感染型病毒防御系统①

郑焕鑫, 叶小平

(华南师范大学 计算机学院, 广州 510631)

摘 要: 通过设计一个简单感染型病毒, 发现主流杀毒软件主动防御策略无法对其进行有效拦截. 根据在正常情况下系统 PE 文件不会发生变化的特性, 提出通过拦截对已有 PE 文件写操作来实现主动防御的方法, 并设计相应系统 PEPS. 仿真实验表明, 该方法对于感染型病毒的防御效果优于主流杀毒软件.

关键词: PE 感染型病毒; 写操作; 拦截; PEPS

Defense of Infectious Viruses

ZHENG Huan-Xin, YE Xiao-Ping

(School of Computer, South China Normal University, Guangzhou 510631, China)

Abstract: After designing a simple infectious virus, we find the Active Defense Strategy of mainstream anti-virus software can't intercept the infectious operations effectively. Under normal circumstances, the original PE files of the system cannot be modified. According to this characteristic, the following article develops a way to realize initiative recovery by monitoring illegal write operation of original PE file and design a system – PEPS. The simulation experiments show that the method is more effective on the defense of infectious viruses than mainstream anti-virus software.

Key words: PE infectious viruses; write operation; interception; PEPS

根据瑞星公司发布 2011 年上半年病毒趋势分析报告, 当前病毒主要是通过使用少量的由汇编语言编写引导部分加载由高级语言编写的主体功能部分. 这与早期采用纯汇编语言方式相比, 降低了门槛, 使得病毒变种速度不断加快. 与一般恶意代码相比, 感染型病毒不能以特征码方法直接查杀. 系统关键文件被感染后, 需要进行相应修复工作. 感染型病毒寄生对象是 windows 可执行文件, 对宿主的寄生方式变化多端, 目前还不存在通用的修复工具. 一旦病毒感染成功, 可能会对用户造成不可还原的破坏, 此时, 针对每个不同的感染型病毒编写专门修复工具将消耗大量人力物力.

在感染型病毒变种快速化和寄生方式多样化的情况下, 采取主动防御阻止这类病毒感染成功是当务之急. 本文在分析感染型病毒寄生方法基础上, 对比主流杀毒软件相应防御方法的缺陷进行相应的研究与改进.

1 主流杀毒软件防御缺陷分析

1.1 感染型病毒基本模型

病毒首先判断目标文件是否为 PE 文件[1,7], 通常判断标准是"MZ"和"PE"组合标志[2]; 接着在区块表最后添加一个写有病毒代码的区块[3], 相应过程如图 1 所示. 为防止重复感染, 在感染处设置感染标志. PE 文件感染病毒后, 宿主程序执行时先执行新建区块中代码, 然后跳转到原来宿主代码区块执行. 通过这个过程, 病毒获得执行权限, 同时也执行了宿主代码, 使得用户认为该执行程序没有问题.

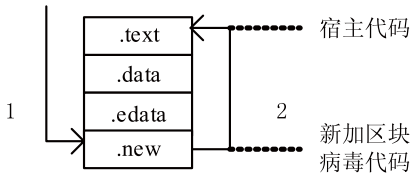

图 1 感染方式数据结构

① 基金项目:广东省自然科学基金(9151027501000054,s2011010003409);2011 年华南师范大学大学生创新实验计划
收稿时间:2012-05-20;收到修改稿时间:2012-08-18

1.2 简易感染性病毒

根据上述分析，可以设计一个简单感染型病毒[3,6]来测试主流杀毒软件的防御能力. 该感染病毒的特征表现为宿主执行时会先释放“中毒了”对话框. 病毒机理为: ① 判断是否为 PE 文件; ② 获取宿主文件区块表信息; ③ 新建一个块，放在最后一个区块之后，区块的内容为获取 Kernel32.dll 模块基址地，进而获取 GetProcAddress 地址，通过 GetProcAddress 地址获取 LoadLibraryA 地址，使用 LoadLibraryA 装入 Test.DLL(这个 DLL 里面可以实现病毒的所有功能，此时弹出“中毒了”的对话框); ④ 新区块最后必须将入口跳转到原 OEP 处; ⑤ 修改最初 OEP，使其先执行新建区块 NYsky; NOTEPAD.EXE 被感染之前的区块信息如图 2 所示(采用 PEid v0.92 软件打开)，被感染之后多了一个命名为.zhxfl 的区块，如图 3 所示.

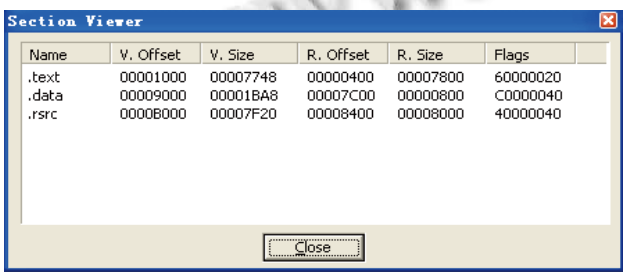

图 2　未感染病毒的 NOTEPAD.EXE 区块信息

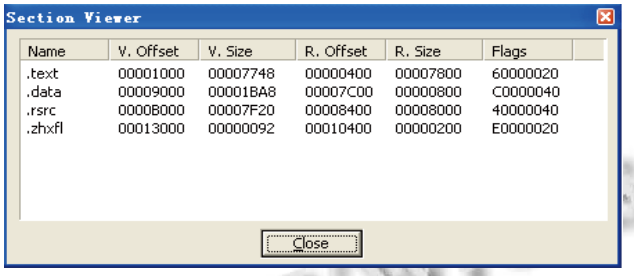

图 3　感染病毒的 NOTEPAD.EXE 区块信息

1.3 主流杀毒软件防御缺陷

根据 1.2 设计的感染型病毒，开启各主流杀毒软件的主动防御，然后运行病毒，感染结果如表 1 所示(YES 为感染成功, NO 为失败).

表 1　主流杀毒软件主动防御效果

金山	360	ESET	QQ 管家	小红伞
Yes	Yes	Yes	Yes	Yes

感染型病毒设计理念是通过寄生于宿主文件，运行宿主时，病毒首先运行，因此寄生宿主一般是系统经常运行的可执行文件. 主流杀毒软件认为，最容易被感染的文件是系统目录下一开机就会自动加载的 PE 文件，所以保护了系统关键目录，但系统中常用文件不止是 C:\WINDOWS 目录下文件，例如还有 office, SQL Server 等，这些常用 PE 文件可能成为病毒成功感染的一个突破口. 主流杀毒软件缺陷在于其所保护的范围不够严格.

2　PEPS设计机理

根据上述分析，设计了针对感染型病毒的主动防御系统 PEPS(PE 保护系统，PE protect system，PEPS)，主要思路是禁止对系统原有 PE 文件进行改写. 为在严格防御病毒感染的同时满足不影响系统进程以及一般应用程序, PEPS 设计基于以下两点: A_假设,常态下 PE 文件的容不会被改写; B_假设，一般应用软件安装过程虽释放 PE 文件，但不会改写系统本来的 PE 文件. 上述两点将在“3”中通过实验表明其合理性.

对于一个首次在系统中执行(包括安装)的 PE 文件 X，必须做出如下防卫措施: ① 检测系统原有 PE 文件内容是否在 X 执行时发生改变，如有则阻止其发生; ② 记录 X 执行过程中释放 PE 文件(有可能是病毒体)，方便发现有感染行为后删除.

2.2 系统设计

实现上述思路需对 windows 的写文件和创建文件操作进行监控和拦截，这就要对 windows 的 I/O 操作进行分析，选择最佳拦截时机. 以写操作为例，拦截过程如图 4 所示.

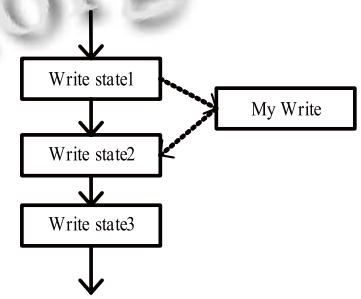

图 4　写操作拦截模型

写操作正常过程是 Write state1(Write state2(Write state3，通过修改跳转状态,使得执行过程为 Write state1(My write(Write state3,拦截者通过 My Write 实现需要的功能，甚至结束整个 write 过程. 任何一个状态的切换过程都可能被拦截，这种拦截技术称 hook 技术[1,4]. 在 windows 发布的 WDF 框架下，通过分析文件

过滤驱动范例 SfILTER[4]，以写操作为例，对用户请求处理的框架描述如图 5 所示[1].

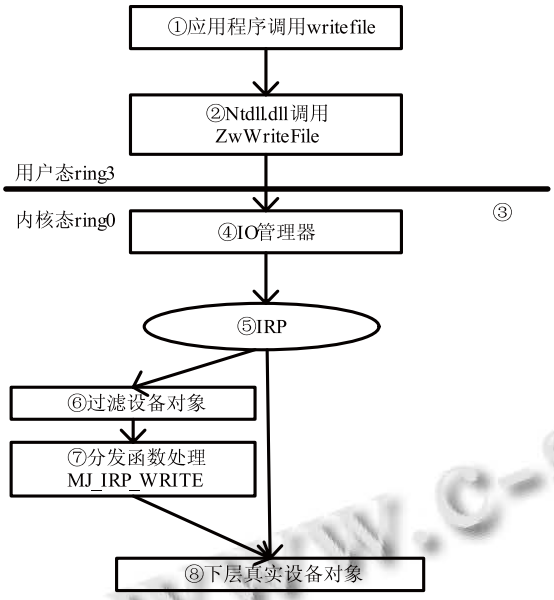

图 5 WDK 驱动框架写操作拦截模型[5]

① 应用程序使用另一个库中函数时，须导入该函数地址，组成一个 IAT 表(Import Address Table),通过修改 IAT 表中 writefile 的地址，就可以实现对 writefile 的 hook.

② ntdll.dll 是系统由用户态转入内核态的入口，对系统的写操作在此转化成 ZwWriteFileGater 或 ZwWriteFIle，可采用同样的方法对这两个函数进行 hook.

③ 系统由用户态进入内核态过程中，存在一个服务调度表(System Service Dispatch Table, SSDT)，用于索引函数在内核中的地址. 通过修改 SSDT 表格，同样可以实现 hook.

④ 利用 WDF 驱动框架，生成过滤设备对象，该对象可在真实设备对象处理 IRP(系统请求的数据结构)之前先处理 IRP，甚至可以直接填写这个结构或者禁止这个请求.

通过分析，windows 捕获写文件操作和创建文件操作的时机非常多. 本文采用在⑦实施拦截. 首先，该位置为驱动层，在驱动层开发比应用程序拥有更高权限；其次，在其他位置拦截写操作和一些病毒行为类似，会引起杀毒软件误判和系统不稳定；另外，在底层拦截更为彻底，编写代码简洁. 与传统防御系统相比，WKD 过滤框架规则简单有效，稳定性较好，拦截足够彻底. 虽采用严格主动防御手段，但不会给正常程序的使用带来干扰，误杀率较低.

3 PEPS实现与评估

3.1 PEPS 实现

PEPS 如图 6 所示.

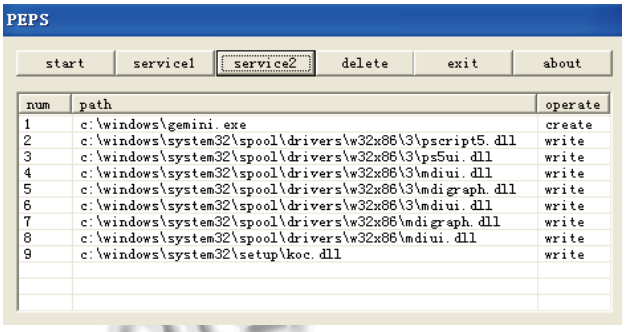

图 6 PEPS 界面及运行效果

PEPS 基于 windows xp 系统，在 vmware 虚拟机中测试通过. 开发环境为 vmware workstation, VC6.0, WDK 7600.16385.1. 分为应用程序和驱动程序两个功能模块. 应用程序与用户交互，初始化驱动的服务类型，显示驱动信息. 驱动程序生成文件过滤设备，拦截非法写操作和对创建操作做分析，将异常操作路径发送应用程序显示.

应用程序的按钮功能说明如下:

start: 连接驱动服务.

service1: 开启服务类型 1，只监控 create 和 write 行为，不做拦截.

service2: 开启服务类型 2，监控 create 行为，拦截对系统已有文件的 write 行为.

delete: 如果发现非法 write 行为，可以对 create 的文件做删除操作. 清除病毒体.

exit: 关闭驱动服务，退出程序.

about: 版本信息.

num: 拦截操作序号.

path: 对应操作的路径.

operate: 对应操作的类型，service2 中对 create 只是监控，对非法 write 进行拦截，发现有非法 write 操作，通过执行 delete，可以删除 create 的所有文件.

在计算机常态下，PE 文件内容不会被改写. 这个假设并非绝对正确，笔者开启 PEPS 后进行简单的文档操作，执行浏览网页，聊天等，对 2.1 中假设前提做以下简单修正: ① pagefile.sys 是 Windows 下的虚拟内存，其作用与物理内存基本相似，作为物理内存的“后备力量”而存在，其大小是变化的. ② C:\DocumentsandSettings\Administrator\Local Settings\

Temp 目录下的文件作为缓存文件, 很多安装程序往往在此目录下生成 install.exe, 发生改写的概率较大.

对 “①”的改写通过特判允许, 对于 “②”在文件执行前对该目录进行清空. 经过特殊处理, “A_假设”仍然合理有效. 对于 “B-假设”, 应用程序安装过程虽然释放 PE 文件, 然而不会改写系统本来 PE 文件. 为此, 我们验证了十个正常程序(这些程序都是第一次在 PEPS 执行), 记录下这些程序创建的 PE 文件个数和改写系统原有 PE 文件个数如表 2 所示.

表 2　正常程序文件写操作和创建操作

正常安装程序	create	write
WinRAR	8	0
GoogleTakl	4	0
CodeBlocks	88	0
Qqplayer	81	0
T 盘	17	0
有道词典	18	0
Gvim	10	0
福昕阅读器	7	0
iku	25	0
金山毒霸	161	0

3.2 实验评估

基于 PEPS 防御特点, 在 http://vx.netlux.org/收集五个典型的感染型病毒即 antares, atix, carume, Gemini 和 seraph 来验证 PEPS 系统的防御能力. 这些病毒都可找到对应源代码, 同时能够运行于 vmware 虚拟机上的 windows xp 系统. 感染目标是 windows 系统的 PE 文件. 为了验证 PEPS 防御能力, 选用如下杀毒软件来做对比: ① 金山毒霸 2012(猎豹) (2012.SP2.2012.03. 11. 20 病毒库); ② 360 杀毒(3.0.0.2121 病毒库); ③QQ 管家 6.6(6.8.2325.201 病毒库); ④ ESET Smart Security (2012.03.12 病毒库); ⑤ 小红伞(2012.02.26 毒库).

对比指标: ① 病毒运行之后, 各个软件是否能检测到病毒, “能”则显示 Yes, 否则 Not; ② 在各个软件保护下病毒是否对系统文件造成破坏, “是”则显示 Yes, 成功保护显示 Not. 测试结果如表 3 和表 4 所示.

从实验结果可以看出, PEPS 防御思想在对抗以系统 PE 文件为目标的 Win32 感染型病毒上相当有效. 常规的主动防御往往在系统文件被感染之后才能检测到感染行为. 本文通过实验仿真和分析, 验证了系统原有 PE 文件不变的特性, 并且根据该特性设计出主动防御系统 PEPS, 保护了系统文件, 降低病毒风险.

表 3　主流杀毒软件是否能检测到感染行为

	金山	360	ESET	QQ 管家	小红伞	PEPS
antares	Yes	Not	Yes	Not	Not	Yes
atix	Not	Not	Not	Not	Not	Yes
carume	Not	Yes	Yes	Not	Not	Yes
gemini	Yes	Yes	Yes	Yes	Not	Yes
seraph	Yes	Yes	Yes	Not	Not	Yes

表 4　在各个防御软件的保护下, 系统文件是否被破坏

	金山	360	ESET	QQ 管家	小红伞	PEPS
antares	Yes	Yes	Not	Yes	Yes	Not
atix	Yes	Yes	Yes	Yes	Yes	Not
carume	Yes	Yes	Not	Yes	Yes	Not
gemini	Yes	Yes	Not	Yes	Yes	Not
seraph	Yes	Yes	Not	Yes	Yes	Not

4 总结

PEPS 针对 win32 PE 感染型病毒设计, 对于非感染型病毒尚不具有检测功能. 设计重心放在了对文件系统的保护上, 后续工作主要是对注册表的关键路径实现监控; 防止病毒通过 rootkits[1]提前进入内核常驻系统. 完成以上两点, 病毒的感染行为将得到有效监控, 主动防御的效果将更佳. PEPS 采用较为严格的主动防御方法, 虽通过实验验证, 对正常程序干扰很少, 但还是存在一定影响, 比如打补丁, 应用程序升级等会造成 PE 文件发生变化, 对此应该通过特殊判断允许其进行.

参考文献

1 Hoglund G, Butler J. Rootkits-Windows 内核的安全防护中文.韩智文译.北京:清华大学出版社,2007.
2 葛长涛.Win32 PE 文件病毒行为分析与检测.解放军信息工程大学,2010.36–46.
3 王成,庞建民,赵荣彩,王强.基于可疑行为识别的 PE 病毒检测方法.计算机工程,2009,(15):132–134.
4 杨阿辉,陈鑫昕.基于 SSDT 的病毒主动防御技术研究.计算机应用与软件,2010(10):288–290.
5 张佩,马勇,董鉴远.深入浅出 Windows 驱动开发.北京:电子工业出版社,2011.88–106.
6 masepu.远程文件捆绑器的原理和实现.黑客防线,2010:240–241.
7 段钢.加密与解密.第 3 版.北京:电子工业出版社,2008.263–304.