VMI 技术研究综述

姜秋生，容晓峰
（西安工业大学 计算机科学与工程学院，陕西 西安 710021）

摘要： 虚拟机自省（Virtual Machine Introspection，VMI）技术充分利用虚拟机管理器的较高权限，可以实现在单独的虚拟机中部署安全工具对目标虚拟机进行监测，为进行各种安全研究工作提供了很好的解决途径，从而随着虚拟化技术的发展成为一种应用趋势。基于为更深入的理解和更好的应用 VMI 技术提供参考作用的目的，本文对 VMI 技术进行了分析研究。采用分析总结的方法，提出了 VMI 的概念，分析其实现原理和实现方式；详细地分析总结了 VMI 技术在不同领域的研究进展，通过对不同研究成果根据实现方式进行交叉分析比较，得出不同研究成果对应的 4 种实现方式；分析了 VMI 技术面临的语义鸿沟问题；最后对 VMI 技术研究进行总结和展望。

关键词： 虚拟机自省；虚拟化技术；虚拟机管理器；语义鸿沟
中图分类号： TP399 **文献标识码：** A **文章编号：** 1674-6236(2013)01-0013-04

A survey on VMI technology

JIANG Qiu-sheng, RONG Xiao-feng
(*School of Computer Science and Engineering, Xi'an Technological University, Xi'an* 710021, *China*)

Abstract: Because of VMI (Virtual Machine Introspection) technology make full use of the Virtual Machine manager of higher authority, it can monitor target Virtual Machine in a separate Virtual Machine which deploying security tools, and it provide a good solution to all kinds of security research work, so with the development of virtualization, it becomes a kind of application trend. In order to provides the reference of further understanding and better application of VMI technology. Using the method of analysis, this paper puts forward the concept of VMI, analyzes its principle and realization way; Detailed summarizes the VMI technology in different areas of research progress according to the implementation approaches for cross analysis and comparison, concludes the analysis results to correspond to the one of the four kinds of realization; Analysis the VMI technology which faces semantic gap problem; Finally, summaries and prospects the VMI technology research.

Key words: VMI; virtualization technology; virtual machine manager; semantic gap

虚拟化技术改变了系统软件与底层硬件紧耦合的方式，可以更加灵活地配置与管理计算机系统，而虚拟化平台中的虚拟机管理器位于操作系统和真实硬件平台之间，比操作系统的特权级更高，而且代码量更少。虚拟机监视器对上层虚拟机的完全控制权以及虚拟机间运行环境的强隔离性，为安全技术研究提供了良好的实施平台。基于虚拟化技术发展起来的虚拟机自省技术，充分利用虚拟化平台的优势。虚拟机自省利用隔离的多个安全的虚拟机，使用虚拟机管理器对虚拟机平台进行管理、资源分配和监视虚拟机的运行状态，各个虚拟机互补干扰，为安全领域的研究工作提供了技术支持。虚拟机自省技术监视虚拟机，获取被监控主机的内存信息，CPU 使用情况等计算机系统的运行状态信息，它被充分应用在安全领域的研究工作中。

文中对虚拟机自省技术进行分析研究，对虚拟机自省技术的概念、实现方式和工作原理进行分析阐述；对虚拟机自省技术的研究现状进行分析比较，根据实现方式对现有研究成果进行交叉比较分析，得出分析结果。分析虚拟机自省技术存在的语义鸿沟问题，最后对 VMI 技术研究工作进行总结和展望，为更好的应用 VMI 技术提供参考作用。

1 虚拟化技术

20 世纪 60 年代虚拟化技术开始正式发展，早期虚拟化技术只是应用在大型主机上，对相对昂贵的硬件资源进行充分利用，使得更多的用户可以更好地共享计算机资源。随着 IT 硬件的丰富多样化以及一些软件公司推出不同的虚拟化软件后，虚拟化技术的应用范围大幅度扩展。

虚拟化技术发展经历了完全虚拟化、半虚拟化和硬件虚拟化 3 个阶段[1]。完全虚拟化是指以虚拟机管理器为中心使得 PC 服务平台实现虚拟化。后来出现的半虚拟化技术，需要对客户机操作系统进行代码级的修改，这种修改非常繁琐，会带来系统指令级别冲突和运行效率的问题。随着虚拟化技术走到了硬件支持的阶段，使得半虚拟化障碍得到解决。硬件虚拟化技术就是把纯软件虚拟化技术的各项功能用硬件

收稿日期：2012-09-17 稿件编号：201209120
作者简介：姜秋生（1986—），女，山东临沂人，硕士。研究方向：网络与分布式技术。

电路逐一实现。

虚拟化平台的隔离性带来安全优势,已经被广泛使用以加强计算机系统的安全性,其基本思想是利用虚拟化技术将安全软件与被保护的软件系统隔离开,同时基于虚拟机监控器监控上层软件和操作系统的行为,使安全软件不必依赖于操作系统内核的安全性,同时虚拟机提供的强隔离性也保证了安全软件自身的安全。

2 VMI 概念

在虚拟机外部监控虚拟机内部运行状态的方法被称为虚拟机自省[2](Virtual Machine Introspection, VMI)。VMI 允许特权域查看非特权域的运行状态,并能获得被监控虚拟机运行状况相关的状态数据,这些数据包括内存使用情况,磁盘空间的使用情况,以及操作系统日志文件的数据等等。

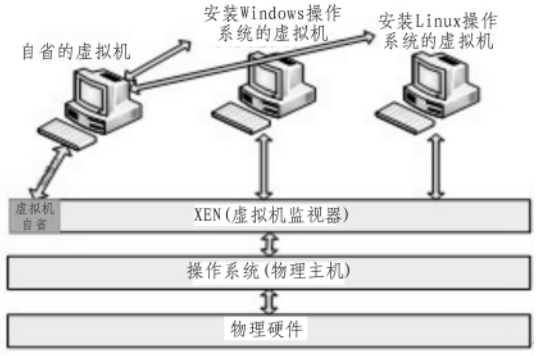

图 1 虚拟机自省在虚拟平台的位置

Fig. 1 Position of vmi in the virtualization platform

2003 年 NDSS 上发表的一篇论文 "A virtual machine introspection based architecture for intrusion detection",提出了 VMI 的概念,之后关于 VMI 的研究广泛开展起来。在这篇论文中这样定义 VMI:以分析虚拟机中运行的软件为目的,在虚拟机外部监测虚拟机的方法。

在这篇文章中主要利用 VMM 即虚拟机管理器实现 VMI 技术。VMM 作为运行在底层硬件上的软件,是一个轻量级的操作系统,提供了操作虚拟机的各种接口。将被监视操作系统(主机)运行在一个虚拟机,VMM 可以直接监测这个虚拟机的硬件状态,VMM 可以干预被监视主机的架构接口,通过监视硬件和软件级事件提供比普通的操作系统级机制更好的监控能力。VMM 对 CPU 和内存进行虚拟,它可以获得虚拟机的所有状态,例如 CPU 中的状态,内存,所有的 I/O 设备状态。

其实在 VMI 概念正式提出以前,VMI 的思想已经在计算机系统的开发应用中被采用。VMI 实质上是对计算机操作系统的运行状态的监视。早期我们在虚拟机中部署一个监视程序,对虚拟机以及其中运行的软件或程序进行监。随着虚拟化技术的发展,直接在虚拟机内部部署监视程序的方法不可靠,这时候将这个监视程序作驱动的形式实施在虚拟机的内核中,这比上述的方式更可靠一些。真正利用虚拟化平台优势进行 VMI 技术的实现是在虚拟机管理器中实施监视程序,最后 VMI 技术的实现就纯粹依赖于 VMM 了,这时 VMI 的概念才正式被提出。

3 VMI 的实现方式

VMI 的实现方式主要有 4 种,第一种是基于主机的 VMI,这种方式是在虚拟平台上的 VM 中置入监视代理如图 2(a)所示。这种方式的特点是需要在目标 VM 内核之上运行与普通应用相似的监视程序实施 VMI,即在虚拟平台上的 VM 中置入监视代理。由于直接运行在被监视虚拟机内部,所以它容易被攻击者攻击,易被旁路或被控制。但是它的最大的优势是语义精确,因为在本机实施所以效率高。

第二种是内核驱动方式实施 VMI 如图 2 (b) 所示,将 VMI 以驱动方式置于目标 VM 的内核中,比第一种方式更加可靠。但仍有可能被攻击者控制,因为依然是位于目标 VM 中,与普通操作系统类似,并未利用虚拟化平台的优势。

第三种方式是基于 trap、断点或 rollback 方式实施 VMI,类似于使用钩子函数,只不过捕获钩子是在 hypervisor 层,这样利用虚拟化平台的特点,由于虚拟化技术对 ring 级的利用,使运行于上层 VM 中的攻击程序无法对底层 VMM 实施控制,使攻击者攻击难度增加。如图 2(c)所示。这种 VMI 实现方式通过添加 Hooks 挂钩到 VMM 上,使在 VMM 中的自省库通过挂钩函数获得被监视虚拟机的状态。由于 VMM 这层管理权限的控制,安全性能得到很实际的提高。

最后一种方式如图 2(d)所示,不需要在 VMM 之外做监控代理,直接由 VMM 实施。该方式的好处很明显,安全性能是最高的。不足之处在于其监控能力:由于未在 VM 中放置任何代理,虽然 VMM 能够对 VM 进行完全控制,但控制粒度不好把握。

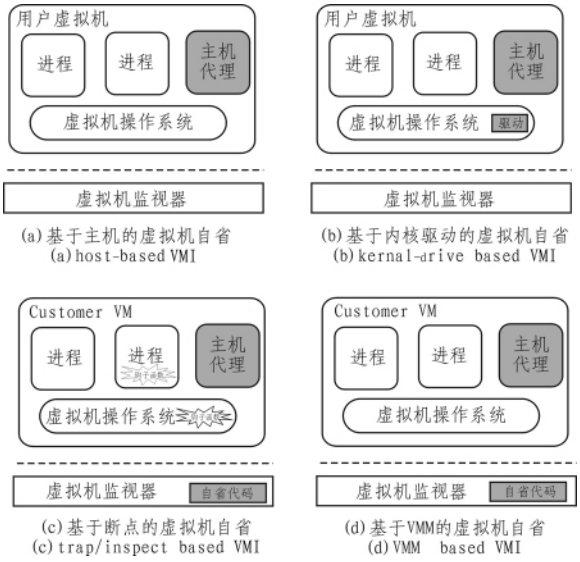

图 2 虚拟机自省技术的实现方式

Fig. 2 Realization of virtual machine introspection technology

4 VMI 技术研究现状

随着 VMI 技术研究的开展,研究人员在网络安全中的不

同领域研究 VMI 技术。

1)LiveWire 是基于 VMI 的入侵检测架构。综合了基于主机和基于网络两种入侵检测系统设计的优缺点，提出了利用 VMM(虚拟机管理器)的入侵检测架构。将入侵检测系统从被监视主机中分离出来，利用虚拟化平台优势提供了强大的隔离，使得耐攻击性能提高，但是对于被监视主机的运行情况仍然有一个比较好的了解。

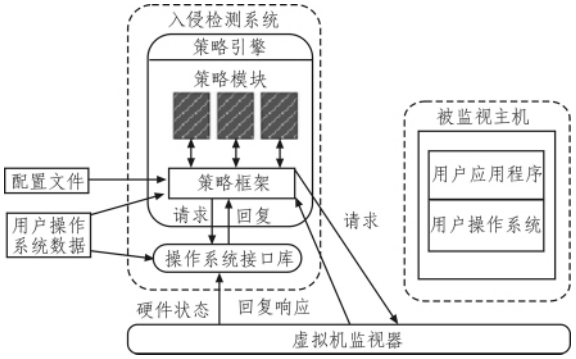

图 3　基于虚拟机自省的入侵检测系统结构

Fig. 3　VMI-Based IDS Architecture

基于 VMI 的入侵检测系统的结构：右侧是运行被监视主机的虚拟机，左边的是基于 VMI 的入侵检测系统的主要部分。操作系统接口库通过解释 VMM 输出的硬件状态对虚拟机提供操作系统级别的了解。策略引擎由一个普通的框架和一个策略模块组成，前者用于建立策略，后者实施具体的入侵检测策略。虚拟机监视器提供给分离的 IDS 和被监视虚拟机一个共同的底层，并且允许 IDS 检测虚拟机的状态。VMM 还允许 IDS 干预用户操作系统或者用户应用程序和虚拟硬件之间的相互作用。同样的，IntroVirt[3]也是利用虚拟化技术和 VMI 技术结合基于主机和基于网络的入侵检测系统的优点，研究的基于 VMI 的入侵检测系统。

2)VMWall[4]是使用虚拟机结合 VMI 技术实现的防火墙。以往的网络防火墙采取的是粗糙的策略，不能很好的观察被监视主机的状态，但是安全性比较高，可以防止被篡改。而基于主机的防火墙具有良好的细粒度策略，对被监视主机的状态拥有很好的观察，但是不能防止被篡改。VMWall 利用虚拟化平台的隔离性，将防火墙实施在安全的虚拟机中，利用 VMI 技术监视其他虚拟机，对被监视虚拟机的状态拥有很好的把握。

用户系统所在的 VM 中 IP/port 与通信进行之间的联系要用 VMI 来解决，它通过直接的内存检查，Xen 提供低级 API 允许 dom0 映射 domU 的任意作为共享内存的内存页。VMwall 使用 XenAccess[5]的 API 映射 dom0 内的 domU 内核的原始内存页。然后，它建立更高层次的内存抽象，如总结构和链接的数据类型，使用已知的客户机操作系统的内核编码语义从原始的内存页的内容获得信息。VMWall 利用高层次的抽象来确定执行在客户虚拟机中的应用程序如何使用网络资源。

3)VMWather[6]应用 VMI 技术进行防恶意软件系统的研究，将防恶意软件系统保护在一个安全的虚拟机中(in-the-ox)，这利用了虚拟化技术的隔离性；利用 VMI 技术在被监视虚拟机外部进行监视(out-the-box)。同时采用了用户视图投影技术解决了 VMI 技术中产生的语义鸿沟问题。

虚拟机管理器提供了虚拟机管理器级别的虚拟机状态的抽象，通过虚拟机自省获得的虚拟机状态有注册表、内存和磁盘。VMWatcher 充分利用 libxc 库，通过 xc_map_foreign_range()这个 API 映射其物理内存，来访问虚拟机的地址空间，然后通过映射的内存读取相应的内容。

4)在国内，基于虚拟化技术的恶意软件行为分析的研究中，提出了一种基于硬件辅助虚拟化技术的恶意软件行为分析系统—THVA[7]。THVA 充分使用虚拟机管理器对于虚拟机的系统控制权优势，实现虚拟机自省来监视虚拟机中的恶意软件行为。

5)国内对于虚拟机自省的另一方面是虚拟机监控器的研究。提出一种以轻量虚拟机监控器作为可信集的安全架构—Cherub[8]架构。Cherub 利用主流处理器的安全扩展指令和硬件辅助虚拟化技术实现虚拟机自省，在运行的操作系统中插入轻量级的虚拟机监控器，并利用该虚拟机监控器作为可信集用于实现多种安全目标。研究人员也对基于虚拟化的监控进行综述研究，虚拟机自省技术成为一种新的虚拟化监控实现方法。

通过比较分析我们可以看出现有的 VMI 技术研究中主要的实现方式是基于断点/trap 和基于 VMM 的实现方式。正是由于这两种方式利用了虚拟化技术的优势，从安全角度考虑实现 VMI 技术，所以应用广泛。

如表 1 所示，不同 VMI 系统对应的实现方式。

表1　VMI系统对应的实现方式

Tab. 1　Implementation Type of VMI Systems

实现方式	Livewire	VMWatcher	VMWall	THVA	Cherub
host-based					
kernal-drive based					√
trap/inspect based	√				
VMM based		√	√	√	

5　VMI 面临的问题——语义鸿沟

由于 VMI 技术实施在 VMM 基础上，VMM 获得的虚拟机运行状态数据是“文件”这种语义级别的，与直接应用操作系统提供的服务获得的“内存地址”这种语义之间存在一个映射解析的过程，这被称之为“语义鸿沟[9]”。语义重构是指由低级语义重构出高级语义(操作系统级语义)。操作系统接口库通过虚拟机管理管理器拦截的状态来恢复出操作系统级语义。

为了解决 VMI 应用程序和目标 VM 之间的“语义鸿沟”问题，研究人员进行了一些研究。VMWather 是解决语义鸿沟的研究，它采用用户视图投影(guest view casting)技术进行实现。它从 Raw 内存提取语义信息，用操作系统知识从语法上分析操作系统数据结构，从而可以重建语义。VMI 技术是在

无干扰情况下通过内部恶意进程推断 VMWatcher 行为，利用 VMM-level 观察获得 VM 状态，通过读取只读文件，复制扇区或者通过策略驱动以用于重构语义，赋予一个 VMM 水平的虚拟机上分配用户操作系统数据结构的语义定义和功能。

6 结束语

对于 VMI 技术的研究，国外非常积极，研究出了一些成果，在国内对于 VMI 技术的研究也开始开展起来，VMI 技术还是一个相对较新的研究领域和开发区域。不管在安全监控领域应用 VMI 技术进行研究，还是对 VMI 技术本身实现进行研究都有着重要的研究意义，VMI 技术对虚拟化技术和网络安全有着很重要的研究价值。

本文认为，VMI 技术还有以下一些方面值得进行深入研究。

1）增强虚拟机管理器的安全性

VMI 技术用于安全监控随着虚拟化技术得到广泛应用而备受关注，现有的研究工作利用虚拟机管理器实现 VMI 技术，实现依赖于虚拟机管理器。而虚拟机管理器是在真实硬件之上运行的软件，它作为整个虚拟计算平台的基础，一旦其出现某种安全问题，那么将会导致整个虚拟化平台上的其他虚拟机遭到非常严重的后果，危害性非常大。虚拟机管理器自身的安全问题就不容忽视[10]。

2）与现有的安全工具的结合

现有的研究工作中，研究人员已经开发出了大量的安全产品。然而 VMI 技术应用的虚拟化环境可以更好地监控虚拟机的内部运行状况，具有更好地隔离性。现有的安全产品如何有效的利用 VMI 技术进行改进，而不用重新开发，需要解决很多问题。一方面，语义差别。传统的系统环境和虚拟化平台中虚拟机管理器获得的信息存在着语义差别[10,11]。另一方面，VMI 技术在现有的安全产品中的实现。这些都要进行综合衡量。

参考文献：

[1] 金海. 计算机系统虚拟化-原理与应用[M]. 北京：清华大学出版社，2008.

[2] Tal G，Mendel R. A virtual machine introspection based architecture for intrusion detection[J]. Network and distributed system security，2003:191-206.

[3] Fabrizio B，Daniele S. Building trust worthy intrusion detection through VMIntrospection[J]. International Accounting Standards，2007，21(6):209-214.

[4] Abhinav S，Jonathon G. Tamper-Resistant. Application-aware blockin of malicious network connections[J]. Recent advances in intrusion detection，2008:39-58.

[5] XenAccess Project [EB/OL].(2007-05-20) [2011-11-16]. http://xenaccess.sourceforge.net/.

[6] Jonas P，Christian S，Claudia E. A formal model for virtual machine introspection [J]. Proceedings of the 2nd ACM workshop on Virtual Machine Security，2009:1145-1154.

[7] 唐源，李建平，白雪，等. 虚拟机监视器结构与实现技术[J]. 计算机应用研究，2009(5):1632-1635.
TANG Yuan，LI Jian-ping，BAI Xue，et al. Architecture and implementation of virtual machine monitor [J]. Computer application research，2009(5):1632-1635.

[8] 李博，李建欣，胡春明，等. 基于VMM层系统调用分析的软件完整性验证[J]. 计算机研究与发展，2011(8):1438-1446.
LI Bo，LI Jian-xin，HU Chun-ming，et al. Software integrity verification based on the VMM layer system call analysis[J]. Journal of computer research and development，2011(8):1438-1446.

[9] Nance K，Hay B. Virtual machine introspection Observation or Interference[J]. IEEE Computer Society，2008(7):1540-1545.

[10] 王丽娜，高汉军，刘炜，等. 利用虚拟机管理器检测及管理隐藏进程[J]. 计算机研究与发展，2011(8):1534-1541.
WANG Li-na，GAO Han-jun，LIU Wei，et al. Detect and manage hidden process with virtual machine manager [J]. Journal of computer research and development，2011(8):1534-1541.

[11] 项国富，金海，邹德清，等.基于虚拟化的安全监控[EB/OL](2012-04-24) [2012-06-20]. http://www.cnki.net/kcms/detail/11.2560.TP.20120424.1526.003.html

（上接第 12 页）

[4] 戚艳军，刑继军. Ajax与Struts结合构建Web应用研究[J]. 西安：现代电子技术，2008，32(22)：57-58，66.
QI Yan-jun，XING Ji-jun. Web application using Ajax and STRUTS [J]. Modern Electronic Technique，2008，32(22)：57-58，66.

[5] 约翰逊，魏海萍. J2EE设计开发编程指南[M]. 北京：电子工业出版，2001.

[6] HUANG Hai，ZHU Yue-long. Research on application of J2EE design patterns and framework technology[J]. Computer and Modernization，2006，22(5)：114-116.