

# PCI 设备的安全性分析

沈永军, 祝跃飞, 张长河

(解放军信息工程大学信息工程学院, 郑州 450002)

**摘 要:** 对 PCI 扩展 ROM 规范进行分析, 研究其存在的安全隐患, 在此基础上, 对计算机系统进行渗透性攻击, 并提出隐患检测和防护措施。利用扩展 ROM 代码在完整性保护上存在的缺陷, 将恶意代码写入扩展 ROM 中, 通过该恶意代码篡改系统的启动模块, 达到攻击系统内核、获得系统权限的目的。实验结果表明, 渗透性攻击能实现对计算机系统的控制, 防护措施能确保系统安全。

**关键词:** PCI 设备; 扩展 ROM; 完整性保护; 启动过程; 中断向量

## Security Analysis of PCI Device

SHEN Yong-jun, ZHU Yue-fei, ZHANG Chang-he

(Institute of Information Engineering, PLA Information Engineering University, Zhengzhou 450002, China)

**【Abstract】** This paper deeply analyzes the secure hidden danger of PCI extended ROM and presents a method to penetratively attack control the computer, system by using this secure vulnerabilities, and proposes the measure for hidden danger detection and prevention. The method writes the malicious codes into extended ROM, using the integrity protection fault of extended ROM codes. The malicious code in extended ROM can tamper system startup module and attack the system kernel to get the system permission. Experimental results show that the attack can control the operating system, detection and prevention measure can protect the system security.

**【Key words】** PCI device; extended ROM; integrity protection; startup procedure; interrupt vector

DOI: 10.3969/j.issn.1000-3428.2011.19.036

### 1 概述

外围设备互连(Peripheral Component Interconnect, PCI)描述一组与外设及主板上的其他总线互连时的计算机总线的标准<sup>[1]</sup>。在目前计算机系统中得到广泛应用。PCI 设备是连接在 PCI 总线上的设备, 常见的 PCI 设备有显卡、网卡和存储控制器等。

PCI 设备作为现代计算机系统的一部分, 系统 BIOS 在机器加电启动过程中将遍历系统中存在的所有 PCI 设备, 执行 PCI 设备的扩展 ROM 代码, 用以进行卡设备的自检及初始化。由于扩展 ROM 代码在操作系统引导前执行, 并且扩展 ROM 代码可以对计算机系统的启动流程进行更改, 因此, PCI 设备代码的安全性以及代码能否安全执行对整个计算机系统的安全性有重要意义。

文献[2-3]提出将 Rootkit 程序写进 ROM 芯片并对系统进行渗透的思路, 但这并没有指出 PCI 设备存在安全隐患的根源。本文通过对 PCI 规范进行详细分析, 挖掘出 PCI 设备安全隐患存在的根本原因, 并对检测和消除 PCI 设备的安全隐患进行一些探讨。

### 2 PCI 扩展 ROM 初始化

#### 2.1 扩展 ROM 代码格式

PCI 扩展 ROM 代码是 PCI 设备用来进行设备初始化或进行系统引导功能的代码。扩展 ROM 代码包含 PCI 扩展 ROM 头、PCI 数据结构、初始化代码和运行时代码 4 个部分, 统称为一个 ROM 映像。为了适应不同的处理器结构, 扩展 ROM 中可以保存多个处理器结构对应的 ROM 映像<sup>[4]</sup>, 且每个 ROM 映像专门针对某个系统或者处理器而设计的功能代码, ROM 映像要保证其长度是 512 Byte 的整数倍, 并且具有正确的校验和。扩展 ROM 代码格式如图 1 所示。

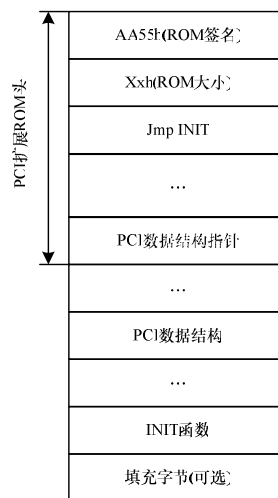


图 1 扩展 ROM 代码格式

扩展 ROM 映像偏移 03h 处保存着映像文件的初始化函数跳转指令, 该指令会被 BIOS 调用, 并将计算机的控制权传递给扩展 ROM 映像的初始化函数进行 PCI 设备的检测和初始化操作。初始化函数执行完后将计算机控制权返回给 BIOS 之后, 继续进行后续的操作系统系统引导, 由此扩展 ROM 映像文件的初始化函数可以对系统引导过程进行篡改。

#### 2.2 PCI 扩展 ROM 初始化流程

计算机系统启动后首要工作便是系统自检, 即跳转到

**基金项目:** 国家“863”计划基金资助项目(2008AA01Z420)

**作者简介:** 沈永军(1986—), 男, 硕士研究生, 主研方向: 信息安全; 祝跃飞, 教授、博士; 张长河, 讲师

**收稿日期:** 2011-04-13 **E-mail:** jjgogo0906@gmail.com

BIOS 启动代码处, 执行 BIOS 代码, 开始进行系统自检 (Power on Self Test, POST)。在 POST 过程中, POST 代码配置一个 PCI 扩展 ROM<sup>[5]</sup>的步骤描述如下:

(1)POST 代码扫描 PCI 总线拓扑, 检测系统中 PCI 设备, 若设备在配置空间中部署了扩展 ROM 基地址寄存器, 则 POST 代码在一个未被占用的内存空间中使能扩展 ROM。

(2)POST 代码检测扩展 ROM 的前 2 Byte 是否是 AA55 的 ROM 标签。若 ROM 标签存在, 则该 PCI 设备存在扩展 ROM, 否则不存在扩展 ROM。若扩展 ROM 存在, 则 POST 代码在扩展 ROM 中搜索一个具有合适代码类型的 ROM 映像。

(3)POST 代码确认厂商标志和设备标志 2 个域是否和设备中相应的域值匹配。

(4)将 ROM 中的映像拷贝到 RAM 中, 调用扩展 ROM 映像中的初始化函数跳转指令。

(5)在扩展 ROM 代码执行前, 扩展 ROM 映像的初始化函数会计算映像文件的校验和对文件进行完整性校验, 如果校验和正确, 那么执行扩展 ROM 映像文件代码, 否则扩展 ROM 映像文件不被执行, 将控制权返回给 BIOS。

在上述扩展 ROM 配置执行的过程中, 是通过一些简单的措施例如检查映像中是否含有 AA55 的 ROM 标签; 检查设备标志和厂商标志是否和设备对应; 判断代码是否符合扩展 ROM 代码格式以及是否和设备匹配<sup>[6]</sup>; 计算扩展 ROM 映像的校验和是否正确判断扩展 ROM 映像文件的完整性。这些措施不能保证映像文件的安全性。下面对扩展 ROM 的安全性进行分析。

### 3 PCI 扩展 ROM 安全性分析

#### 3.1 扩展 ROM 完整性保护存在的缺陷

扩展 ROM 的代码规范要遵循 PCI Specification 和 Plug and Play BIOS Specification 要求。扩展 ROM 映像采用的是 8 bit 校验和算法作为 ROM 映像的完整性校验机制, BIOS 在执行代码前, 通过扩展 ROM 中的 INIT 函数来计算这个 ROM 映像的 8 bit 校验和, 如果结果为 0, 那么 BIOS 认为该 ROM 映像是完整的, 则执行 ROM 映像的初始化函数, 并将控制权传递给 ROM 映像。仅通过校验和对 ROM 进行完整性保护, 不能保证 ROM 映像的完整性不受破坏。即对 ROM 代码进行篡改后, 通过校验和算法重新计算 ROM 的校验值, BIOS 对 ROM 映像进行完整性校验, 若校验值匹配, 则该代码会被 BIOS 执行。图 2 为 PCI 扩展 ROM 代码注入框架。

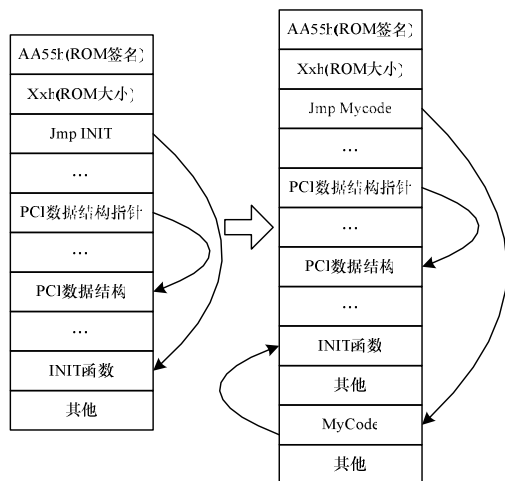


图 2 PCI 扩展 ROM 代码注入框架

图 2 显示了新的功能函数被写入到扩展的 ROM 映像空余空间中过程。原始的 PCI 初始化过程函数指针被更改为新的功能函数地址。新的功能函数执行完成以后, PCI 的初始化过程函数被调用, 以完成正常的设备初始化工作。

#### 3.2 ROM 芯片扩展刷写问题

扩展 ROM 芯片分为 ROM、PROM、EPROM、EEPROM、FLASH ROM 等。目前大部分扩展 ROM 采用 EEPROM 和 FLASH ROM 这 2 种芯片类型, 且这些类型的芯片能够在操作系统下被刷写, 黑客可利用该特性, 将恶意代码写入到扩展 ROM 芯片中。

不同的操作系统有不同的软件层, 在 x86 平台下, 对扩展 ROM 映读写操作的逻辑步骤基本相同, 这是由 x86 构架的编程模型决定。大部分 x86 构架操作系统的应用程序, 使用硬件提供的 2 个优先级来访问系统资源, 这 2 个优先级是 ring0 或者内核模式、ring3 或者用户模式。任何运行在内核模式的软件可自由访问和操纵硬件。在操作系统下访问扩展 ROM 芯片的步骤如下:

(1)在操作系统中, 通过设备驱动程序访问扩展 ROM 芯片的地址空间, 由于设备驱动程序具有操作系统的内核模式的访问权限, 因此能直接访问扩展 ROM 芯片的地址空间。

(2)进行特定硬件操作来访问和操纵扩展 ROM 芯片的内容, 在这个步骤中, 需要了解访问扩展 ROM 芯片时的硬件细节, 该硬件的具体细节可通过芯片的数据表获得。硬件操作方法如下:

1)配置芯片寄存器, 使能对扩展 ROM 芯片进行读写操作。

2)在预定义好的地址空间中读出扩展 ROM 芯片的厂商标志和设备标志, 以探测扩展 ROM 芯片, 并通过这些标志字节确定访问 ROM 芯片的方法。

3)通过厂商的规范读写芯片的内容。

### 4 ROM 安全隐患的危害性及测试

#### 4.1 扩展 ROM 安全隐患危害性分析

当 BIOS 执行 PCI 扩展 ROM 时, BIOS 对机器的初始化已经进行完毕, 执行环境是实模式, 且内存及各种中断向量和硬件资源均可以被使用。

攻击者可以改变中断向量例程, 具体如下所述:

在实模式下, 当中断被触发时, 将会运行扩展 ROM 代码中被安置好的代码, 从而实现对 CPU 执行流程的控制。

如扩展 ROM 代码在 BIOS 调用 INT 19h 时去 HOOK INT 13h 中断服务<sup>[7]</sup>(其中, INT 19h 是 BIOS 引导自举中断服务, 负责将主引导扇区读到内存然后把控制权交给主引导扇区; INT 13h 是 BIOS 的磁盘操作服务中断, 可以为操作系统提供磁盘的读写等服务)。

当 BIOS 调用 INT 13h 中断服务, 并且加载主引导扇区或者分区引导记录时, 扩展 ROM 代码先会被执行, 通过修改主引导扇区以及相邻的其他扇区, 将一段攻击代码嵌入主引导扇区的执行流程。

通过如上方式, 恶意代码在扩展 ROM 中获得执行权限, 对计算机的启动过程逐步进行攻击, 并且将控制权传递给操作系统中。

#### 4.2 实验方案设计

本文实验使用的 PCI 设备是 RTL 8139C 网卡及其扩展 ROM 芯片 Winbond W27C512。测试硬件平台配置如表 1 所示。

表 1 测试硬件平台配置

名称	配置
处理器	DualCore Intel Core 2 Duo E6300, 1 866 MHz
主板	Lenovo ThinkCentre M55
北桥	Intel Broadwater Q965
南桥	Intel 82801HB ICH8
显卡	Intel GMA 3000
PCI 设备	RTL8139C 网卡
RAM	2 GB

在 Windows XP sp3 下进行以下步骤: (1)计算 ROM 的剩余空间大小, 将攻击内核的功能代码写入 ROM 的空余空间, 并保存该空间的首地址。(2)将入口点函数指针 INIT 的地址偏移替换为步骤(1)保存的地址偏移, 并保存 INIT 函数地址, 在功能代码执行后调用 INIT 函数, 以完成正常的设备检测和初始化工作。(3)计算修改后的 ROM 映像大小, 判断大小是否是 512 Byte 的整数倍, 如果不是, 那么在 ROM 映像尾部填充 0, 使其符合判断标准。(4)重新计算校验和, 将校验和写入到 ROM 映像中, 即新的 ROM 映像生成。

插入到扩展 ROM 映像中的功能代码在扩展 ROM 执行时获得执行, 该功能代码挂钩中断向量表的 13h 号中断服务例程, 对磁盘操作进行监控, 实现对后续加载的操作系统启动文件以及系统内核进行攻击。

功能代码在系统启动后保持运行, 且具有系统权限。由于代码在操作系统启动前获得计算机的控制权, 具有很强的隐蔽性, 对安全软件的查杀造成很大的威胁, 因此将给系统带来较大安全隐患。扩展 ROM 的整个攻击过程如图 3 所示。

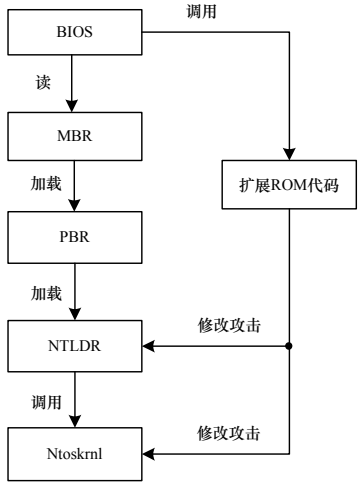


图 3 扩展 ROM 攻击模型

5 检测及防护措施

由上文分析可知, PCI 设备之所以存在安全隐患, 根源在于以下 2 点: (1)对扩展 ROM 代码的完整性校验机制不完善, 使得扩展 ROM 映像被改写后仍可以被执行。(2)对扩展 ROM 的刷写没有认证机制, 即任何人都可以改写扩展 ROM 映像。

可通过审计的方法检测扩展 ROM 是否被修改。在设备厂商的官方网站上获得一份原始的 ROM 程序。读取 PCI 卡上的 ROM 代码, 将得到的 2 份 ROM 代码进行比较, 由于扩

展 ROM 在由 ROM 向 RAM 中复制时, 可能会将代码的初始化部分丢弃, 因此 2 份 ROM 的大小或内容上的存在一些差异, 并不能确定要比较的 ROM 代码是否被修改。通过分析 ROM 是否具有某些特征来进一步判定要比较的 ROM 代码是否被更改。如查看扩展 ROM 中是否包含保护模式代码: 因为扩展 ROM 代码是运行在实模式下, 所以扩展 ROM 中不应该包含保护模式代码; 还可以通过检查扩展 ROM 是否对中断向量表进行更改。如果设备挂钩了其自身功能无关的中断向量, 那么该扩展 ROM 可能包含恶意代码; 如果扩展 ROM 中的代码被反汇编以后包含很多难以理解的代码, 则说明扩展 ROM 代码以某种方式编码过, 这也是值得怀疑的。

要从根本上消除 PCI 设备的安全隐患, 必须保证扩展 ROM 不被篡改, 最有效的方法就是通过物理的方法进行防护, 即通过将 PCI 设备添加物理开关对其进行写保护, 这样可有效预防扩展 ROM 被更改, 但同时也增加了设备的复杂性和成本。另外可信平台模块<sup>[8]</sup>的安全芯片也可以用来消除 PCI 扩展 ROM 的安全隐患。

6 结束语

本文分析 PCI 设备扩展 ROM 映像文件中存在的安全隐患, 提出针对该安全隐患的攻击方案, 并提出隐患检测和防护措施。对 RTL 8139 网卡设备的扩展 ROM 映像文件进行攻击, 并将恶意代码写入到扩展 ROM 映像中。实验结果表明, 恶意代码能在扩展 ROM 中运行, 并对系统启动模块进行攻击, 从而获得系统权限, 以实现对系统的控制。对于当前计算机系统的安全防御而言, 不仅可以通过实施一些常规方法, 如及时给系统以及第三方软件打补丁, 或者及时升级病毒库, 还需要对 PCI 等外围设备加以密切关注, 这将是以后的研究重点。

参考文献

[1] Tom S, Don A. PCI System Architecture[M]. [S. l.]: MindShare Inc., 1999.

[2] John H. Implement and Detecting an ACPI Bios Rootkit[C]//Proc. of Black Hat. [S. l.]: IEEE Press, 2006.

[3] John H. Implementing and Detecting a PCI Rootkit[EB/OL] (2010-10-20). <http://feishare.com/pci/ref-implementing-and-detecting-a-pci-rootkit>.

[4] PCI-SIG. PCI Firmware Specification Revision 3.0[EB/OL]. (2010-10-20). [http://www.pcisig.com/specifications/conventional/pci\\_firmware](http://www.pcisig.com/specifications/conventional/pci_firmware).

[5] Compaq, Phoenix. BIOS Boot Specification Version 1.01[EB/OL]. (2010-10-21). [http://www.phoenix.com/docs/specsbbs\\_101.pdf](http://www.phoenix.com/docs/specsbbs_101.pdf).

[6] 李清宝, 孟庆倩, 曾光裕. 基于扩展 ROM 技术的网络安全隔离卡设计[J]. 计算机工程, 2008, 34(1): 281-282.

[7] 邓立丰, 王宏霞. 基于协同特征的 BIOS Rootkit 检测技术[J]. 计算机工程, 2010, 36(15): 134-136.

[8] Trusted Computing Group. TPM Main Part 2 TPM Structures[EB/OL]. (2010-10-21). <https://www.trustedcomputinggroup.org/specs/TPM/mainP2Structrev103.pdf>.

编辑 刘冰