(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101589365 B
(45)授权公告日 2012.07.04

(21)申请号 200780050510.8

(22)申请日 2007.12.19

(30)优先权数据
11/627,314 2007.01.25 US

(85)PCT申请进入国家阶段日
2009.07.27

(86)PCT申请的申请数据
PCT/US2007/088219 2007.12.19

(87)PCT申请的公布数据
WO2008/091462 EN 2008.07.31

(73)专利权人 微软公司
地址 美国华盛顿州

(72)发明人 B·贝克 S·A·费尔德 E·托奥特 S·辛哈 J·甘吉利 F·福尔茨 D·柯特勒

(74)专利代理机构 上海专利商标事务所有限公司 31100
代理人 蔡悦 钱静芳

(51)Int.Cl.
G06F 9/06(2006.01)
G06F 11/30(2006.01)

(56)对比文件
CN 1511286 A,2004.07.07,
EP 1271313 A2,2003.01.02,
WO 2006/063274 A1,2006.06.15,
US 5469556 A,1995.11.21,
US 2003/0120856 A1,2003.06.26,
US 2004/0123288 A1,2004.06.24,

审查员 刘佳

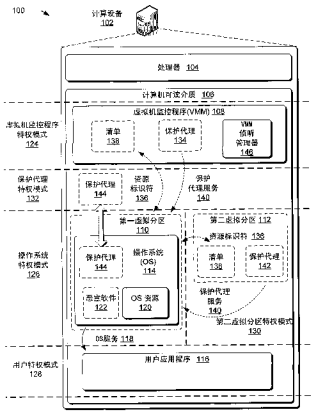
权利要求书 2 页 说明书 13 页 附图 9 页

(54)发明名称

用于保护操作系统资源的方法

(57)摘要

本文描述了能够使得保护代理能从不可从操作系统特权模式访问的存储器中确定操作系统的一个或多个资源是否已被修改的工具。在某些实例中,这些工具可使得保护代理能够驻留在虚拟机监控程序中。在其他实例中,这些工具可使得保护代理能够驻留在由虚拟机监控程序提供的不同的虚拟分区中。通过在操作系统特权模式之外操作,保护代理可较不易受到在操作系统特权模式中操作的实体的攻击。



CN 101589365 B

1. 一种保护操作系统资源的方法，所述方法包括：

通过虚拟机监控器提供至少一个第一分区和一个第二分区；

在不可由在操作系统特权模式中操作的实体访问的存储器中接收标识运行在虚拟机监控器提供的第一分区中的操作系统的一组一个或多个资源的强制实施策略；

利用所述强制实施策略从不可由在操作系统特权模式 (126) 中操作的实体访问的存储器中标识 (704) 所述运行在虚拟机监控器提供的第一分区中的操作系统 (114) 的那组一个或多个资源 (120)；

从所述存储器中确定 (706) 所述那组一个或多个资源 (120) 中的一个或多个是否已被更改；以及

响应于确定所述那组一个或多个资源 (120) 中的一个或多个已被更改：(i) 终止所述操作系统或者 (ii) 在重新引导时将非法操作通知给所述操作系统，

其中接收所述强制实施策略、标识一组一个或多个资源、以及确定所述那组一个或多个资源中的一个或多个是否已被更改发生在 (i) 虚拟机监控器下或者 (ii) 由所述虚拟机提供的第二分区内。

2. 如权利要求 1 所述的方法，其特征在于，还包括向所述操作系统 (114) 展示应用程序编程接口 (API) 并经由所述 API 来接收对所述那组一个或多个资源 (120) 中的一个或多个的标识。

3. 如权利要求 1 所述的方法，其特征在于，所述那组一个或多个资源 (120) 包括系统服务分派表 (SSDT)、中断分派表 (IDT) 或全局描述符表 (GDT)。

4. 一种保护操作系统资源的方法，包括：

更改 (802) 虚拟机监控程序的侦听管理器 (146) 以便有效地允许接收与操作系统资源 (120) 相关联的存储器页或寄存器已被更改的指示，所述操作系统资源位于由所述虚拟机监控程序提供的分区内；

在所述虚拟机监控程序处接收 (804) 标识所述操作系统资源 (120) 以及一个或多个其他操作系统资源 (120) 的强制实施策略；

由所述虚拟机监控程序的侦听管理器接收 (806) 所述与操作系统资源 (120) 相关联的存储器页或寄存器已被更改的指示；以及

响应于接收到所述指示来关闭 (808) 操作系统特权模式 (126) 以便有效地关闭与所述操作系统资源 (120) 相关联的操作系统 (114)。

5. 如权利要求 4 所述的方法，其特征在于，所述操作系统资源 (120) 是在所述存储器页处的中断分派表 (IDT) 并且所述与操作系统资源 (120) 相关联的寄存器是 IDT 寄存器。

6. 如权利要求 4 所述的方法，其特征在于，所述操作系统资源 (120) 是系统服务分派表 (SSDT) 或全局描述符表 (GDT)。

7. 如权利要求 4 所述的方法，其特征在于，所述操作系统特权模式 (126) 的关闭 (808) 由虚拟机监控程序 (108) 执行。

8. 如权利要求 4 所述的方法，其特征在于，所述强制实施策略还描述与所标识的操作系统资源 (120) 中的每一个相关联的保护属性，所述保护属性中的至少一个将对应资源 (120) 描述为只读。

9. 如权利要求 8 所述的方法，其特征在于，还包括针对与将所述资源描述为只读的保

护属性相关联的资源 (120) 强制实施不变性。

10. 一种保护系统资源的方法，所述方法包括：

将计算设备虚拟化为至少第一和第二虚拟机分区，其中与操作系统特权模式相关联的操作系统驻留在第一分区，而一个或多个计算机可读介质驻留在第二分区内；

标识被设计成在操作系统特权模式中操作的一个或多个操作系统资源并确定所述一个或多个操作系统资源中的一个或多个是否已被更改，其中所述一个或多个计算机可读介质不易受到来自于所述操作系统特权模式中的攻击；以及

响应于确定所述一个或多个操作系统资源中的一个或多个已被更改，在第一虚拟机分区下关闭与所述操作系统特权模式相关联的操作系统。

用于保护操作系统资源的方法

[0001] 背景

[0002] 计算设备中的处理器通常包括特权和非特权模式。以特权模式运行的软件一般能够执行处理器所支持的每一条指令。通常，操作系统内核在特权模式中运行，该模式有时被称为“环 0”、“管理员模式”或“内核模式”。

[0003] 相反，可约束在计算设备上运行的某些软件仅以非特权模式运行。该模式一般允许软件执行处理器的指令的子集。操作系统由此可使用该非特权模式来限制以该模式运行的软件的活动。例如，软件可被限于计算设备的存储器的特定子集。该非特权模式有时被称为“环 3”或“用户模式”。一般而言，计算设备用户应用程序以该非特权模式操作。

[0004] 如果软件应用程序以该非特权模式操作，则该应用程序可请求访问无法直接从该非特权模式访问的存储器部分。该应用程序可能希望例如在该存储器部分中执行诸如“创建新文件”等操作。该请求通常经由将该非特权模式代码转换成特权模式代码的调用门或其他系统调用指令来路由。该转换确保非特权模式无法直接访问被指定为只可从特权模式访问的存储器。

[0005] 根据这些模式，恶意代码的制作者可访问特权模式并安装改变计算设备的行为的恶意软件。该恶意软件可例如更改文件位置、隐藏文件、修改文件、改变键击等。这些恶意软件中的某一些可包括“rootkit”（根套件），其不仅改变计算设备的行为而且将其自身隐藏在特权模式的存储器中。在计算设备上运行的反病毒应用程序因此可能无法发现该隐藏的 rootkit，由此允许该恶意软件继续其恶意行动。此外，这些恶意软件可如以下所讨论地遮蔽 (patch over) 操作系统的内置保护系统。

[0006] 恶意软件制作者可访问特权模式并且以各种方式将恶意软件加载到计算设备上，包括通过欺骗计算设备用户不知不觉地将恶意软件安装到该用户自己的计算设备上。结果，当前操作系统通常采用一个或多个保护系统来检测这些恶意软件。这些保护系统通常监视某些重要的操作系统资源以检测对这些资源的任何改变。如果这一保护系统检测到这一改变，则该保护系统可判定该特定 资源已被恶意软件感染。这些保护系统还可向用户的反病毒应用程序提供当前驻留在非特权模式的存储器中的应用程序的列表。当然，如果恶意软件成功隐藏，则它不会出现在所提供的列表上。此外，如果恶意软件成功遮蔽保护系统，则该保护系统可能无法运行或无法以其他方式检测对重要的操作系统资源的任何改变。

[0007] 虽然这些保护系统可以是有效的，但它们也可具有几个弱点。第一，这些系统通常依赖于隐匿并由此容易在被恶意软件标识的情况下受到恶意利用。即，如果恶意软件解密了保护系统的身份并定位了该保护系统，则该恶意软件可禁用该保护系统本身。恶意软件制作者还可指示其他人如何做同样的事情。此外并且与第一个弱点有关，这些保护系统一般在与操作系统相同的保护范围内操作（例如，在特权模式自身中）。因此，如果恶意软件获得对特权模式的访问权并且能够去除隐匿保护系统的屏蔽，则保护系统自身容易受到攻击。最后，这些保护系统与操作系统或特权模式同时初始化。因此，如果恶意软件或恶意软件制作者在该初始化之前获得对计算设备的控制，则它或他可防止保护系统初始化。

[0008] 概述

[0009] 本文描述了能够使得保护代理能从不可从操作系统特权模式访问的存储器中确定操作系统的一个或多个资源是否已被修改的工具。在某些实施例中，这些工具可使得保护代理能够驻留在虚拟机监控程序中。在其他实施例中，这些工具可使得保护代理能够驻留在由虚拟机监控程序提供的不同的虚拟分区中。通过在操作系统特权模式之外操作，保护代理可较不易受到在操作系统特权模式中操作的实体的攻击。

[0010] 提供本概述是为了以简化的形式介绍将在以下详细描述中进一步描述的一些概念。本概述不旨在标识所要求保护的主题的关键或必要特征，也不旨在用于帮助确定所要求保护的主题的范围。例如，术语“工具”可以指上述上下文和通篇文档所准许的系统、方法、计算机可读指令、和 / 或技术。

[0011] 附图简述

[0012] 图 1 示出了工具的各个实施例可在其中操作的示例性操作环境。

[0013] 图 2 展示了图 1 所示模块的不同的计算设备存储器权限。

[0014] 图 3 表示图 1 所示模块中的某一些驻留在其中的计算设备存储器的不同部分。

[0015] 图 4 是示出虚拟机监控程序可保护与保护代理相关联的存储器部分并设置定时器以运行该代理的示例性方式的流程图。

[0016] 图 5 示出了具有能够将物理处理器虚拟化为多个操作系统虚拟处理器和保护代理虚拟处理器的虚拟机监控程序的示例性体系结构。

[0017] 图 6 示出了图 5 的物理处理器在各虚拟处理器中可分配到多少带宽。

[0018] 图 7 是示出工具可启用并运行驻留在不可从操作系统特权模式访问的存储单元的保护代理的某些方式的示例性过程。

[0019] 图 8 是示出工具可更改虚拟机监控程序以启用并运行驻留在不可从操作系统特权模式访问的存储单元的保护代理的某些方式的示例性过程。

[0020] 图 9 是示出工具可通过向虚拟机监控程序作出请求来创建保护代理特权模式的某些方式的示例性过程。

[0021] 图 10 是示出工具可通过将真实计算机处理器虚拟化为其中至少一个用于运行保护代理的多个虚拟计算机处理器来创建保护代理特权模式的某些方式的示例性过程。

[0022] 图 11 是示出工具可使得能够添加不存在于底层物理处理器上的特权模式的某些方式的示例性过程。

[0023] 贯穿本公开和各附图，使用相同的标号来引用相同的组件和特征。

[0024] 详细描述

[0025] 概览

[0026] 下文描述了能够以使得保护代理不可从操作系统特权模式更改或访问的方式操作该保护代理的工具。这些工具因此使得能够对保护代理本身进行保护，由此确保保护代理检测对重要的操作系统资源的更改的能力。另外，这些工具可响应于检测到资源更改或响应于所尝试的对保护代理本身的修改来关闭操作系统或操作系统特权模式。此外，这些工具可使得保护代理能够对操作 系统资源强制实施不变性，而无需在此之后检测资源修改。

[0027] 以下在标题为“示例性操作环境”的章节中阐明这些工具可在其中启用这些和其

它动作的环境。之后是标题为“自主保护代理”的章节并且该章节包括两小节。标题为“虚拟机监控程序保护代理”的第一小节描述了保护代理可驻留在虚拟机监控程序中并在其中执行的一种示例性方式。这之后是标题为“虚拟分区保护代理”的另一小节，该小节描述了保护代理可居留在与操作系统的分区分开的虚拟分区中并在其中执行的一种示例性方式。

[0028] 之后是标题为“自主保护代理特权模式”的另一章节并且该章节也包括两小节。第一小节描述了虚拟机监控程序定时器可将保护代理特权模式添加到底层处理器的一种示例性方式，并且标题为“向虚拟机监控程序作出的保护请求”。之后是标题为“保护代理虚拟处理器”的小节并且该小节描述了可创建保护代理特权模式的另一种方式，在这该实例中使用包括被配置成以保护代理特权模式运行保护代理的一个虚拟处理器在内的多个虚拟处理器。之后是标题为“工具的示例性使用”的章节并且该章节描述了在操作中的先前描述的工具的示例。最后，标题为“工具的其他实施例”的章节描述了其中这些工具可起作用的各种其他实施例和方式。包括这些章节标题和概述的本概览是出于方便读者的目的而提供的，而非旨在限制权利要求或所命名的各节的范围。

[0029] 示例性操作环境

[0030] 在详细描述这些工具之前，提供示例性操作环境的以下讨论来帮助读者理解可采用这些工具的各发明性方面的某些方式。以下描述的环境仅构成一个示例且并非旨在将这些工具的应用限于任一个特定操作环境。可使用其它环境而不背离所要求保护的主题的精神和范围。例如，虽然以下各章节以单个保护代理描述了各实施例，但也可利用多个保护代理。在某些实例中，这些保护代理可独立或并排运行。在这些实例中，保护代理通常只能够访问其相应分区内的存储器。此外，可同时利用以下所描述的各种技术。即，不同的保护代理可在相同的操作环境中利用不同的技术。

[0031] 转向当前示例，图 1 在 100 处概括地示出了一个这样的示例性操作环境。该环境包括计算设备 102，其本身包括一个或多个处理器 104 以及计算机可读介质 106。计算机可读介质 106 包括虚拟机监控程序 108（例如，管理程序），其可使得该一个或多个处理器能够虚拟化为多个虚拟处理器。虚拟机监控程序 108 还可启用多个虚拟分区。可使一个或多个虚拟处理器与每一个分区相关联，并在可用的物理处理器上调度这些虚拟处理器。如图所示，在某些实施例中，虚拟机监控程序可启用第一虚拟分区 110 和第二虚拟分区 112。如以下所详细讨论的，这些分区可用于将操作系统功能与保护代理服务分开。

[0032] 同样如图所示，计算机可读介质 106 还包括操作系统 (OS) 114 以及一个或多个用户应用程序 116。操作系统 114 向用户应用程序 116 提供操作系统服务 118，由此允许这些应用程序在计算设备上运行。另外，一个或多个操作系统资源 120 驻留在操作系统上。示例性资源包括系统服务分派表 (SSDT)、中断分派表 (IDT)、全局描述符表 (GDT) 等。同样如图所示，操作系统可包括恶意软件 122（即，具有恶意企图的代码），其可能已经以上述方式或以其他方式被加载到计算设备上。以下所讨论的一个或多个保护代理可检测恶意软件对操作系统资源作出的改变，并且响应于该检测来采取防御动作。如果代理作出这一判定，则该保护代理可关闭操作系统和 / 或计算设备或可采取其他抵抗动作。

[0033] 在讨论了计算设备的结构之后，注意力现在转向存在于底层的一个或多个物理处理器 104 上的不同的特权模式。虚拟机监控程序特权模式 124 表示图 1 所示的最具特权模式。该特权模式可访问所有或基本上所有的设备资源和存储器。从虚拟机监控程序特权模

式124，虚拟机监控程序可调度处理器并允许访问针对每一个虚拟分区的存储器区域。虽然在分区内运行的操作系统可能相信它控制着物理处理器的所有资源，但实际上它只控制如由虚拟机监控程序来确定的部分。

[0034] 具有比虚拟机监控程序特权模式少的特权的操作系统特权模式126可访问所有操作系统资源120以及大多数或所有操作系统存储器。然而，该特权模式不可访问与诸如第二虚拟分区112等另一分区相关联的任何资源或存储器。然而，因为该特权模式一般可访问所有操作系统存储器，所以它有时被称为“具有特权的模式(Privileged Mode)”。“环0”、“管理员模式”或“内核模式”也可描述该特权模式。如上所述，在操作系统特权模式126中运行的用户应用程序一般能够执行处理器所提供的除为虚拟机监控程序模式保留的指令之外的大多数指令。

[0035] 该操作系统特权模式与有时被称为“非特权模式”、“环3”或简称为“用户模式”的用户特权模式128形成对比。同样如上所述，用户应用程序在从用户特权模式128操作时可能无法访问或更改与操作系统相关联的特定存储器。一般而言，计算设备用户应用程序在执行基本操作时以该用户特权模式操作。

[0036] 除了上述模式之外，图1还示出了第二虚拟分区特权模式130和保护代理特权模式132。如将在以下详细讨论的，虽然一般不具有与虚拟机监控程序特权模式一样多的存储器访问权，但保护代理特权模式132可访问操作系统特权模式不可访问的存储器部分。由此，该特权模式可比操作系统特权模式更具特权，但比虚拟机监控程序特权模式所具有的特权少。

[0037] 同样如将在以下详细描述的，第二虚拟分区特权模式一般可访问与第二虚拟分区112相关联的存储器。另外，该模式可访问第一虚拟分区。这一附加访问可例如允许驻留在第二虚拟分区中的保护代理扫描与第一虚拟分区及其对应操作系统相关联的存储器。该模式一般不可访问虚拟机监控程序，并且因此具有比虚拟机监控程序特权模式少的特权。然而，第二虚拟分区特权模式仍然可访问操作系统特权模式不可访问的存储器部分。

[0038] 同时，图2示出了计算设备存储器权限200。该附图由此表示可由图1的 模块访问的存储器数量。如图所示，以虚拟机监控程序特权模式124操作的虚拟机监控程序108在所示的所有模块中具有最多的存储器权限。事实上，该虚拟机监控程序驻留在存储器部分202中并且只有它可访问存储器部分202。接着，保护代理204(例如，图1所示保护代理中的任一个)以保护代理特权模式132操作并且可访问除了对应于虚拟机监控程序的部分202之外的所有存储器。然而，该保护代理可访问该保护代理本身驻留在其中的存储器部分206。

[0039] 同时，操作系统114以操作系统特权模式126操作并且可访问除部分202和部分206之外的所有存储器。虽然操作系统可能无法访问与保护代理相关联的存储器部分206，但该操作系统及其相关联的特权模式可访问存储器部分208。该存储器部分208有时被称为内核存储器或操作系统的最低层组件，并且一般包含图1所示的资源。然而，即使恶意软件在存储器部分208中加载并操作，该恶意软件也无法访问与保护代理相关联的存储器部分206。

[0040] 最后，图2示出用户应用程序116只可访问存储器部分210。这些用户应用程序和对应的用户特权模式不可访问与操作系统的最低层组件相关联的存储器部分208。使用该

操作环境的概念，以下四个章节详细描述了可使得保护代理不可从操作系统特权模式更改或访问的各示例性方式。

[0041] 自主保护代理

[0042] 以下章节描述了能够从不可由在操作系统特权模式中操作的实体访问的存储器中确定一个或多个操作系统资源是否已被修改的工具。由此，这些工具可允许保护代理驻留在除了操作系统存储器本身的存储单元之外的存储单元。更具体而言，以下各小节描述了保护代理可如何驻留在虚拟机监控程序或自主虚拟分区中。

[0043] 虚拟机监控程序保护代理

[0044] 该小节描述了保护代理 134 可如何如图 1 所示地驻留在虚拟机监控程序本身中。因为操作系统特权模式无法访问虚拟机监控程序，所以该存储单元保护了保护代理免遭位于操作系统存储器中的恶意软件的攻击。为了从该存储单元进行操作，保护代理接收该保护代理 134 可监视的一个或多个操作系统资源 120 的标识。该标识可经由资源标识符 136 来接收。如图所示，操作系统可通过应用程序编程接口 (API) 调用来将该信息提供给虚拟机监控程序，或者该操作系统可以按清单 138 的形式来提供该信息。如上所述，这些资源可包括 SSDT、IDT 和 GDT。

[0045] 一旦保护代理接收到了资源标识，该保护代理 134 就将保护代理服务 140 扩展至操作系统 114。这些保护代理服务一般包括 :确定所标识资源中的任一个是否已被更改。如果作出这一判定，则保护代理或虚拟机监控程序可例如关闭操作系统。保护代理服务还可包括针对被标记为不可更改（例如，“只读”）的任何资源强制实施不变性。

[0046] 采用这一体系结构开始于加载并初始化能够主存一个或多个操作系统的虚拟机监控程序。在该示例中，虚拟机监控程序主存单个操作系统 114，该操作系统本身在虚拟机监控程序加载之后开始初始化。在操作系统初始化期间，与该操作系统的最低层组件相关联的存储器部分 208（例如，内核）首先加载。操作系统资源 120（例如，SSDT、GDT、IDT）中的部分或全部一般居留在该存储器部分 208 中。

[0047] 在操作系统初始化之前或同时，保护代理 134 可开始从虚拟机监控程序中运行。如上所述，保护代理一般接收一组一个或多个操作系统资源的标识并确定所标识的资源中的一个或多个是否已被更改。注意，每一个所标识资源通常都包括处于多个存储单元的多个组件，保护代理可监视这些组件中的每一个以完全保护整个资源。例如，如果清单将 SSDT 标识为将要监视和保护的资源，则保护代理不仅保护实际表而且还保护 SSDT 的其他组件。例如，保护代理还可监视并扫描指向该表的存储单元的寄存器。此外，保护代理还可监视将 SSDT 的虚拟地址转换成物理地址的存储器转换数据结构（例如，页表）。如果保护代理无法这样做，则恶意代码可创建具有不同页表映射的另一张表（即，绕过 SSDT 本身）。

[0048] 除了标识之外，保护代理还可接收指示该保护代理如何保护相应资源的保护属性。例如，保护代理可接收 SSDT 资源的标识以及相应的保护属性“只读”。该保护代理因此获悉 SSDT 应保持只读并由此不应被更改。“初始化只读”是另一可能的保护属性，其指示保护代理相应资源可在初始化期间写入一次，但 在此之后该资源应保持只读。

[0049] 保护代理既可主动也可被动地按多种方式接收资源的标识和资源保护属性。例如，操作系统可提供标识保护代理可监视的资源的经数字签名的清单。该经数字签名的清单可以按多种方式标识资源，诸如通过名称（例如，SSDT、IDT、GDT 等）或通过将资源映射

到存储器部分 208 中的对应存储单元的地址等。在后面的实例中，该清单可标识资源的来宾物理地址、来宾虚拟地址或系统物理地址。注意，在某些情况下，可将来宾物理地址映射到实际系统物理地址以便发现对应资源组件的实际物理地址。

[0050] 在虚拟机监控程序或保护代理接收到清单之后，这些组件可确定该清单是否已被篡改或修改。如果虚拟机监控程序或保护代理作出这一判定，则该虚拟机监控程序或保护代理可选择使启动操作系统失败。另外，可使与资源列表相关联的加密无效，由此保护其安全性。

[0051] 作为对清单的补充或替换，保护代理可经由对虚拟机监控程序的一个或多个应用程序编程接口 (API) 调用 (例如，“超调用 (hypercall)”) 来接收资源和保护属性标识。在操作系统初始化时，该操作系统 (以及或许操作系统的最低层组件 208) 可作出对虚拟机监控程序的超调用，从而将可监视和保护的特定资源通知给保护代理。这些超调用可以按与上述相同的方式来标识相关资源。同样如上所述，这些超调用还可标识资源的保护属性。

[0052] 在利用经数字签名的清单以及一个或多个超调用的实施例中，保护代理可在操作系统引导之前或同时首先扫描清单中所标识的资源。在该初始扫描之后，操作系统然后可作出对虚拟机监控程序的超调用以指示保护代理确定该超调用所标识的页面是否已被更改。清单由此在每一次操作系统引导时标识要扫描的资源，而超调用在其相应初始化时标识要动态扫描的资源。

[0053] 在标识了要监视的资源后，保护代理然后确定资源 (例如，上述 SSDT 的所有部分) 是否已被更改。保护代理还可对所标识的资源强制实施不变性。例如，保护代理可确保被指定为“只读”的任何资源都不变为“可写”。

[0054] 为了以此方式监视并保护资源，在虚拟机监控程序中执行的代码可采用虚拟机监控程序侦听管理器 (例如，图 1 的管理器 146)。如果这样指示，则该侦听管理器可在所标识资源的各个组件上注册侦听。由于该注册，虚拟机监控 程序中的保护代理现在可在作出访问或修改这些所标识资源的尝试的情况下接收侦听。由此，保护代理可校验和扫描所标识资源的各个组件。它还可主动阻止修改这些资源的尝试。

[0055] 在某些实施例中，保护代理扫描资源并确定资源的初始状态以供在比较将来扫描的结果时使用。在其他实施例中，保护代理已经知道资源的初始状态以比较将来扫描的结果。在任何情况下，保护代理都可计算该初始状态的散列或校验和值。在该计算之后，保护代理在操作系统引导之前、之后或同时扫描资源。在扫描后，保护代理计算结果的散列或校验和，并将此与初始状态散列或校验和值进行比较。如果相等，则保护代理确定对应的资源尚未被更改。当然，保护代理可绕过散列或校验和值并改为直接将初始状态与扫描结果进行比较。

[0056] 然而，如果值不同，则保护代理和 / 或虚拟机监控程序可采取一个或多个响应动作。首先，保护代理本身可关闭操作系统或操作系统特权模式，或者它可指示虚拟机监控程序来这样做。再一次，因为保护代理驻留在虚拟机监控程序中并且因为虚拟机监控程序主存操作系统，所以这两个组件能够关闭操作系统。此外，因为保护代理驻留在虚拟机监控程序中，所以即使从操作系统特权模式也无法篡改操作系统的关闭。

[0057] 除了关闭操作系统之外，保护代理和 / 或虚拟机监控程序可首先向操作系统警告即将来临的关闭。虚拟机监控程序和操作系统之间的通信信道可允许这一通信。在替换实

施例中，保护代理和 / 或虚拟机监控程序可向存储单元写入警告或发信号通知操作系统所监视的事件。

[0058] 不考虑是否已给予警告，操作系统关闭可以是突然的或者优雅的。在前一种情况下，虚拟机监控程序可仅在获悉不同的散列或校验和值后立即关闭操作系统。在后一种情况下，虚拟机监控程序可给予操作系统一特定时间量来将其自身干净地关闭。此时，操作系统可例如关闭任何打开的文件并转储清除任何相应的数据。操作系统还可释放所分配的资源。此外，该关闭可利用两种方法。例如，如果虚拟机监控程序主存多个分区，则它可立即关闭具有不同的散列或校验和值的分区，同时允许其他分区干净地关闭。在任何情况下，关闭的方式可根据策略来配置并且可以是可调整的。

[0059] 除了关闭和相应的警告之外，保护代理和 / 或虚拟机监控程序可响应于对 所标识资源的不被允许的更改来采取后引导动作。例如，虚拟机监控程序和 / 或保护代理可在重新引导操作系统时将资源更改通知给操作系统。作为响应，操作系统可执行反病毒扫描以检测任何恶意软件是否的确驻留在诸如部分 208（例如，内核）等操作系统存储器中。此外，虚拟机监控程序可将操作系统引导至安全模式中，或者操作系统本身可选择引导至安全模式中。同样响应于通知，操作系统可将其本身标识为已遭到攻击，并由此可不允许其本身访问其耦合的任何网络。

[0060] 虚拟分区保护代理

[0061] 保护代理（例如，图 1 的保护代理 142）可驻留在单独的虚拟分区（例如，图 1 的第二虚拟分区 112）中，而不是驻留在虚拟机监控程序本身中。在这些实施例中，该单独分区担当虚拟机监控程序的可信代表。保护代理 142 由此不可从操作系统特权模式访问。如上所述，虚拟机监控程序 108 允许这样的对计算设备 102 的虚拟化。虽然虚拟机监控程序可将计算设备虚拟化为任何数量的分区，但图 1 示出了主存操作系统的第一分区以及主存保护代理的第二分区。保护代理驻留在其中的第二虚拟分区在某些情况下可以是专用安全分区，其主要或唯一功能是运行保护代理。在其他实施例中，该第二虚拟分区可执行附加功能，诸如主存另一操作系统等。

[0062] 驻留在第二虚拟分区中的保护代理 142 能够执行许多或所有与以上关于驻留在虚拟机监控程序中的保护代理 134 所描述的相同的功能。即，保护代理 142 可主动或被动地接收一个或多个操作系统资源 120 的标识。响应于该标识，该保护代理同样可扩展保护代理服务 140，其一般包括确定所标识资源中的一个或多个是否已被更改，并且如果是则采取响应动作。这些服务还可包括强制实施指定资源的不变性。保护代理 142 可经由与上述技术类似的技术来执行这些功能。

[0063] 如图所示，保护代理 142 可从第二虚拟分区特权模式 130 访问，但不可从操作系统特权模式 126 访问。由此，所得体系结构允许对保护代理本身进行保护以免遭位于操作系统中的任何恶意软件的攻击，即使该恶意软件驻留在与该操作系统的最低层组件相关联的存储器部分 108 中。

[0064] 自主保护代理特权模式

[0065] 本章节描述了能够使得与保护代理相关联的操作系统存储器部分不可从操作系统特权模式更改或访问，同时仍旧允许该存储器部分物理地驻留在操作系统物理存储空间中。这些工具由此创建自主保护代理特权模式，其可访问与保护代理相关联的存储器部分

以及可在操作系统特权模式中访问的其余存储器部分。该特权模式因此比操作系统特权模式更具特权。

[0066] 第一小节描述了能够通过请求虚拟机监控程序保护与保护代理相关联的存储器部分来创建保护代理特权模式的工具。同时，第二小节描述了允许通过将物理处理器虚拟化为包括用于运行保护代理的专用虚拟处理器在内的多个虚拟处理器来创建保护代理特权模式的工具。

[0067] 向虚拟机监控程序作出的保护请求

[0068] 本小节描述了保护代理可如何请求虚拟机监控程序保护与保护代理相关联的存储器并由此对保护代理本身进行保护。该保护导致保护代理 144 以保护代理特权模式 132 操作，如图 1 所示。如图所示，保护代理 144 在移位到保护代理特权模式之前最初可驻留在操作系统特权模式中。在以该后一特权模式操作时，该保护代理一般不会受到来自以操作系统特权模式 126 操作的实体的攻击的影响。

[0069] 在以保护代理特权模式 132 操作时，实体具有比在以操作系统特权模式 126 操作的情况下稍多的特权，但所具有的特权仍然少于虚拟机监控程序特权模式 124。如图 2 所示，以该特权模式操作的保护代理可访问除与该保护代理本身相关联的存储器部分 206 之外的与操作系统相关联的所有存储器。虚拟机监控程序 108 强制实施所添加的保护代理可访问性。

[0070] 图 3 和 4 示出了创建该保护代理特权模式的示例性方式。图 3 描绘了所有或基本上所有计算设备存储器 300。计算设备存储器 300 包括与操作系统特权模式（例如，内核）相关联的存储器部分 302 以及与用户特权模式相关联的存储器部分 304。如图所示，存储器部分 302 还包括与保护代理 144 相关联的存储器部分 306 以及驱动程序加载在其中的存储器部分 308。

[0071] 如图 4 所示，创建保护代理特权模式 132 的过程 400 通过初始化存储器部分 302（例如，内核）在动作 1 开始。在动作 2，存储器部分 306 或保护代理 144 本身调用虚拟机监控程序 108 来请求该虚拟机监控程序保护与该保护代理相关联的存储器部分。在这样请求时，保护代理或对应的存储器要求不允许更改在操作系统特权模式中运行的代码或以其他方式接触该存储器部分 306。保护代理还可向虚拟机监控程序 108 进行自我验证（例如，通过数字签名）。该存储器部分或保护代理本身还可请求虚拟机监控程序设置定时器并在该定时器到期时运行该保护代理。动作 3 表示虚拟机监控程序响应于该请求来保护该存储器免遭在操作系统特权模式中操作的实体的攻击并设置定时器。注意，因为与保护代理相关联的该存储器部分 306 现在是不可更改和 / 或不可从操作系统特权模式访问的，所以该保护代理现在驻留在保护代理特权模式中。

[0072] 在动作 4，驱动程序加载到存储器部分 308 中。注意，动作 2 的请求和动作 3 的相应保护一般在驱动程序加载到存储器中之前进行，因为恶意软件可能以驱动程序的形式存在。如在以下“对工具的示例性使用”章节中所讨论地，恶意软件制作者通常欺骗用户将恶意驱动程序安装到计算设备中。如果一个或多个恶意驱动程序的确在保护存储器部分 306 之前加载到存储器中，则这些恶意驱动程序可能遮蔽该请求以保护其本身。这一遮蔽将由此经由虚拟机监控程序阻碍保护代理的周期性运行，并因此阻碍保护代理特权模式的创建。然而，通过请求虚拟机监控程序在早期设置定时器，该过程确保在操作系统特权模式中

运行的代码无法这样禁止保护代理的周期性运行。

[0073] 同时，动作 5 可能在驱动程序已经被加载后的某一时刻进行。如图所示，动作 5 表示虚拟机监控程序定时器到期，并因此运行保护代理。在运行时，保护代理 144 执行与先前章节中所讨论的功能相似或相同的功能。同样如上所述，保护代理可响应于确定一个或多个所标识资源已被更改来采取动作。保护代理还可响应于来自在操作系统特权模式中操作的实体对保护代理或其对应存储器的访问或更改尝试来采取这一动作。

[0074] 动作 6 表示保护代理在该保护代理完成运行时通知虚拟机监控程序。最后，动作 7 表示重复动作 3、5 和 6。由此，虚拟机监控程序可重置其定时器并以诸如每 100 毫秒 (ms) 等周期性间隔运行保护代理。

[0075] 通过在虚拟机监控程序处设置故障安全定时器，过程 400 由此消除了操作系统代码篡改与保护代理相关联的存储器部分的能力。由此，该过程确保保护代理将继续运行并且不会被在操作系统特权模式中行动的恶意软件遮蔽。相反，保护代理将在自主特权模式中运行，同时仍旧驻留在分配给操作系统的物理存储器中。

[0076] 保护代理虚拟处理器

[0077] 本小节描述了虚拟机监控程序可如何通过调度虚拟处理器运行保护代理 144 来创建保护代理特权模式。图 5 示出了包括将计算设备 102 虚拟化为各自包括一操作系统的两个分区的虚拟机监控程序 108 的体系结构 500。如图所示，该示例中的计算设备包括两个真实处理器 104(a) 和 104(b)，虚拟处理器可在这两个处理器中的每一个上调度多个虚拟处理器。同样如图所示，虚拟机监控程序创建第一虚拟分区 502 和第二虚拟分区 504。第一虚拟分区包括用于运行第一操作系统的第一虚拟处理器 506。类似地，第二虚拟分区包括用于运行第二操作系统的第二虚拟处理器 508。然而，在此实例中，虚拟机监控程序还包括用于运行诸如图 1 的保护代理 144 等保护代理的保护代理虚拟处理器 510。

[0078] 为了创建体系结构 500，虚拟机监控程序首先加载并初始化。如图 6 所示，虚拟机监控程序然后虚拟化各个虚拟处理器并在这样做时，分配真实处理器带宽 600。为了开始该虚拟化和分配，虚拟机监控程序在第一真实处理器上虚拟化第一虚拟处理器。在当前示例中，该虚拟化是在如图 6 所示的一对一的基础上完成的。即，只有该单个虚拟处理器 506 对应于真实处理器 104(a)，并且由此虚拟机监控程序将该真实处理器的所有带宽都分配给该虚拟处理器。虚拟机监控程序然后在第二真实处理器 104(b) 上虚拟化第二虚拟处理器 508。然而，虚拟机监控程序保留第二真实处理器的带宽的某一部分，而不是在一对一基础上。同样如图 6 所示，虚拟机监控程序然后在第二真实处理器 104(b) 的其余带宽上虚拟化保护代理虚拟处理器 510。

[0079] 在第二真实处理器上操作的每一个虚拟处理器一般都在时间分片的基础上行动。即，第二虚拟处理器可在该第二虚拟处理器的操作挂起之前的某一时间量内在第二真实处理器上操作。此时，该第二真实处理器在某一其他时间量 内切换到保护代理虚拟处理器的操作。例如，第二虚拟处理器可在第二真实处理器上操作 90 毫秒，此时该第二虚拟处理器的操作挂起并且保护代理虚拟处理器的操作开始 10 毫秒。保护代理虚拟处理器对于两个操作系统分区以及第一和第二虚拟处理器两者而言一般都是透明的。由此，两个操作系统都相信其对应的虚拟处理器对应于相应的真实处理器。

[0080] 除了分配真实处理器带宽之外，虚拟机监控程序还管理每一个虚拟处理器都可访

问的存储器部分。在当前示例中，第一虚拟处理器可访问与第一操作系统相关联的所有存储器。同时，第二虚拟处理器可访问除了与保护代理相关联的存储器部分之外的与第二操作系统相关联的所有存储器。仅保护代理虚拟处理器可访问除了分配给第二操作系统的存储器之外的与保护代理相关联的存储器部分。

[0081] 此外，第一和第二虚拟处理器只具有更改其相关联存储器的能力。由此，操作其各自操作系统的虚拟处理器都无法更改与保护代理相关联的存储器部分。然而，保护代理虚拟处理器可更改与该保护代理相关联的存储器，并且在某些实施例中，也可更改与第二虚拟处理器相关联的存储器。

[0082] 按照其程序化特性，保护代理虚拟处理器将周期性地运行保护代理。虽然在某些实例中保护代理虚拟处理器可运行其他应用程序，但当前示例示出了专用的保护代理虚拟处理器。由此，该虚拟处理器一般只用于周期性地运行保护代理。同样，保护代理可以按与上述保护代理相似或相同的方式执行与上述保护代理相似或相同的功能。

[0083] 通过调度专用的保护代理虚拟处理器，虚拟机监控程序确保保护代理将在该处理器的控制下并以自主保护代理特权模式周期性地运行。此外，因为只有该保护代理虚拟处理器可访问与保护代理相关联的存储器部分，所以虚拟机监控程序保护该存储器免遭操作系统中的代码的攻击。因此，在操作系统特权模式中操作的恶意软件无法遮蔽保护代理及防止保护代理运行。由此，该技术基本上消除了操作系统篡改保护代理的能力。

[0084] 对工具的示例性使用

[0085] 在先前描述了能够确保对保护代理进行保护的工具之后，以下章节仅描述 了这些工具在操作中的一个示例。首先，想象计算机用户在因特网上冲浪，并且同时在特定网站上冲浪，具有恶意企图的对话框在该用户的显示器上弹出。该对话框请求用户许可将某种恶意软件安装在该用户的计算机上。尽管该请求可以是直接的，但想象该对话框通常对该请求进行伪装。例如，该对话框可虚假地通知该用户他或她中奖了。在这样通知时，该对话框恶意地指示用户键击该对话框上的“OK”按钮以领取奖品。想象用户的确选择了 OK 按钮并且该用户选择继续所请求的操作而不管来自在计算设备上运行的软件（例如，反病毒应用程序）的一个或多个警告。

[0086] 此时，计算设备开始安装包含恶意软件的驱动程序。如对于驱动程序一般是如此的，该恶意驱动程序被准予访问操作系统特权模式并加载到与该特权模式相关联的存储器（例如，内核）中。一旦在内核中加载，恶意驱动程序及其附随的恶意软件本质上具有对计算机的存储器和操作系统的全权委托 (carteblanch) 访问权。对于用户而言不幸的是，想象该恶意软件包括记录用户的键击的按键记录器。现在想象用户导航到他或她的银行网站并登入到他或她的银行账户。由于其记录键击的能力，按键记录器获悉该用户的银行账户密码并将该密码通过因特网发送到该恶意驱动程序的制作者。

[0087] 使情况变得更糟糕地，想象该恶意软件是“rootkit”，即尝试主动对保护代理和用户的反病毒软件进行隐藏的恶意软件。在常规系统中，保护代理驻留在内核（即，在恶意软件可访问的存储器）中。因此，在这些常规系统中，恶意软件可访问保护代理并可尝试对该保护代理隐藏其本身。如果成功，则对于保护代理而言恶意软件将会看上去并不存在于内核中。因此，当用户的反病毒软件调用保护代理并请求存在于计算机的存储器中的所有应用程序的列表时，该恶意软件将不在其中。该不在其中致使反病毒软件没有能力知晓并移

除恶意软件。此外，恶意软件可遮蔽保护代理，由此从根本上防止该保护代理运行。由此，保护代理可能无法注意到恶意软件是否更改任何操作系统资源。

[0088] 然而，想象用户的计算设备上的保护代理驻留在存储器中或以不可从操作系统特权模式访问的模式运行，而不是如在常规系统中地驻留在内核中。因此，当恶意驱动程序加载到内核中时，它不可访问保护代理驻留在其中的存储器或保护代理运行的模式。因此，驱动程序及其附随恶意软件不可访问保护代理本 身。恶意软件由此无法对保护代理隐藏其本身并因此也无法对反病毒软件隐藏其本身。因此，当反病毒软件向保护代理询问存在于计算机的存储器中的所有应用程序的列表时，所返回的列表包括恶意软件。反病毒软件然后将该代码识别为恶意软件并因此将其从用户的计算机设备中移除。此外，保护代理本身可注意到恶意软件是否更改操作系统资源并作为响应，可关闭用户的计算设备。

[0089] 因此，通过驻留在存储器中或以不可从操作系统特权模式访问的模式运行，此处所描述的各实施例防止恶意软件对保护代理隐藏其本身或遮蔽保护代理。在以上示例中，用户的计算设备因此能够从该机器中移除恶意软件或者在某些实例中，在恶意软件更改重要资源时关闭系统。在任一种情况下，这些实施例都用于减少恶意软件期望造成损害的有效性。

[0090] 工具的其它实施例

[0091] 以上各章节描述了其中使得保护代理不可从操作系统特权模式更改或访问的几个特定示例。在本章节中，描述工具的其他实施例，诸如将不存在于底层处理器上的特权模式添加到处理器等。

[0092] 这些示例性实施例作为图 7 到 11 的过程 700 到 1100 的一部分来描述。参考图 1 到 6 描述或示出的这些过程以及各示例性过程能以任何合适的硬件、软件、固件、或其组合来实现；在软件和固件的情况下，这些过程表示被实现为存储在计算机可读介质中并由一个或多个处理器执行的计算机可执行指令的操作集合。本章节中所描述的工具的这些实施例不旨在限制该工具或权利要求的范围。

[0093] 参考图 7，框 702 接收标识一个或多个操作系统资源的强制实施策略。可包括加密数据的该强制实施策略可经由经数字签名的清单或通过向操作系统展示应用程序编程接口 (API)（例如，超调用）来接收。框 704 从不可从在操作系统特权模式中操作的实体访问的存储器中标识一个或多个操作系统资源。示例性资源包括系统服务分派表 (SSDT)、中断分派表 (IDT) 和 / 或全局描述符表 (GDT)。如上所述，该标识可以在虚拟机监控程序（例如，由图 1 的保护代理 134）或单独的虚拟分区（例如，由图 1 的保护代理 142）中进行。

[0094] 同时，框 706 表示确定所标识资源中的任一个是否已被更改。同样，这可 以在虚拟机监控程序或单独的分区中进行。如果框 706 确定所标识资源中的一个或多个的确已被更改，则框 708 响应于该判定来终止操作系统。最后，框 710 在重新引导操作系统时将非法操作通知给该操作系统。

[0095] 图 8 示出了用于允许保护代理在虚拟机监控程序中运行的过程 800。框 802 更改虚拟机监控程序侦听管理器以便有效地允许接收与操作系统资源相关联的存储器页或寄存器已被更改的指示。该资源可包括参考图 7 描述的资源中的一个，或者可以是另一操作系统资源。在任何情况下，框 804 接收标识操作系统资源以及可能的一个或多个其他操作系统资源的强制实施策略。同样，该标识可经由以上所讨论的各种技术来完成。如上所述，

资源的保护属性（例如，“只读”或“初始化只读”）可伴随资源的标识。同时，框 806 表示接收与操作系统资源相关联的存储器页或寄存器的确已被更改的指示。作为响应，框 808 关闭操作系统特权模式以便有效地关闭与操作系统资源相关联的操作系统。在某些实例中，图 1 的虚拟机监控程序 108 可完成此对操作系统特权模式的关闭。

[0096] 接着，图 9 描述了用于创建如图 1 所示的保护代理特权模式 132 的保护代理特权模式的示例性过程 900。框 902 接收使得存储器的特定范围不可从操作系统特权模式更改或访问的请求。同样，虚拟机监控程序可接收该请求，该请求可源自该存储器范围本身或来自驻留在该存储器范围内的保护代理。框 904 保护该存储器范围并设置定时器以便周期性地运行驻留在该存储器范围内的保护代理。同样，虚拟机监控程序可设置这一定时器，该定时器可指示虚拟机监控程序以规律的间隔运行保护代理。

[0097] 同时，框 906 接收描述操作系统资源的强制实施策略。同样，该强制实施策略和所描述的资源可以与上述强制实施策略和资源相似或相同。框 908 运行可由虚拟机监控程序实现的保护代理。判定框 910 询问操作系统资源是否已被更改。保护代理可通过以上文中所详述的方式起作用来作出该判定。如果框 910 的确确定已发生更改，则框 912 关闭操作系统。然而，如果未作出这一判定，则框 914 接收保护代理已完成运行的通知。在某些实例中且如上所述，保护代理本身可这样通知虚拟机监控程序。同时，框 916 表示在运行保护代理和不运行保护代理之间循环。最后，注意，虽然保护代理不运行，但虚拟机监控程序可响应于来自在操作系统特权模式中操作的实体对与该保护代理相关联的存 储器范围的访问尝试来关闭操作系统。

[0098] 图 10 示出了用于创建如图 1 所示的保护代理特权模式 132 的保护代理特权模式的另一示例性过程 1000。框 1002 将真实计算机处理器虚拟化为多个虚拟计算机处理器。这些虚拟处理器可包括一个或多个操作系统虚拟处理器，其各自具有更改其自己的操作系统存储器并且如图 6 所示地使用真实处理器的处理带宽的一部分的特权。虚拟处理器还可包括至少一个保护代理虚拟处理器，其具有更改其自己的保护代理存储器并使用真实处理器的处理带宽的不同部分的特权。虽然所有虚拟处理器都可由虚拟机监控程序来调度，但保护代理虚拟处理器对于操作系统虚拟处理器可以是透明的。在某些实例中，操作系统虚拟处理器可能无法更改分配给保护代理虚拟处理器的存储器。此外，保护代理虚拟处理器可以是专用处理器，其主要或唯一目的是如上所述地使得保护代理执行。

[0099] 接着，框 1004 使得保护代理虚拟处理器执行保护代理，该保护代理可有效地确定所述操作系统存储器部分是否已被更改。同时，框 1006 接收操作系统存储器的一部分已被更改的指示。作为响应，框 1008 关闭对应的操作系统。

[0100] 最后，图 11 描绘了用于将特权模式添加到真实计算处理器的过程 1100。框 1102 表示对存在于底层物理处理器上的一个或多个特权模式进行确定、标识或分类。这些特权模式一般由底层物理处理器本身来定义。无论如何，框 1104 添加不存在于底层物理处理器上的特权模式。在某些实例中，所添加的特权模式能够更改计算设备中与可由一个或多个现有特权模式更改的存储器部分不同的存储器部分。所添加的特权模式还能够添加并执行先前不存在或不可在底层处理器中执行的指令。

[0101] 此外，存在于底层物理处理器上的一个或多个特权模式可包括用户特权模式和操作系统特权模式。在这些实施例中，所添加的特权模式可以比用户特权模式和操作系统特

权模式两者都更具特权，比用户特权模式更具特权但比操作系统特权模式所具特权少，或者比用户和操作系统特权模式两者所具特权都少。最后，注意，添加特权模式的一个实例可包括以上述多种方式添加保护代理特权模式（例如，图 1 所示的保护代理特权模式 132）。例如，保护代理及其相关联的存储器范围可请求使该存储器范围不可从在操作系统特权模式中 操作的实体访问。虚拟机监控程序还可通过调度保护代理虚拟处理器运行保护代理来创建该特权模式。

[0102] 结论

[0103] 上述工具能够通过使保护代理能驻留在不可从操作系统特权模式访问的存储单元或者通过创建保护代理特权模式来使得保护代理不可从操作系统特权模式更改或访问。虽然已经用对结构特征和 / 或方法动作专用的语言描述了该工具，但是应该理解，在所附权利要求中定义的该工具不必限于所述的具体特征或动作。相反，这些具体特征和动作是作为实现该工具的示例性形式而公开的。

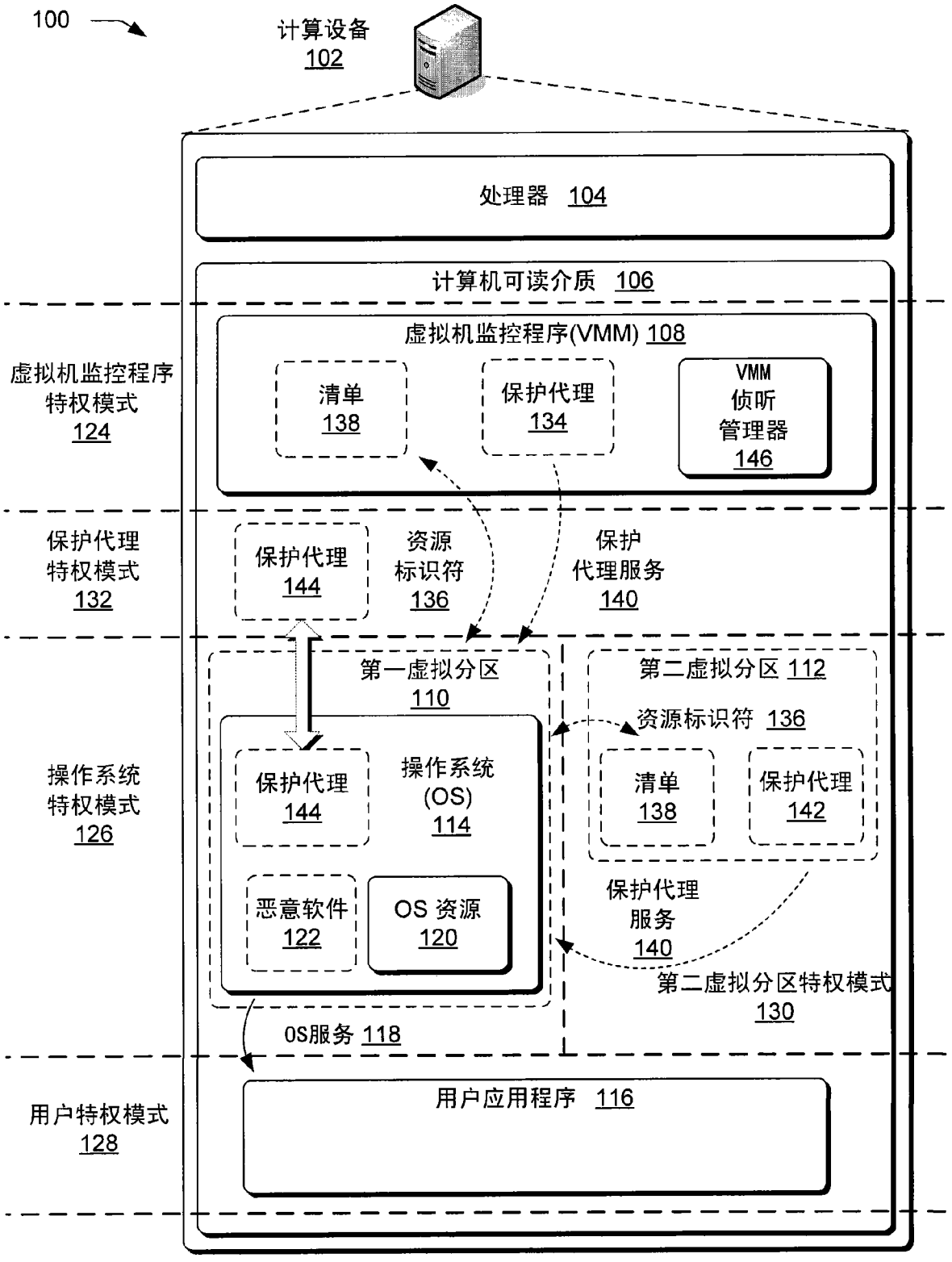

100
计算设备
102
处理器 104
计算机可读介质 106
虚拟机监控程序(VMM) 108
清单
138
保护代理
134
VMM
侦听
管理器
146
虚拟机监控程序
特权模式
124
保护代理
特权模式
132
保护代理
144
资源
标识符
136
保护
代理服务
140
第一虚拟分区
110
第二虚拟分区 112
资源标识符 136
保护代理
144
操作系统
(OS)
114
清单
138
保护代理
142
操作系统
特权模式
126
恶意软件
122
OS 资源
120
保护代理
服务
140
第二虚拟分区特权模式
130
OS服务 118
用户应用程序 116
用户特权模式
128

图 1

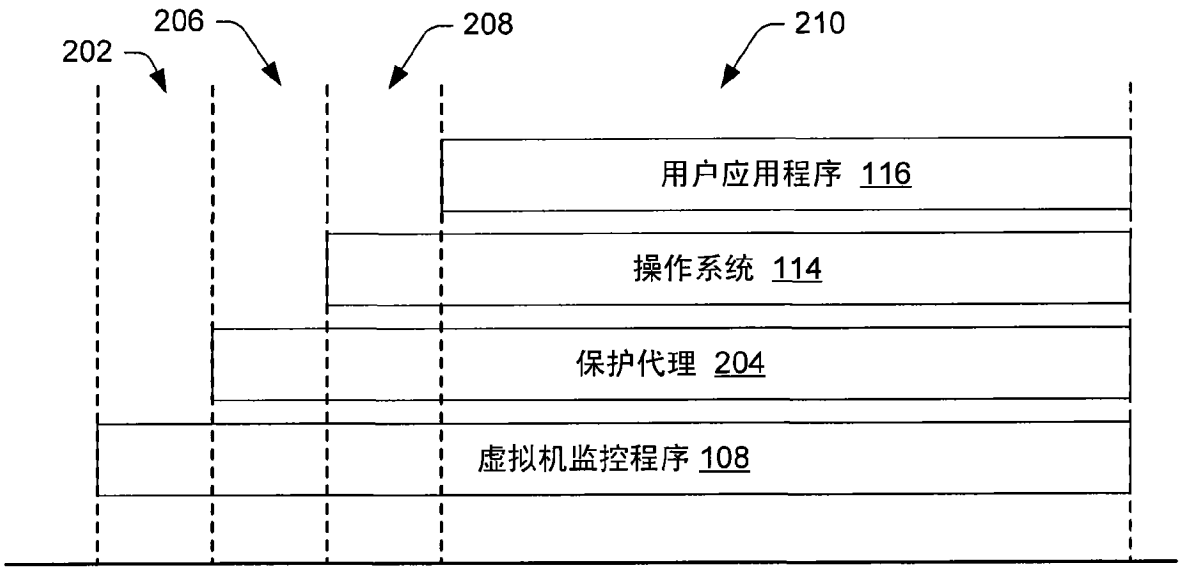

206
208
210
202
用户应用程序 116
操作系统 114
保护代理 204
虚拟机监控程序 108
计算设备存储器权限
200

图 2

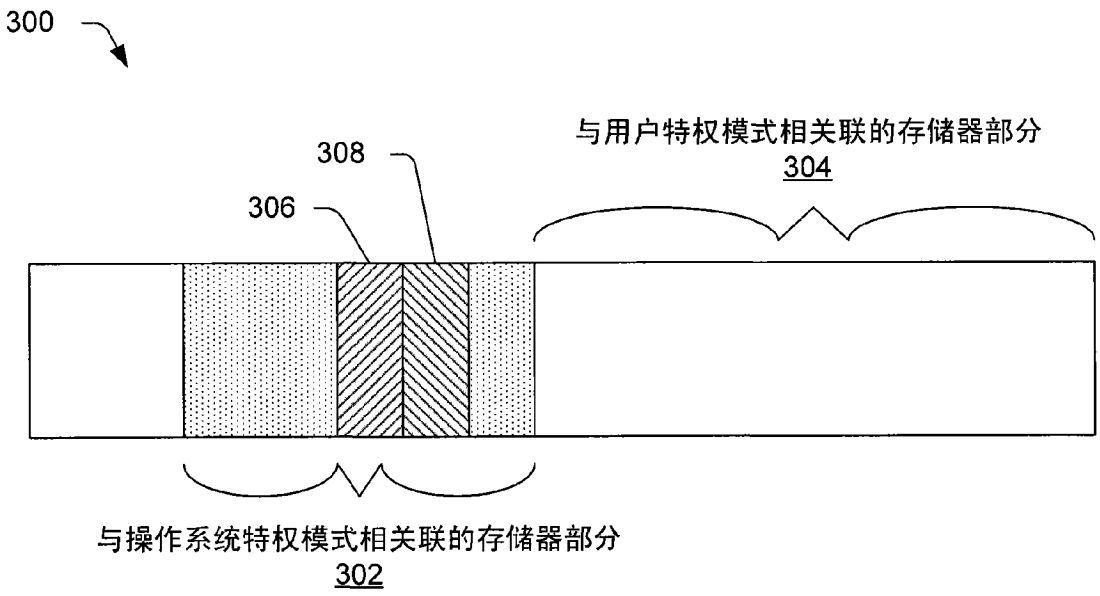

300
与用户特权模式相关联的存储器部分
304
308
306
与操作系统特权模式相关联的存储器部分
302

图 3

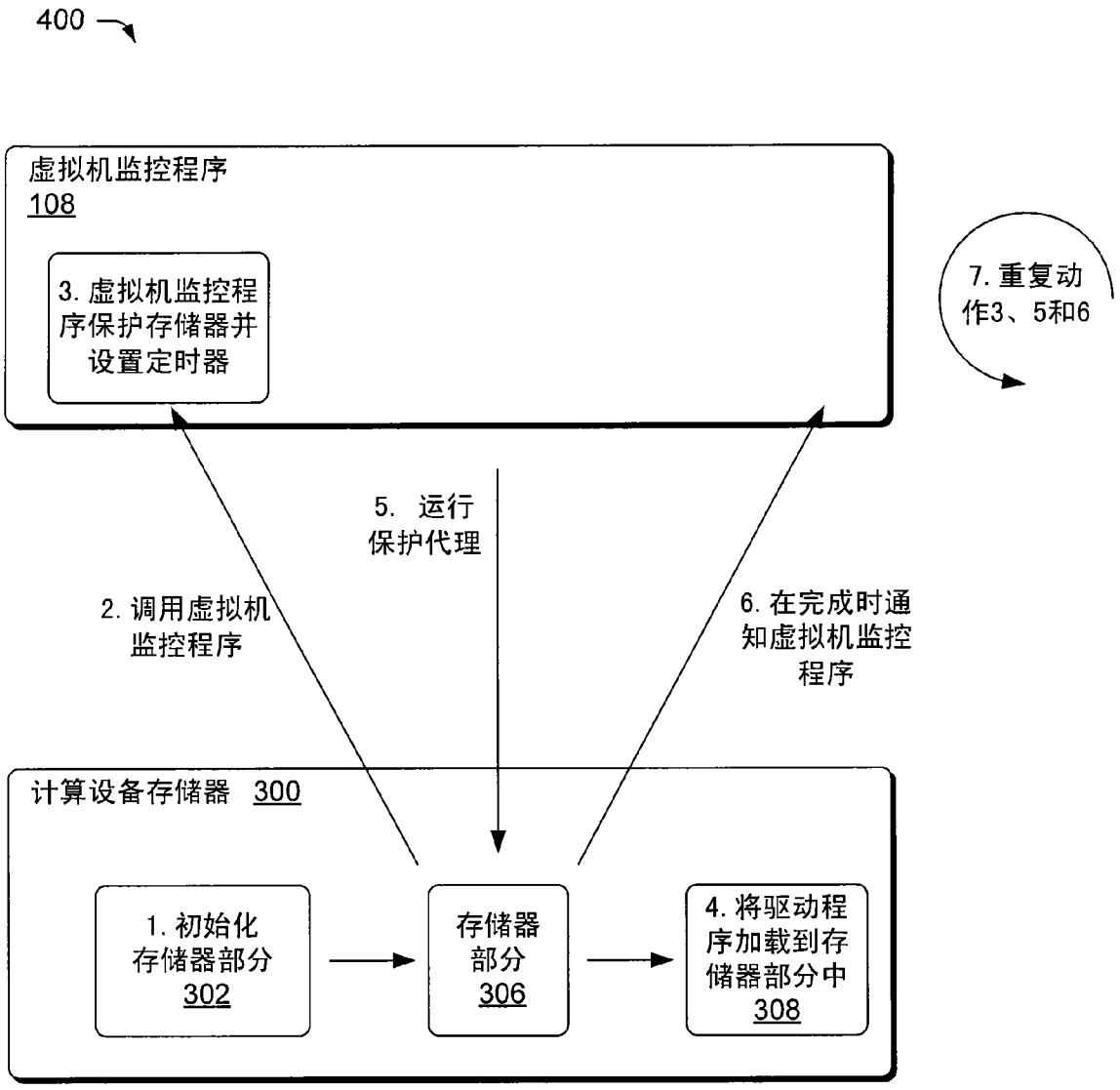

400
虚拟机监控程序
108
3. 虚拟机监控程序保护存储器并设置定时器
7. 重复动作3、5和6
5. 运行保护代理
2. 调用虚拟机监控程序
6. 在完成时通知虚拟机监控程序
计算设备存储器 300
1. 初始化存储器部分
302
存储器部分
306
4. 将驱动程序加载到存储器部分中
308

图 4

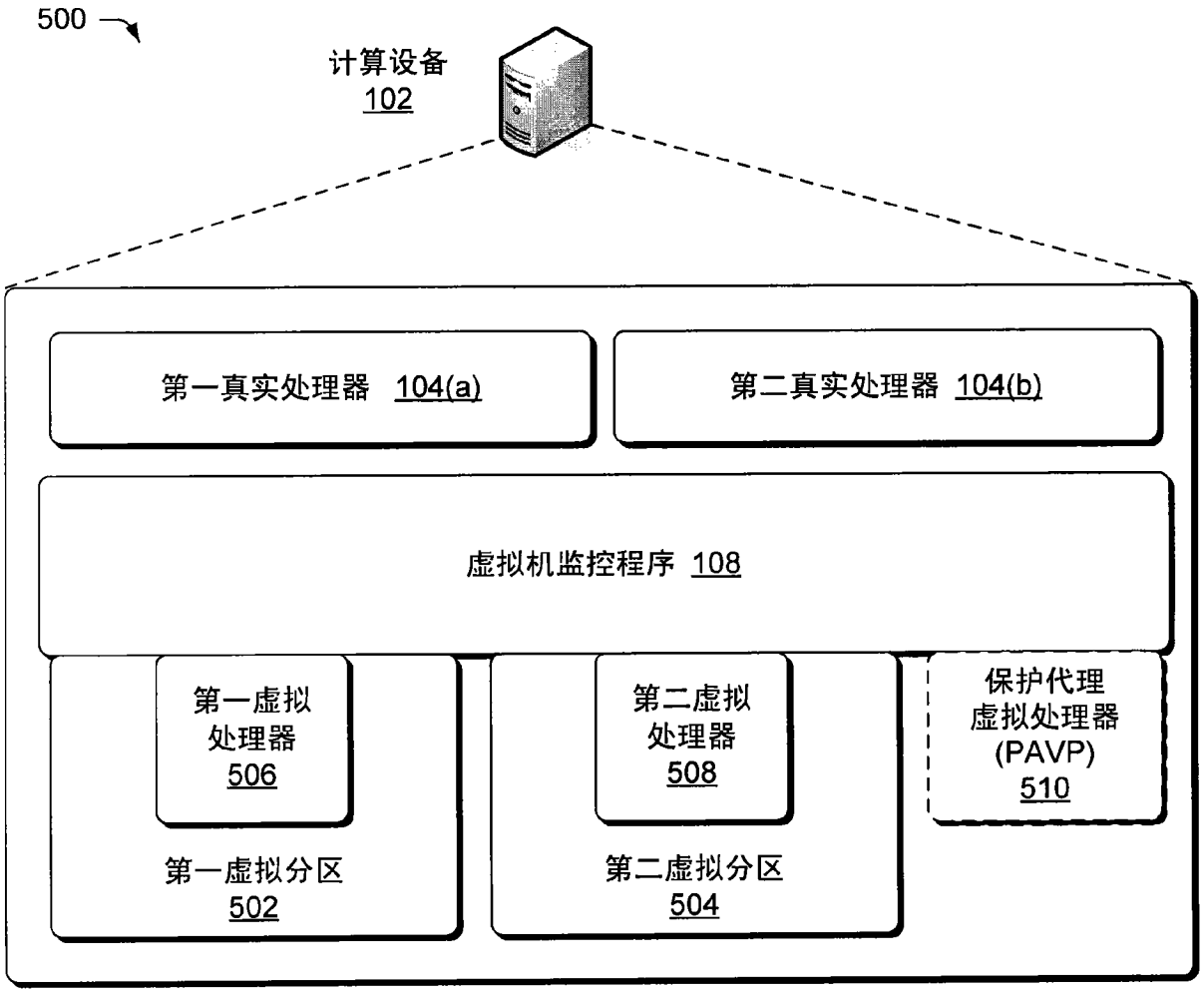

500
计算设备
102
第一真实处理器 104(a)
第二真实处理器 104(b)
虚拟机监控程序 108
第一虚拟
处理器
506
第二虚拟
处理器
508
保护代理
虚拟处理器
(PAVP)
510
第一虚拟分区
502
第二虚拟分区
504

图 5

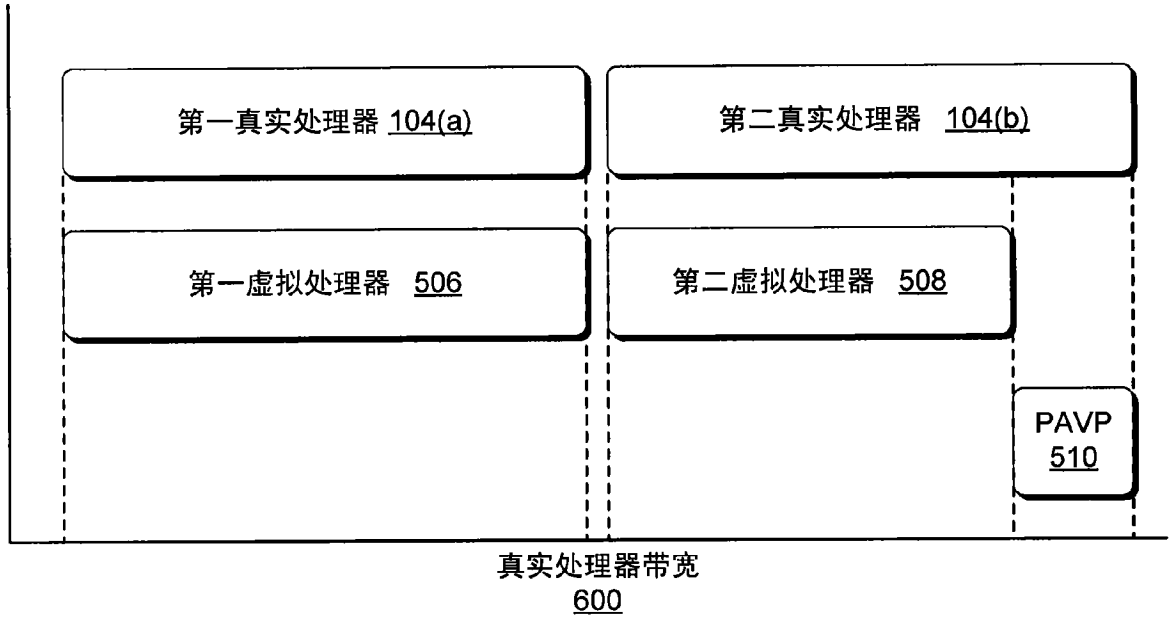

第一真实处理器 104(a)
第二真实处理器 104(b)
第一虚拟处理器 506
第二虚拟处理器 508
PAVP
510
真实处理器带宽
600

图 6

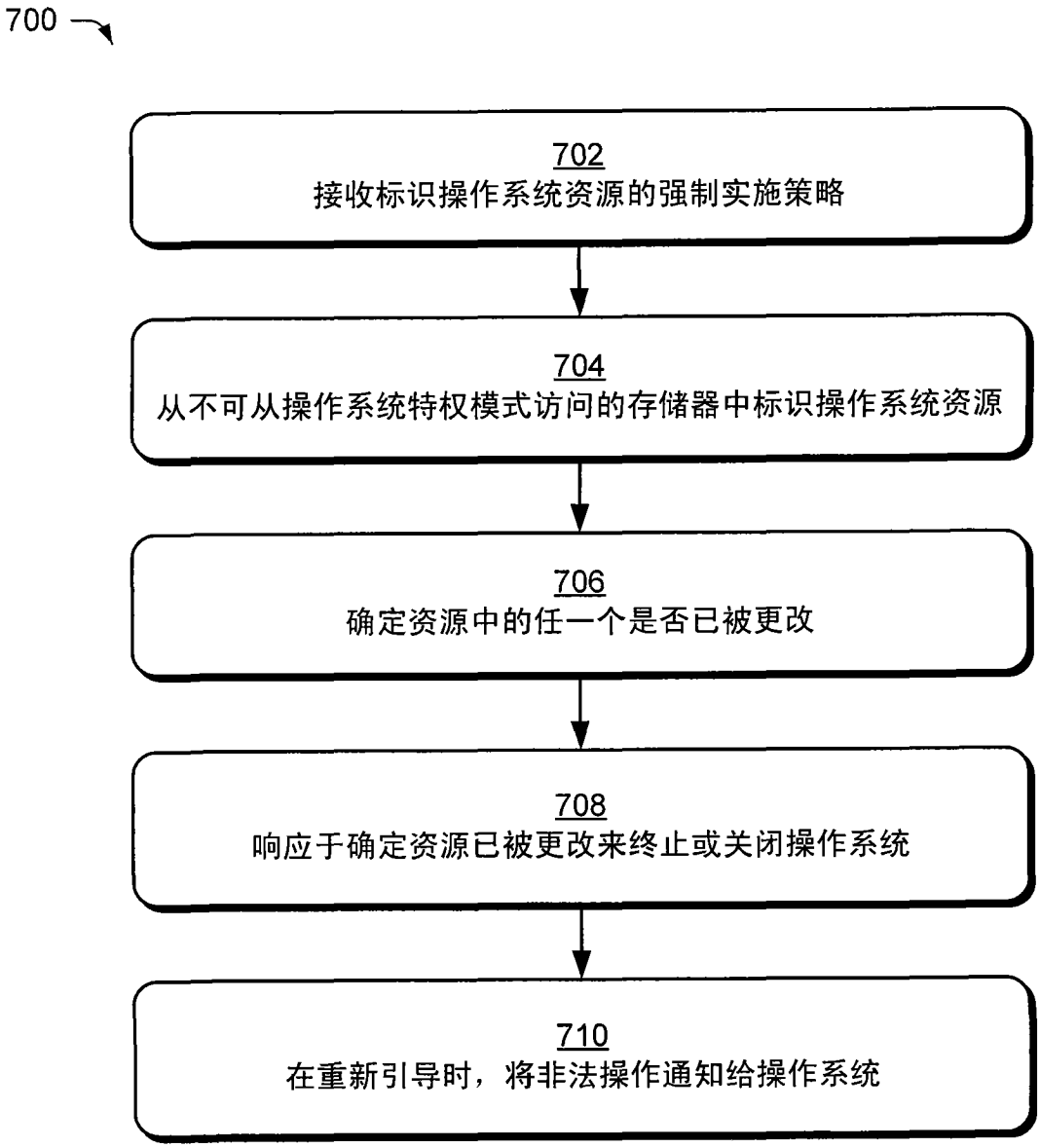

700
702
接收标识操作系统资源的强制实施策略
704
从不可从操作系统特权模式访问的存储器中标识操作系统资源
706
确定资源中的任一个是否已被更改
708
响应于确定资源已被更改来终止或关闭操作系统
710
在重新引导时，将非法操作通知给操作系统

图 7

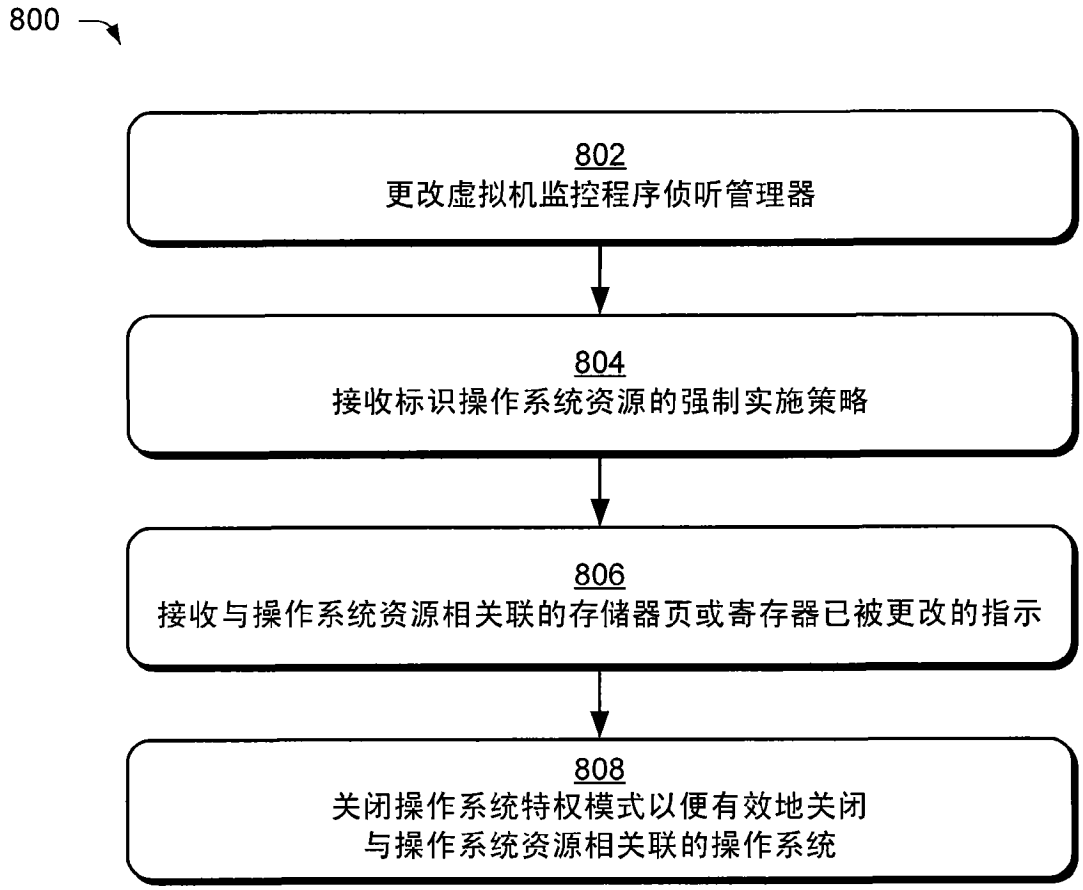

800
802
更改虚拟机监控程序侦听管理器
804
接收标识操作系统资源的强制实施策略
806
接收与操作系统资源相关联的存储器页或寄存器已被更改的指示
808
关闭操作系统特权模式以便有效地关闭
与操作系统资源相关联的操作系统

图 8

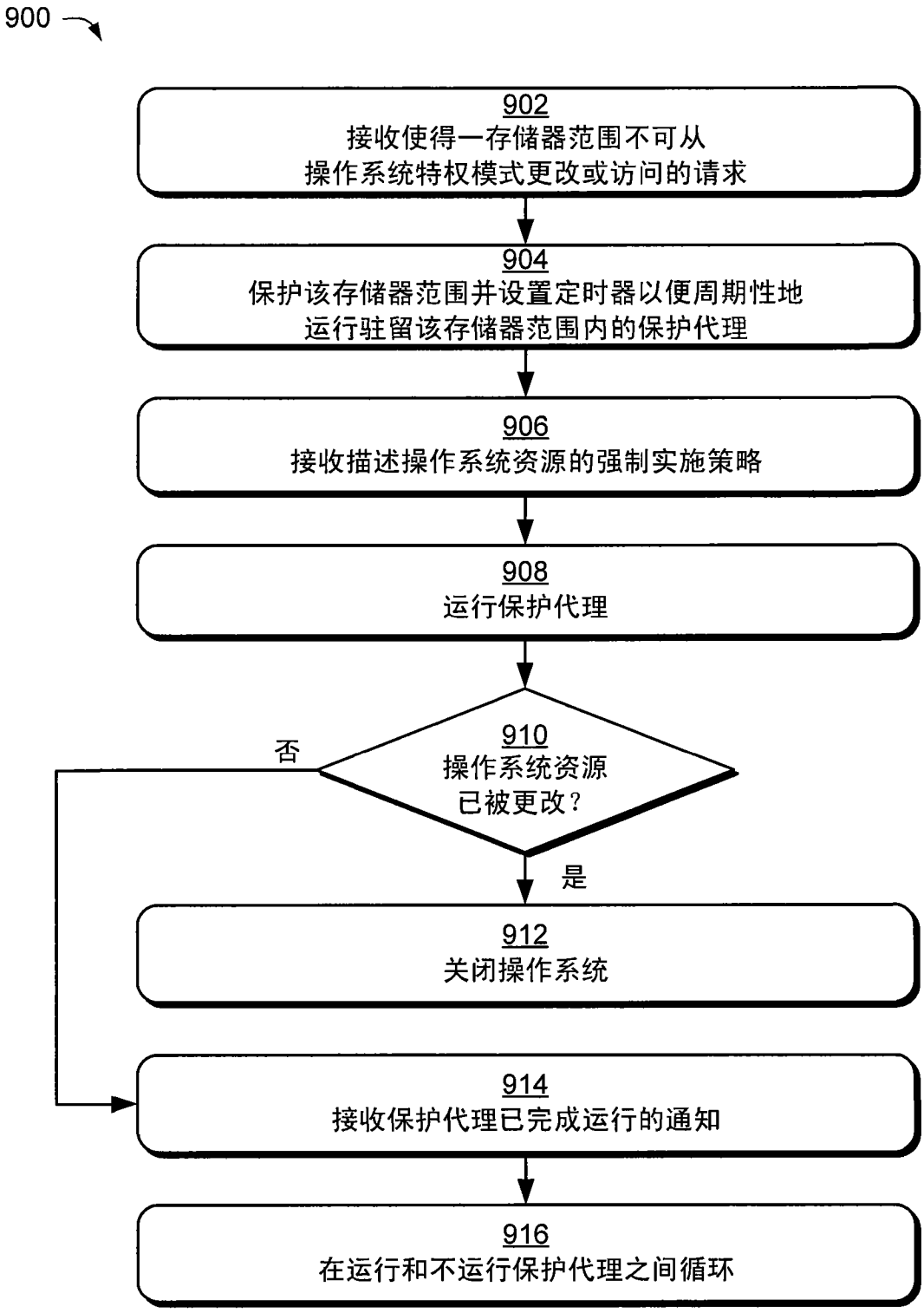

900
902
接收使得一存储器范围不可从
操作系统特权模式更改或访问的请求
904
保护该存储器范围并设置定时器以便周期性地
运行驻留该存储器范围内的保护代理
906
接收描述操作系统资源的强制实施策略
908
运行保护代理
910
操作系统资源
已被更改？
否
是
912
关闭操作系统
914
接收保护代理已完成运行的通知
916
在运行和不运行保护代理之间循环

图 9

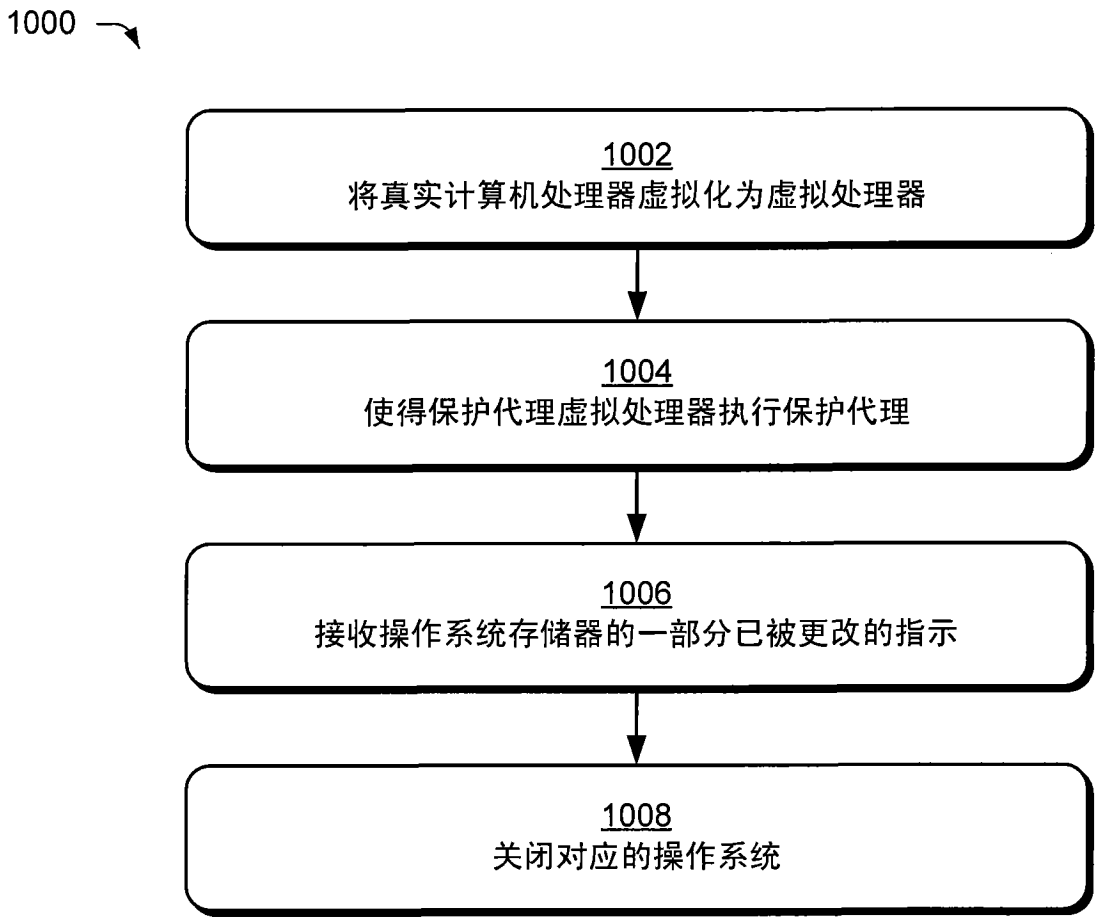

1000
1002
将真实计算机处理器虚拟化为虚拟处理器
1004
使得保护代理虚拟处理器执行保护代理
1006
接收操作系统存储器的一部分已被更改的指示
1008
关闭对应的操作系统

图 10

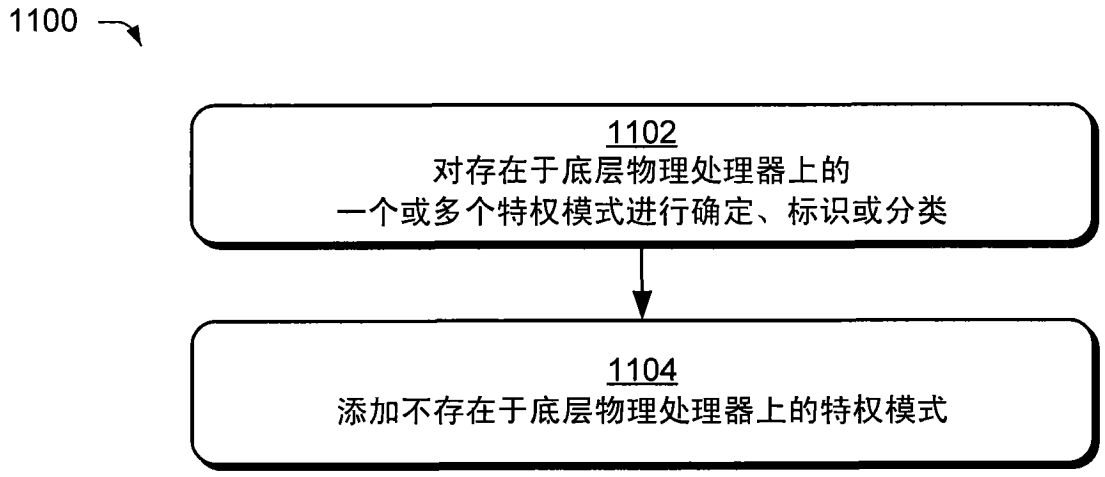

1100
1102
对存在于底层物理处理器上的
一个或多个特权模式进行确定、标识或分类
1104
添加不存在于底层物理处理器上的特权模式

图 11