基于独立核心安全组件的高安全体系结构

邵 婧[1,2] 殷红武[1,2] 陈左宁[2] 余 婷[2]
(解放军信息工程大学 郑州 450001)[1] (江南计算技术研究所 无锡 214083)[2]

摘 要 构建高安全体系结构是高安全级信息系统的一个重要前提。针对现有可信计算架构和基于VMM的虚拟化架构的核心模块存在易被篡改和被旁路的威胁,设计了一个基于独立核心安全组件的高安全体系结构HAICC。该体系结构通过硬件层有效实现了安全功能与计算功能的强隔离,将系统划分为独占不同物理资源的安全服务子系统和目标计算子系统,前者作为独立核心安全组件实施对整个计算系统的主动度量、实时监控、安全关键数据恢复。系统攻击实例及安全性分析表明,HAICC体系结构有效缓解了核心安全组件被篡改和被旁路的风险,提高了系统安全机制的完整有效性。

关键词 高安全,体系结构,防旁路,防篡改,强隔离
中图法分类号 TP309 **文献标识码** A

High-security Architecture on Independent Core Component

SHAO Jing[1,2] YIN Hong-wu[1,2] CHEN Zuo-ning[2] YU Ting[2]
(PLA Information Engineering University, Zhengzhou 450001, China)[1] (Jiangnan Institute of Computing Technology, Wuxi 214083, China)[2]

Abstract Building a high-security architecture is an important precondition of high-security information system. The core components of trusted computing architectures and virtualization architecture may be modified and bypassed. Aiming at this risk, a high-security architecture on independent core component(HAICC) was proposed. The architecture realizes strong isolation of security and computing functions by hardware. The system is divided into secure server subsystem and targeted computing sub-system, which occupy different physical resources. The former sub-system implements active measurement, runtime monitoring and key data recovery of the whole computing sub-system. The attack instance and security analysis show that, HAICC reduces the risk of modification and bypass for core security component, and enhances the integrity of security mechanisms.

Keywords High security, Architecture, Bypass prevention, Modification prevention, Strong isolation

1 引言

高安全级信息系统的研究一直以来都是信息系统安全领域的热点问题。作为安全系统的核心,TCB的要求是防篡改、防旁路和可分析,而TCB的设计与系统体系结构的设计是密不可分的。因此,构建高安全级系统的首要前提就是设计一个高安全的体系结构。

目前关于高安全体系结构的研究,主要包括基于TPM/TCM的可信计算架构和基于VMM的虚拟化架构。可信计算架构采用硬件物理保护来作为系统的安全基础,极大地弥补了应用层、核心层软件机制所存在的容易被卸载和被旁路的缺陷,如微软的NGSCB[1]、IMA[2]、PERSEUS[3]等。但是,由于TPM芯片是一个被动响应的弱功能设备,需要依赖于CPU不断对TPM模块进行访问,因此其仍然存在被旁路的可能。

虚拟化架构采用底层VMM来作为系统的安全基础,支持在同一硬件平台上运行多个相互隔离的虚拟机,更容易实现多级多域多策略的管理机制,可有效提高系统的安全性。典型的有IBM开发的Secure Hypervisor(sHyper)[4]、美国海军实验室开发的Xenon[5],以及XSM[6]、KVM[7]等。但是,由软件实现的VMM仍然存在安全漏洞[8],也存在被旁路和被篡改的威胁。

通过上述分析可知,为了保证系统安全机制的完整有效,高安全的体系结构需要具备的基本特征是,系统中的安全功能应该与计算功能隔离,避免安全部件受到其它功能部件的影响,同时应该采用硬件保护措施来对最核心的安全机制实施强隔离,使得实施核心安全机制的组件近似于一个主动控制、资源独立、密闭的物理设备。

对此,本文借鉴可信架构与虚拟化架构的思想,基于多核处理器,设计了一个基于独立核心安全组件的高安全体系结构(High-security Architecture on Independent Core Component, HAICC),其通过硬件层实施的强隔离,将系统划分为安全服务子系统和目标计算子系统,二者分别独占不同的物理资源。由安全服务子系统这个独立的核心安全组件来实施对整个计算系统的主动度量、实时监控、安全关键数据恢复,保

本文受国家高技术研究发展计划项目("863"项目)(2013AA01A210,2013AA013203),核高基项目(2013ZX01029002-001-001)资助。
邵 婧(1986—),女,博士生,主要研究方向为安全操作系统、云计算安全,E-mail:shaojingfox@163.com;殷红武(1973—),男,高级工程师,硕士生导师,主要研究方向为安全操作系统等;陈左宁(1957—),女,研究员,博士生导师,中国工程院院士,主要研究方向为信息安全、计算机体系结构等;余 婷(1983—),女,工程师,主要研究方向为云存储安全。

证了系统启动过程的安全可信，以及系统运行时的动态安全监控和可信恢复。系统攻击实例及安全性分析表明，HAICC结构可以有效防止核心安全组件被篡改和被旁路，为高安全信息系统设计提供有力保障。

2 HAICC体系结构组成

2.1 总体结构

本文设计的HAICC体系结构以多核处理器为底层硬件平台，由安全服务子系统(Secure Server Sub-system，SSS)和目标计算子系统(Targeted Computing Sub-system，TCS)两部分构成。如图1所示，两个子系统由硬件层来实施完全的物理隔离，其所占用的系统资源(如CPU核心、内存、缓存等)都是相互隔离的。其中，TCS只能通过专用单向数据通道给SSS发送数据，而SSS可以通过专用单向安全通道给TCS发送数据和指令。

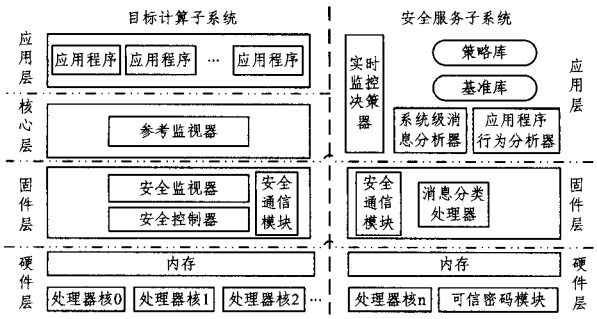

图1 HAICC体系结构

目标计算子系统负责与用户的正常交互，并实施常用的操作系统安全机制(如自主访问控制、强制访问控制、安全身份认证等)。安全服务子系统负责对目标计算子系统实施主动度量和实时监控，保证目标计算子系统的启动安全和运行时安全。该体系结构具有如下几个特征。

·安全组件的高安全性

SSS是一个相对独立封闭的组件，不存在与其它任何组件共享的物理资源。此外，SSS不与外部用户交互，除目标计算子系统发送的数据之外，不存在接收其它任何数据和指令的可能。这种硬件隔离及专用通路的方式在最大程度上保证了SSS的高安全性。因此，可以认为任何软件实施的攻击对安全服务子系统都是无效的(硬件攻击不属于本文研究的范畴)。

·主动度量与数据恢复

由于SSS被攻击的可能性极小，因此整个系统在启动时先启动SSS，然后再由SSS来保证TCS的启动安全。与传统的逐级度量可信链不同，TCS在启动过程中的每一级度量都以SSS为信任起点，并不依赖于上一级的可信性。此外，SSS对于所有被度量代码都进行数据备份，对于被破坏的部分可以进行数据恢复。

·实时秘密监控

SSS可以对TCS进行实时秘密监控。由于SSS未提供与系统外部的任何接口，且SSS对TCS的监控并不影响TCS的正常操作和运行，因此，上层用户无法感知SSS的存在，攻击者也因此很可能不会隐藏其攻击的痕迹，更有利于事后的追踪和处理。

·安全策略独立下发与修复

TCS使用的重要安全策略配置文件并不存放于本地，而是存放在SSS的策略库中。TCS在启动过程中，向SSS申请下发安全策略。同时，在TCS运行过程，SSS可以定期检查安全策略的完整性，保证安全策略在被破坏后可以得到及时修复。

2.1.1 目标计算子系统

目标计算子系统从底向上依次划分为物理层、安全固件层、核心层和应用层，如图1所示。其中，在核心层和物理层之间引入的安全固件层具有比核心层更高的特权。安全固件层中设置了安全监视器、安全控制器和通信模块，核心层设置了参考监控器。该子系统实施的主要安全功能如下。

(1)系统访问控制

主体对客体的访问控制，由核心层的参考监视器来实施，可以根据系统需求向SSS请求下发所采用的安全策略进行灵活配置。

(2)系统状态信息实时采集

安全监视器负责监视并采集系统的安全状态相关信息，将其通过唯一的数据通路发送给SSS，这些监控是由硬件体系结构来保证实现的。安全监视器运行在安全固件层，即使内核被完全“攻陷”了，也能正常工作。

根据监视信息的不同，可以将监视分成如下几类。

·异常监视，包括用户异常信息和系统异常信息。

·中断监视，包括设备中断、事件中断、核间中断等信息。

·系统调用监视，由于系统调用种类比较多，仅包括安全服务子系统的策略所关心的系统调用的状态信息。

·度量点监视，设计主动度量接口，采集用户状态信息。

(3)系统安全控制

安全控制器负责接收SSS发送的指令，对于安全、可疑和不安全的指令，对TCS分别采取相应的控制措施，即不处理、终止不安全进程、终止整个系统的运行。

2.1.2 安全服务子系统

安全服务子系统从顶向下依次划分为应用层、固件层和物理层。其中，固件层中设置了通信模块和消息分类器，应用层设置了系统级消息分析器、应用程序行为分析器、实时监控决策器、策略库和基准库。该子系统实施的主要安全功能如下。

(1)系统安全状态决策

SSS从TCS接收的数据依次经过消息分类器和消息分析器处理后，由实时监控决策器根据相关的监视策略进行决策，判断TCS目前处于何种安全状态，并给出相应的处理措施。

(2)系统关键资源完整性保护

对于系统启动过程涉及的代码，如BIOS、OS Loader、OS内核等，进行静态度量。对于系统运行时涉及的系统关键资源，如TCS的内核代码、安全策略配置文件、系统调用表、中断向量表、内核函数指针等，进行周期监控和度量。一旦发现其中某个资源被破坏，可以对其进行可信恢复。

(3)应用程序完整性保护

对应用程序的行为进行实时监控和动态度量，通过分析应用程序的行为特征和行为轨迹，判断应用程序是否受到恶意攻击和破坏。

(4)安全策略存储与下发

策略库中统一存放系统的安全策略，包括 TCS 使用的强制访问控制策略、网络安全配置策略等，以及 SSS 使用的度量策略和监控决策策略。系统启动过程中，SSS 根据 TCS 的请求下发相应策略。

(5)系统基准信息存储

基准库中存放需要被度量的代码和信息的基准值，包括操作系统关键资源的代码和应用程序行为特征等基准信息。

2.2 系统启动监控流程

整个系统加电后，安全服务子系统首先启动运行，然后再去引导和度量目标计算子系统的启动过程，依次度量 TCS 的 BIOS、OS Loader、基本内核、内核可加载模块 LKM，并将相关的安全策略从 SSS 加载至 TCS。每一级度量都以 SSS 为信任起点，且对度量未通过的代码进行可信恢复。

(1)开机引导阶段

Step1 系统加电，CPU 复位，TCM 初始化，依次启动相关安全监控固件及软件。

Step2 实时监控决策器从策略库读取度量和监控策略，并进行解析。

Step3 实时监控决策器从基准库获取 TCS 的度量基准值。

Step4 SSS 度量 TCS 的安全监视器、安全控制器，若度量未通过，则进行代码恢复。

Step5 SSS 度量 TCS 的 BIOS，若度量未通过，则进行代码恢复。

Step6 SSS 度量 TCS 的 boot loader，若度量未通过，则进行代码恢复。

Step7 SSS 度量 TCS 的基本内核，若度量未通过，则进行代码恢复。

(2)内核自启动阶段

Step1 SSS 度量 TCS 的 init 程序。

Step2 SSS 度量 TCS 自启动过程中依次执行的初始化脚本及相应配置文件的完整性，如/sbin/init、/etc/inittab 等。

Step3 SSS 度量 TCS 的可加载内核模块 LKM。

Step4 TCS 向 SSS 发送安全策略下发请求，例如，请求 SElinux 的策略、网络安全配置策略等。

Step5 SSS 根据 TCS 的请求，到策略库中查询相关策略，例如，SElinux 策略文件。

Step6 SSS 将查询的安全策略配置文件发送给 TCS。

Step7 TCS 接收安全策略配置文件，将其加载到相应的安全目录中。例如 SELinux 的安全策略通常存放在/etc/selinux 目录下。

2.3 系统运行监控流程

系统安全启动后，在运行过程中，安全服务子系统可以实时监控系统的安全状态并及时采取相应的处理措施。其监控的信息主要包括两方面，一方面是对系统本身安全性的监控，这主要包括内核映像等操作系统关键资源监控；另一方面是对系统中运行的应用程序安全性的监控，主要包括应用程序行为监控。

(1)系统级周期监控

Step1 安全监视器定时采集 TCS 的系统关键资源信息，包括内核代码、安全策略配置文件、系统调用表、中断向量表、内核函数指针等，然后通过消息通信模块发送给 SSS。

Step2 消息分类器根据系统信息标识将信息发送给系统级消息分析器。

Step3 系统级消息分析器对系统关键信息进行语义分析和相关处理，抽取出相关的特征值并发送给实时监控决策器。

Step4 实时监控决策器根据信息标识从基准库中获取相关基准信息，将其与接收的特征值进行比较验证。如果验证通过，则通过消息通信模块向 TCS 发送指令“安全”；否则，从基准库中获取被破坏资源的备份数据，通过消息通信模块向 TCS 发送指令“不安全”及相应的恢复数据。

Step5 安全控制器根据接收的决策指令对 TCS 进行相应的操作。若安全，则让 TCS 继续运行；若不安全，则对被破坏的资源进行可信恢复。

(2)应用程序级实时监控

Step1 安全监视器对应用程序的行为进行实时监视，收集应用程序的系统调用、用户异常等信息，加上相应标识后发送给 SSS。

Step2 消息分类器根据消息标识将信息传送至应用程序行为分析器。

Step3 应用程序行为分析器对收到的信息进行分析，构造出应用程序的行为特征序列和功能特征，发送给实时监控决策器。

Step4 实时监控决策器根据相应的监控策略，决断出应用程序处于何种安全状态，包括安全、可疑、不安全，并给出相应的决策指令。将指令发送给 TCS，并将状态信息记录到相关安全日志中。

Step5 安全控制器根据接收的决策指令，给系统发送相应的控制指令。若安全，则系统继续运行；若可疑，则系统中止可疑进程，重新加载应用程序；若不安全，则系统停止运行，启动恢复模式。

3 系统安全性分析

该体系结构中，TCB 主要包括整个安全服务子系统以及目标计算子系统中的安全监视器、安全控制器和参考监视器。其中，目标计算子系统只是提供一些辅助的措施，基础核心安全功能和机制都是由安全服务子系统来实施的。该体系结构的安全性体现在以下几方面。

(1)由硬件保证的强隔离，防止 TCB 被篡改。安全服务子系统对外部用户是不可见的，与系统外部无任何交互，唯一接收的信息就是目标计算子系统发送的数据。系统中所有安全相关信息的度量、分析、决策、控制都由安全服务子系统来实施，不会受到目标计算子系统中的各种应用程序、系统程序的影响。同时，目标计算子系统中的安全策略等关键系统资源是在安全服务子系统的监控之下。因此，系统的安全机制不会被篡改。

(2)实施独立主动控制。安全服务子系统拥有自己独立的硬件资源，不存在与目标计算子系统共享的任何系统资源，包括 TCM 模块、CPU 核心、内存、缓存等。它不是被动地等待被度量实体的 CPU 发送度量命令，而是主动发起度量和监控。此外，它可以根据决策结果，主动向目标计算子系统发送控制指令。

(下转第 371 页)

参 考 文 献

[1] Zhang Jun-cai, Zhao Jin-hui, Qian Xun. Risk assessment of mobile payment system security based on extension theory[C]// Proceedings of 2012 International Conference on Computer Science and Service System. 2012:880-883

[2] 戴宏. 移动支付系统安全风险评估[D]. 北京:北京交通大学, 2010

[3] 李峰. 移动支付安全研究[D]. 济南:山东大学, 2008

[4] Sun Wang-quan. The risk analysis and safety strategy on remote mobile payment[C]// Proceedings of 2012 IEEE Symposium on Digital Object Identifier. 2012:400-402

[5] 卫红春, 马丁. 基于改进的 3-D Secure 协议的移动支付安全解决方案[J]. 计算机应用与软件, 2011, 28(4):189-192

[6] 许峰, 崔隽, 黄皓. 基于 J2ME 的移动支付安全方案研究[J]. 计算机科学, 2008, 35(10):94-121

[7] 黄晓芳, 周亚建, 赖欣等. 基于第三方的安全移动支付方案[J]. 计算机工程, 2010, 36(18):158-162

[8] 张旋, 林逸风, 白川, 等. 基于贝叶斯网络的移动支付风险评估模型[J]. 计算机工程与应用, 2014, 50(5):60-64

[9] Saaty T L. The Analytic Hierarchy Process[M]. USA: McGraw Hill, 1980

(上接第 347 页)

(3)实施运行时监控,防止 TCB 被旁路。攻击者无法通过软件攻击的方式探测到安全服务子系统的存在,因此无法将其旁路。因此,攻击者虽然可以利用目标计算子系统的内核漏洞,旁路掉核心层的安全机制,但是无法旁路安全服务子系统实施的运行时监控。对运行系统的实时监控,可以及时检测出恶意程序的存在,并采取相应措施防止恶意行为的扩散。

(4)系统被破坏时可以进行可信恢复。安全服务子系统中将系统关键资源进行了备份储存,一旦检测出其中某个资源被破坏,可以及时将被破坏的资源进行替换,提高了系统的可用性和健壮性。

4 系统受攻击实例

在本节中,通过一类典型的攻击实例来分析说明在保证 TCB 的不可旁路和不可篡改方面,本文提出的体系结构比传统的体系结构有明显的优势。

传统的体系结构中,内核通常共享地址空间,任何一个模块都可以直接访问其他模块的功能,并修改其数据,甚至内核关键数据结构,因此可以轻易地旁路和篡改系统的安全机制。系统攻击一般通过篡改系统关键数据结构并插入恶意代码来改变系统正常的运行控制流,以获取更高的权限实施攻击。比较典型的如内核级 Rootkit,以劫持系统调用、拦截中断、模块注入等攻击手段,可以绕过传统的参考监视器来窃取重要数据。本文设计的体系结构中,由于安全服务子系统占用的是一块独立的物理内存,攻击者无法访问到这一块物理地址空间。下面以 Knark 等为样本,测试系统的防攻击能力。

图 2 给出了系统调用表的部分动态度量结果,从图中可知,独立监视器可以监测出 Knark 对系统调用表的修改状态。独立监视器也可以对中断向量表、内核内存映像等其余的关键数据结构进行动态度量,并能根据动态度量的结果与策略库中的策略进行及时处理,有效保障了 TCS 系统安全。

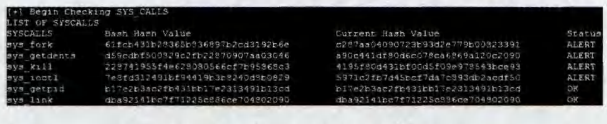

图 2 系统调用表度量结果

结束语 构建高安全级信息系统的首要前提,就是设计一个高安全的体系结构。针对目前基于 TPM/TCM 的可信计算架构和基于 VMM 的虚拟化架构存在被旁路和被篡改的威胁,本文提出了一个基于独立核心安全组件的高安全体系结构 HAICC。该体系结构通过硬件层实现了安全功能与计算功能的强隔离,将系统划分为独占不同物理资源的安全服务子系统和目标计算子系统。实施核心安全机制的安全服务子系统近似于一个主动控制、资源独立、密闭的物理设备,实施对整个计算系统的主动度量、实时监控、安全关键数据恢复。系统安全性分析及攻击实例表明,HAICC 体系结构有效缓解了核心安全组件被篡改和被旁路的风险,有效保障了系统安全机制的完整有效。

参 考 文 献

[1] Peinado M, Chen Y, England P, et al. NGSCB: A trusted open system[M]// Information Security and Privacy. Springer Berlin Heidelberg, 2004:86-97

[2] Sailer R, Zhang X, Jaeger T. Design and Implementation of a TCG-based Integrity Measurement Architecture[C]// Proceedings of 13th Usenix Security Symposium. San Diego, California, 2004:223-238

[3] Pfitzmann B, Riordan J, Stuble C, et al. The PERSEUS system architecture: RZ 3335[R]. 2001

[4] Sailer R, Valdez E, Jaeger T, et al. sHype: Secure hypervisor approach to trusted virtualized systems: RC23511[R]. 2005

[5] McDermott J, Freitas L. A formal security policy for xenon [C]// Proceedings of the 6th ACM workshop on Formal methods in security engineering. ACM, 2008:43-52

[6] Coker G. Xen security modules(xsm)[C]// Xen Summit. 2006: 1-33

[7] Kivity A, Kamay Y, Laor D, et al. Kvm: the Linux virtual machine monitor[C]// Proceedings of the Linux Symposium. 2007: 225-230

[8] 项国富, 金海, 邹德清, 等. 基于虚拟化的安全监控[J]. 软件学报, 2012, 23(8):2173-2187