(11)：76-80.

(收稿日期：2009-02-20)