太行安全 BIOS 可信体系结构与实现研究

周振柳[1,3],李 铭[2],许榕生[1],宋东生[3]
ZHOU Zhen-liu[1,3],LI Ming[2],XU Rong-sheng[1],SONG Dong-sheng[3]

1.中国科学院 高能物理研究所 计算中心,北京 100049
2.中国电子科技集团 信息化工程总体研究中心,北京 100083
3.沈阳航空工业学院,沈阳 110034
1.Computing Center,Institute of High Energy Physics,Chinese Academy of Sciences,Beijing 100049,China
2.Center of Information System Architecture Research,CETC,Beijing 100083,China
3.Shenyang Institute of Aeronautical Engineering,Shenyang 110034,China
E-mail:zhouzl@ihep.ac.cn

ZHOU Zhen-liu,LI Ming,XU Rong-sheng,et al.Research on trusted architecture and implementation of Taihang secure BIOS.Computer Engineering and Applications,2008,44(18):76-79.

Abstract: Security requirements of trusted BIOS are analyzed,and architecture of trusted BIOS is developed in this paper.To construct Pre-OS chain of trust,message digest and digital signature are used to verify integrity and authenticity of entities in different phases of bootstrap.Trusted components,workflow and methods of implementation of Taihang secure BIOS are introduced in detail.Performance of trusted measurement is also analyzed at the end.

Key words: trusted BIOS;trusted measurement;security requirements;chain of trust

摘 要: 分析了可信 BIOS 的安全需求,提出一种可信 BIOS 体系结构。使用消息摘要和数字签名技术验证系统引导阶段实体的完整性和真实性,构建了操作系统运行前的信任链。介绍了太行安全 BIOS 的可信部件、执行流程及其实现方法,讨论了可信测量对 BIOS 引导过程的性能影响。

关键词: 可信 BIOS;可信测量;安全需求;信任链
DOI: 10.3778/j.issn.1002-8331.2008.18.024 **文章编号:** 1002-8331(2008)18-076-04 **文献标识码:** A **中图分类号:** TP393.08

1 引言

可信计算环境要求从硬件到软件,组成系统的各部分都是安全可信的,任何不能证明自身是安全可信的部件,例如恶意代码、病毒、木马等,都被排斥在这个系统之外。这种计算和网络环境称为健康的可信计算生态环境。建立可信计算生态环境,需要从加强计算终端的硬件、固件、操作系统等底层安全入手,从源头上控制不安全因素。

传统 BIOS 设计没有考虑安全问题,存在较多安全隐患:CIH 病毒利用固件 BIOS 可以在操作系统环境下软件修改的缺陷对 BIOS 进行拒绝服务攻击;文献[1]描述了一种在 BIOS 中实现 Rootkit 恶意代码的方法;互联网上在局部范围内出现向某些版本 BIOS 系统内植入木马的 BIOS 系统组合攻击工具包。伴随固件 BIOS 技术发展对 BIOS 在线更新升级管理的要求和应用扩展,固件 BIOS 面临越来越多的安全威胁,并且进一步危及操作系统安全。

William A.Arbaugh 在 1997 年提出一种计算机安全引导架构 AEGIS[2]。AEGIS 基于 IBM PC 传统 BIOS,采用认证方法保障固件 BIOS 代码的完整性,增强 BIOS 引导过程中代码的安全保护。AEGIS 缺乏硬件保护措施,也没有考虑固件层对系统软件层的延伸保护。文献[3]提出将 BIOS 系统融合进 TPM 芯片的方法,虽然能够保障 BIOS 安全,但需要对现有的计算机体系结构作较大的改变。文献[4]研究了 Linux 操作系统的可信启动与测量问题,即引导工具(Initial Program Loader,IPL)对 OS 的完整性检测和保护,但只建立了可信链的一部分,未涉及对 BIOS 层的初始可信链建立的研究。Dexter Kozen 在 1998 年提出一种基于语言证明的方法[5],在编译过程中检查代码控制流安全性、内存访问安全性、堆栈操作安全性以增强代码安全,并将该方法用于对固件 BIOS Open Firmware 中的恶意代码检测[6]。由于语言证明方法的复杂性,这种方法尚不能应用于实际中。

可信计算组织(Trusted Computing Group,TCG)提出可信计算平台概念,以计算平台硬件安全可信为基础,以“信任传递”和“完整性测量”为手段,从硬件层安全着手解决信息安全问题[7]。本文以构建可信计算终端为出发点,通过分析可信BIOS的安全需求,提出一种可信 BIOS 体系结构。信产部“基于 EFI

基金项目: 信产部电子信息产业发展基金资助项目;北京市教委科技发展计划项目(No.KM200610772006)。
作者简介: 周振柳(1971-),男,博士生,研究方向为网络安全、可信计算;李铭(1951-),男,博士,研究员,研究方向为信息安全与计算机司法检验;许榕生(1947-),男,博士生导师,研究员,研究方向为互联网和网络安全;宋东生(1979-),男,工程师,研究方向为网络攻击与防范。
收稿日期: 2007-09-20 **修回日期:** 2007-11-12

的新一代安全 BIOS 与产业化”和“高安全与可管理 BIOS”基金项目使得本文的实现得到 Intel 的新一代 EFI BIOS 代码的支持,使本文可信 BIOS 的原型产品—太行安全 BIOS 得以实现。

2　可信 BIOS 安全需求

可信 BIOS 的安全需求，源自于实际应用中 BIOS 面临的安全威胁和可信计算实施[7,9]对 BIOS 的需求。BIOS 产品的在线补丁、升级、更新等促使主板厂商普遍采用 FLASH 芯片存储 BIOS 系统固件,由此导致第三方恶意者向 BIOS 中植入恶意代码、病毒、木马等成为可能[1]。攻击者还可破坏 BIOS 代码的完整性,通过修改 BIOS 系统少量字节代码使得计算机系统不能正常引导和启动，达到拒绝服务攻击目的。此类典型攻击为 CIH 病毒及其系列变种。采用对 BIOS FLASH 芯片的物理写保护或双 BIOS 机制能一定程度减少这种安全危害,但并不是一个完整的解决方案。可信 BIOS 的一种安全需求,是阻止攻击者植入 BIOS 芯片中的恶意代码的执行，保证 BIOS 系统的执行代码只来自可信任的 BIOS 厂商、硬件驱动厂商等,保障 BIOS 系统自身代码和数据的完整性。可信 BIOS 这种对自身完整性保护的安全需求,符合 Clark-Wilson 完整性模型的三个目标[8]。

可信计算终端的实施,要求从可信硬件开始,从底层到上层软件,每一步骤的执行都能够通过信任传递,建立可信链。信任传递是通过可信测量来实现的。可信测量是指对被测量对象完整性和真实性进行校验的过程。CPU 加电后执行的指令代码依次为 BIOS→IPL→OS→Application,在这个过程中,BIOS 负责对下一层的 IPL 进行可信测量,IPL 负责对 OS 进行可信测量,逐层建立信任链[9]。对 IPL 的完整性和真实性的可信测量,是可信 BIOS 的另一种安全需求。这里的 IPL 特指 MBR 或其他引导操作系统的引导软件(如 Lilo、Grub 等)。

可信测量虽然能够保障非法代码不被执行,但不能阻止对可信 BIOS 的拒绝服务攻击。当由于不可预知的故障或攻击导致 BIOS 部分代码或数据完整性遭到破坏时,BIOS 系统自身必须具备安全可靠的失败自恢复机制。实施这种失败自恢复机制的 BIOS 代码必须受到硬件保护,保证失败自恢复机制自身不会遭到破坏。恢复过程同样需要对所提供的恢复内容进行可信测量,这个过程称为可信恢复。可信恢复是可信 BIOS 的第三种安全需求。

3　可信 BIOS 体系结构

可信 BIOS 采用模块化结构,除具备普通 BIOS 的功能外,还必须满足第 2 章中提出的安全需求。图 1 描述了可信 BIOS 体系结构。

可信 BIOS 由两部分构成:CRTM(Core Root of Trust for Measurement)和 N-CRTM(Non-CRTM)。CRTM 是可信链建立的起点,是被无条件信任的。CRTM 要满足 4 个约束条件:(1)CRTM 具备对后续执行模块进行可信测量的能力;(2)CRTM 自身受到物理保护，不能通过软件写方式更新和升级;(3)当 N-CRTM 完整性被破坏时,CRTM 具备对 N-CRTM 进行可信恢复的能力;(4)CRTM 只包含满足可信测量和可信恢复功能执行环境初始化的代码最小集。满足上述约束条件的可信 BIOS 的 CRTM 包含 4 个功能模块:平台初始化模块完成 CPU、芯片组、主板、内存、堆栈的初始化工作,建立后续 BIOS 程序最小化运行环境;可信硬件驱动模块初始化可信硬件设备,提供

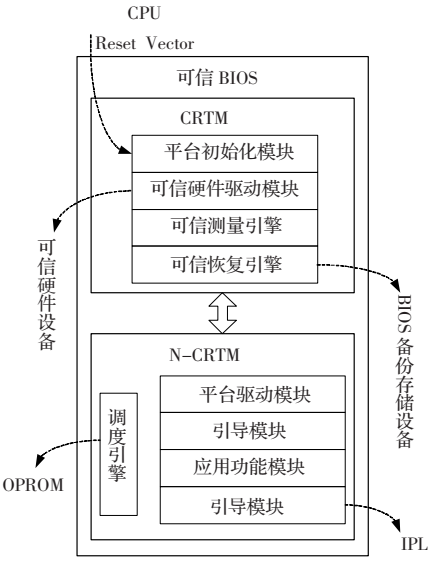

图 1　可信 BIOS 体系结构

可信硬件设备调用的协议函数;可信测量引擎为后续可信测量提供消息摘要和非对称加密引擎;可信恢复引擎在 N-CRTM 完整性遭到破坏时驱动 BIOS 备份存储设备，并从该设备中对可信 BIOS 的 N-CRTM 进行可信恢复。

可信 BIOS 的 N-CRTM 完成 BIOS 固件的其他功能，包括对硬件平台的进一步检测和初始化、设备驱动、模块调度、系统管理、板卡上 OPROM(Option ROM)固件代码的调用执行,操作系统引导代码等。用于安全/管理的其他 BIOS 层的应用程序也可在 N-CRTM 阶段加载执行。N-CRTM 包含的代码必须通过可信测量才能够被加载执行。

系统开机上电后,CPU 初始指向并执行可信 BIOS 的 CRTM 包含的指令代码,CRTM 执行的最后会对非 N-CRTM 的调度引擎进行可信测量。测量成功则执行调度引擎,测量失败说明加载代码不可信(遭到修改或破坏、或来源不可信),则 CRTM 会启动可信恢复引擎要求用户对 BIOS 系统进行可信恢复。调度引擎加载后续的 N-CRTM 的其他模块时，需调用 CRTM 提供的可信测量引擎提供的加解密协议对其进行可信测量。测量成功则执行下一个模块,否则调用 CRTM 的可信恢复引擎作可信恢复。可信 BIOS 执行的最后一个阶段是由可信 BIOS 引导模块对 IPL 进行可信测量，成功后进入操作系统引导阶段。

可信 BIOS 采用可信硬件模块作为可信测量的存储根核。可信硬件模块中存储可信测量需要用到的各种密钥,是计算机可信的硬件基础。可信硬件模块可以采用 TPM 或 IKEY 智能设备。BIOS 备份存储设备中存储着用于可信恢复的 N-CRTM 部分的映像。

4　可信 BIOS 的可信测量

可信 BIOS 使用数字签名和消息摘要实现对各种实体的可信测量。在可信 BIOS 生产过程中，要求对可信 BIOS 中 N-CRTM 的各个模块进行可信封装。可信封装是指将用于可信测量的包含实体消息摘要的数字签名同实体代码/数据一起按预定义格式进行封装。可信测量过程则根据封装实体中的数字签名,解签计算得到包含在签名中的消息摘要,对实体重新计算消息摘要,通过两者的比较判定实体的完整性。由于公私钥具

有对应唯一性,因此可同时判定实体是否来自合法的厂商。可信封装和可信测量的原理如图 2 所示。

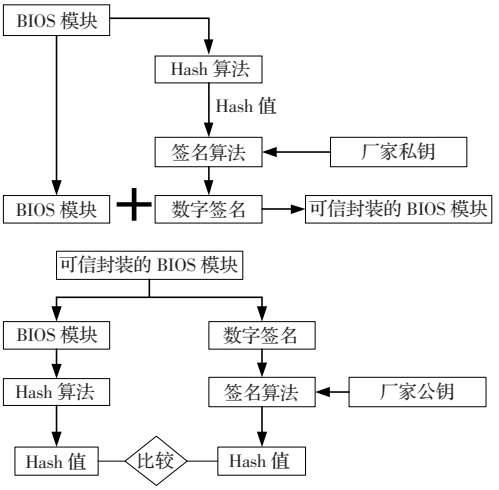

图 2 可信封装和可信测量

考虑下面的一种攻击情形:攻击者使用自己的私钥对恶意模块进行签名,并使得可信 BIOS 在运行过程中能够使用对应的公钥进行解签校验,则会得到成功的可信测量结果,导致攻击者可向 BIOS 中植入恶意模块并成功运行恶意代码。为防止此种攻击的出现,用于可信测量的解签公钥不能以数字证书的形式同被测量模块一同存放,而必须密封保存在可信硬件模块中进行保护。通常意义下非对称加密的公钥不需受到保护,可信 BIOS 的可信测量在这点上同一般情况有所区别。可信测量过程用到的数字证书,必须事先通过安全途径导入到可信硬件模块中。

对于可信 BIOS 外部实体(如板卡上的 Option ROM、IPL 等)的可信测量情况比较复杂。如果这些外部实体按照可信测量要求进行了可信封装,则可按照上述测量方法进行;否则需要计算机系统在初次运行时进行可信初始化。这里的可信初始化过程是指对不包含数字签名的实体计算其消息摘要并保护这些信息用于以后系统重启过程的可信测量。

5 可信 BIOS 的一种实现

2003 年 INTEL 向外界公布了其开发制订的新一代 BIOS 规范:EFI(Extensible Firmware Interface),并推出可用的 EFI BIOS 产品。其后 Intel 联合业界采用开源方式,共同制定推出 UEFI 规范[10]和符合 UEFI 规范的开源 BIOS 框架基础代码(www.tianocore.org)。我国信息产业部通过电子信息产业发展基金项目促进和支持国内 BIOS 研发单位与 Intel 合作,研发国产化安全 BIOS,为本文的实现提供了试验平台。本文基于新一代 EFI BIOS 实现了一种可信 BIOS 产品—太行安全 BIOS:硬件主板采用 Intel D945G 芯片组的 FOXCOM 主板,可信硬件设备采用 SINOSUN 的 SSX35B 型的 TPM 产品,采用 USB 块存储设备保存用于可信恢复的可信 BIOS 的备份。EFI BIOS 产品几乎全部采用 C 代码实现,方便了可信测量中消息摘要和非对称加密算法的实现。

5.1 太行安全 BIOS 可信部件与执行流程

太行安全 BIOS 采用硬件安全芯片 TPM 作为可信测量的存储根核。TPM 中存储可信测量需要用到的各种密钥,是可信的硬件基础。

太行安全 BIOS 基于由 Intel 公司授权的 Intel Platform Innovation Framework for EFI[12](以下简称 Intel Framework)的新一代 BIOS 代码实现。Intel Framework 是符合 EFI 标准的 BIOS 的一种实现代码,由 7 个执行阶段构成:SEC、PEI、DXE、BDS、TSL、RT、AL。太行安全 BIOS 的 CRTM 部分由 Intel Framework 的 SEC 和 PEI 两个阶段合并构成,存储在 FLASH 芯片的高 64 K 块中,受到软硬件双重写保护。

基于 Intel Framework 实现的太行安全 BIOS 的可信部件结构与执行流程如图 3 所示。其中 Intel Framework 的 RT 是操作系统运行阶段的支持,AL 是操作系统返回 BIOS 的支持阶段,在图中省略了这两个阶段的表示。

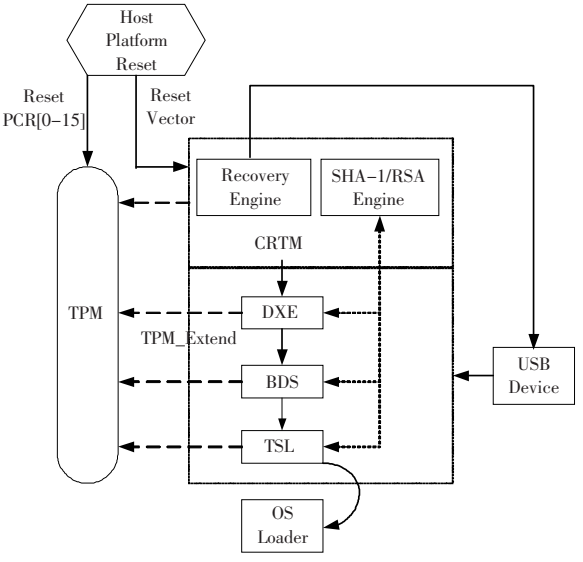

图 3 太行安全 BIOS 可信部件与执行流程

太行安全 BIOS 可信执行流程为:(1)执行 Initialize Code,做 CPU、chipset、Memory、Stack 初始化工作,建立 C 代码执行环境;(2)进入 PEI Core,调度 CRTM 其他代码,初始化 TPM,安装 TPM Protocol、Recovery Protocol、Hash Protocol;(3)PEI Core 对 DXE 代码进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行 DXE Core,否则进入可信恢复模式;(4)DXE Core 根据需要加载内部或外部的 EFI Driver、EFI Application,或设备的 Option ROM,并在加载过程中对被加载代码进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行代码,否则进入可信恢复模式;(5)DXE Core 对 BDS 代码进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行 BDS,否则进入可信恢复模式;(6)BDS 根据需要加载设备的 Option ROM,并在加载过程中对被加载代码进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行代码,否则进入可信恢复模式;(7)BDS 对 TSL 代码进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行 TSL,否则进入可信恢复模式;(8)TSL 对 OS Loader Code 进行可信测量,保存可信测量 Log。如可信测量成功,则加载执行 OS Loader,否则进入可信恢复模式。

OS Loader 开始执行后,操作系统执行前阶段的可信链建立成功,下一步就是 OS Loader 对操作系统内核的可信测量,不在本文讨论范围。

进入可信恢复模式后,CRTM 阶段安装的 Recovery Proto-

col 会被执行，负责从 USB 设备中读入除 CRTM 之外的可信 BIOS 的其他部分的映像,并写回到 Flash 芯片中。可信恢复的过程同样需要对被加载的 BIOS 映像进行可信测量,要求操作者提供完整合法的恢复映像。

5.2　TPM 的设置与管理

TPM 通过 LPC 接口同主板的南桥芯片组连接。在 CRTM 的最初阶段对 LPC 作初始化，读取 TPM 芯片的 VID(Vendor ID)和 DID(Device ID)[11],确定主板上是否存在 TPM,以决定运行过程中是否使用 TPM 进行可信测量。若检测到 TPM 存在,则向 TPM 发出 TPM_Startup(ST_CLEAR)和 TPM_ContinueSelfTest 命令使 TPM 进入完全操作状态,使 TPM 可用于后续的可信测量过程。为方便计算机用户对 TPM 进行管理设置,在 BIOS Setup 菜单中增加了对 TPM 的管理设置选项。

5.3　可信测量的实现

本文的实现采用的消息摘要算法为 SHA1 算法,生成 160 bit 的模块 Hash 值。签名和解签采用 2 048 bit 的 RSA 算法。模块封装的格式采用文献[12]定义的 GUIDed Section 类型。GUIDed Section 除包含一般 Section 的头部数据外,还允许包含一个自定义的 Section 头部，使用这个自定义头部存放用于可信测量的数字签名信息。

可信测量过程中要建立事件日志,并利用 TPM 的 PCR 累积保存事件日志条目的 Hash 值，通过 ACPI 表将事件日志传递给操作系统。用户在操作系统环境下利用事件日志可对 BIOS 引导过程进行重构和验证。

5.4　外部实体处理

需进行可信测量的 BIOS 外部实体包括板卡的 Option ROM、IPL。由于目前的 OPROM 和 IPL 均没有被可信封装,本文设计的对这些外部实体可信初始化处理方法为:系统初次可信初始化时,由可信 BIOS 自动生成一对用于外部实体验证的 RSA 密钥；可信 BIOS 计算外部实体的 Hash 值并使用 RSA 私钥对实体的 Hash 值进行签名,签名后抛弃 RSA 私钥;利用存储 BIOS 的 FLASH 芯片中 NVRAM 数据区保存外部实体的数字签名,利用 TPM 保护这个 RSA 公钥。第一次可信初始化完成后,以后每次启动就可实现对外部实体的可信测量。本文的实现，只对板卡的 Option ROM 和 MBR 这两种外部实体作可信测量。

5.5　可信测量对 BIOS 的性能影响

可信测量主要是做 SHA1 消息摘要的生成和密钥为 2048 bits 位的 RSA 解密操作。RSA 加密操作(即签名过程)需要占用大量时间,但加密过程只在生产过程中使用,可信 BIOS 正常运行过程中则只有解密操作。表 1 是在 1 GHz Pentium 机上,使用 SHA1 算法和不同长度公钥的 RSA 算法对 1 500 000 Byte 长度的数据进行一次数字签名校验过程所花费的时间[13]。

表 1　数字签名验证时间

Hash 函数	公钥尺寸/bit	验证时间/s
SHA-1	512	0.093
SHA-1	768	0.093
SHA-1	1 024	0.094
SHA-1	2 048	0.098

本文实现中可信测量的数据量包括 1 MB 的 BIOS 固件影像,2 个 64 KB 的 Oprom 和 1 个 512 B 的 MBR，总数据量约为 1 153 KB。由表 1 计算可知整个可信测量只需消耗 78 ms 左右的时间。本文在采用 Pentium 3.0 GHz 的实验中实际测得可信测量的累计时间为 496 ms。这包括了在 FLASH 影像中读取各模块映像数据的时间。不做可信测量的 BIOS 其启动时间一般在十几秒到几十秒之间(跟实际加载和测试的设备数量有关),实际测得关闭可信测量的启动过程为 29 s。可见可信测量对 BIOS 的启动速度的影响几乎可以忽略不计。

6　小结

固件 BIOS 系统虽小,但却是计算机系统可信链的起点,是可信测量的根核心。BIOS 的可信是计算机终端可信,网络连接可信的基础。本文通过分析可信 BIOS 的安全需求,提出一种可信 BIOS 体系结构，以 EFI BIOS 产品为基础实现了一种可信 BIOS—太行安全 BIOS 产品,解决了操作系统运行前阶段的可信链建立问题。下一步研究的重点,一是研究可信 BIOS 的信任管理问题,二是研究构建后续的可信链,构造可信计算终端。

参考文献:

[1] Heasman J.Implementing and detecting an ACPI BIOS rootkit[EB/OL].http://www.ngssoftware.com/jh_bhf2006.pdf.

[2] Arbaugh W A,Farber D J,Smith J M.A secure and reliable bootstrap architecture[C]//Procedings IEEE Symposium on Security and Privacy,4-7 May 1997:65-71.

[3] Wand Xin-cheng,Sun Hong,Cai Ji-ren,et al.TPM-chip based security startup system design[J].Application of Electronic Technique,2006,32(10):40-42.

[4] Fang Yan-xiang,Huang Tao.Design and implementation of trusted startup process for Linux [J].Computer Engineering,2006,32(9):51-53.

[5] Kozen D.Efficient code certification,98-1661[R].Computer Science Department,Cornell University,1998-01.

[6] Adelstein F,Stillerman M,Kozen D.Malicious code detection for open firmware[C]//Proceedings 18th Annual Conference on Computer Security Applications,9-13 Dec 2002:403-412.

[7] TCG.TCG Infrastructure architecture version 1.0.[S/OL].http://www.trustedcomputinggroup.org/specs/.

[8] Clark D D,Wilson D R.A comparison of commercial and military computer security policies[C]//Proceedings of the 1987 IEEE Symposium on Security and Privacy,1987.

[9] TCG.TCG PC specific implementation specification version 1.1[S/OL].2003-08-18.http://www.trustedcomputinggroup.org/groups/pc_client/.

[10] The Unified EFI Forum.Unified extensible firmware interface specification version 2.0[S/OL].2006-01-31.http://www.uefi.org.

[11] Sinosun.SSX35 TPM BIOS porting guide version 1.01 revision0,2006-03.

[12] Intel Corporation.Intel platform innovation framework for EFI architecture specification version 0.9[S/OL].2003-09-16.http://www.intel.com/technology/framework/.

[13] Menasce D A.Security performance[J].IEEE Internet Computing,2003,7(3):84-87.