(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 121233227 A
(43)申请公布日 2025.12.30

(21)申请号 202511129464.7

(22)申请日 2025.08.13

(71)申请人 南方科技大学
地址 518055 广东省深圳市南山区桃源街道学苑大道1088号

(72)发明人 张锋巍 汪湛博 展家鑫

(74)专利代理机构 广州嘉权专利商标事务所有限公司 44205
专利代理师 洪嘉兴

(51)Int.Cl.
G06F 9/455(2018.01)
G06F 9/445(2018.01)
G06F 9/4401(2018.01)
G06F 9/50(2006.01)
G06F 9/54(2006.01)

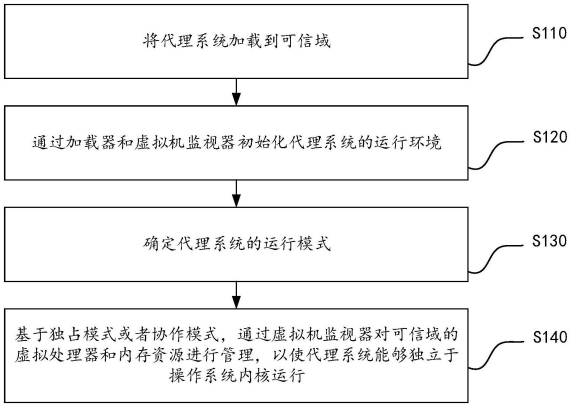
权利要求书2页 说明书13页 附图5页

(54)发明名称

虚拟机自省方法、装置、电子设备及存储介质

(57)摘要

本申请实施例提供了一种虚拟机自省方法、装置、电子设备及存储介质，属于计算机技术领域，该方法包括：将代理系统加载到可信域，代理系统包括可信代理组件和加载器，可信域还包括虚拟机的操作系统内核，虚拟机运行在可信域内，虚拟机处于安全仲裁模式，通过加载器和虚拟机监视器初始化代理系统的运行环境，虚拟机监视器处于非安全仲裁模式，确定代理系统的运行模式，运行模式为独占模式或者协作模式，基于独占模式或者协作模式，通过虚拟机监视器对可信域的虚拟处理器和内存资源进行管理，以使代理系统能够独立于操作系统内核运行，能够为可信域所有者提供一个安全的可信执行通道。



CN 121233227 A

1.一种虚拟机自省方法,其特征在于,所述方法包括:

将代理系统加载到可信域;其中,所述代理系统包括可信代理组件和加载器,所述可信域还包括虚拟机的操作系统内核,所述虚拟机运行在所述可信域内,所述虚拟机处于安全仲裁模式;

通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境;其中,所述虚拟机监视器处于非安全仲裁模式;

确定所述代理系统的运行模式;其中,所述运行模式为独占模式或者协作模式;

基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,以使所述代理系统能够独立于所述操作系统内核运行。

2.根据权利要求1所述的方法,其特征在于,所述可信域具有物理地址空间,所述通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境,包括:

通过所述加载器向所述虚拟机监视器发送初始化请求;

基于所述初始化请求,通过所述虚拟机监视器从所述物理地址空间确定所述代理系统的内存区域;

通过所述加载器根据所述内存区域的区域信息进行页表构建,得到可信域页表,并通过预设的可信域扩展模块为所述可信域分配安全扩展页表,以初始化所述代理系统的运行环境;其中,所述代理系统和所述操作系统内核共享所述安全扩展页表。

3.根据权利要求1所述的方法,其特征在于,所述基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,包括:

基于所述独占模式,通过所述加载器向所述虚拟机监视器发送目标请求;

基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理。

4.根据权利要求3所述的方法,其特征在于,所述虚拟处理器包括代理虚拟处理器和其他虚拟处理器,所述内存资源包括所述代理系统的内存区域,所述基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理,包括:

若所述目标请求用于请求所述虚拟机监视器将所述可信代理组件设置为休眠状态,则通过所述虚拟机监视器对所述内存区域进行内存阻断;

将所述代理虚拟处理器和承载所述代理虚拟处理器的逻辑处理器解绑,并将所述其他虚拟处理器调度到对应的所述逻辑处理器。

5.根据权利要求4所述的方法,其特征在于,所述基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理,包括:

若所述目标请求用于请求所述虚拟机监视器将所述可信代理组件设置为激活状态,则通过所述虚拟机监视器对所述内存区域进行阻断解除;

将所述其他虚拟处理器与对应的所述逻辑处理器解绑,并将所述代理虚拟处理器调度到目标逻辑处理器。

6.根据权利要求3所述的方法,其特征在于,所述可信域具有物理地址空间,所述基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,包括:

基于所述协作模式,通过所述虚拟机监视器从所述物理地址空间中确定所述代理系统

的秘密工作区；

根据所述秘密工作区更新安全扩展页表；

通过所述加载器接收更新后的所述安全扩展页表，并屏蔽每个针对所述代理系统的外部中断，以启动所述代理系统；

通过所述虚拟机监视器对启动后的所述代理系统的代理虚拟处理器和所述秘密工作区进行管理。

7.根据权利要求6所述的方法，其特征在于，所述通过所述虚拟机监视器对启动后的所述代理系统的代理虚拟处理器和所述秘密工作区进行管理，包括：

通过所述虚拟机监视器将启动后的所述代理系统的代理虚拟处理器下线，并解除所述秘密工作区；

或者，

通过所述虚拟机监视器将所述代理虚拟处理器调度到相应的逻辑处理器，并随机初始化所述秘密工作区。

8.一种虚拟机自省装置，其特征在于，所述装置包括：

加载模块，用于将代理系统加载到可信域；其中，所述代理系统包括可信代理组件和加载器，所述可信域还包括虚拟机的操作系统内核，所述虚拟机运行在所述可信域内，所述虚拟机处于安全仲裁模式；

初始化模块，用于通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境；其中，所述虚拟机监视器处于非安全仲裁模式；

确定模块，用于确定所述代理系统的运行模式；其中，所述运行模式为独占模式或者协作模式；

管理模块，用于基于所述独占模式或者所述协作模式，通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理，以使所述代理系统能够独立于所述操作系统内核运行。

9.一种电子设备，其特征在于，所述电子设备包括存储器和处理器，所述存储器存储有计算机程序，所述处理器执行所述计算机程序时实现权利要求1至7任一项所述的方法。

10.一种计算机可读存储介质，所述计算机可读存储介质存储有计算机程序，其特征在于，所述计算机程序被处理器执行时实现权利要求1至7任一项所述的方法。

虚拟机自省方法、装置、电子设备及存储介质

技术领域

[0001] 本申请涉及计算机技术领域,尤其涉及一种虚拟机自省方法、装置、电子设备及存储介质。

背景技术

[0002] 虚拟机自省技术是一种在虚拟机外部获取虚拟机信息,并根据虚拟机信息对虚拟机的运行状态进行监控和分析的技术。相关技术中,通过特权分层方法在不信任虚拟机内核的前提下实现自省功能。然而,特权分层方法必须信任最高层或者等效的特权软件模块,一旦该特权软件模块所在层遭受攻击,整个虚拟机的安全性将完全丧失。

发明内容

[0003] 本申请实施例的主要目的在于提出一种虚拟机自省方法、装置、电子设备及存储介质,旨在为可信域所有者提供一个安全的可信执行通道。

[0004] 为实现上述目的,本申请实施例的第一方面提出了一种虚拟机自省方法,所述方法包括:

[0005] 将代理系统加载到可信域;其中,所述代理系统包括可信代理组件和加载器,所述可信域还包括虚拟机的操作系统内核,所述虚拟机运行在所述可信域内,所述虚拟机处于安全仲裁模式;

[0006] 通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境;其中,所述虚拟机监视器处于非安全仲裁模式;

[0007] 确定所述代理系统的运行模式;其中,所述运行模式为独占模式或者协作模式;

[0008] 基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,以使所述代理系统能够独立于所述操作系统内核运行。

[0009] 在一些实施例,所述可信域具有物理地址空间,所述通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境,包括:

[0010] 通过所述加载器向所述虚拟机监视器发送初始化请求;

[0011] 基于所述初始化请求,通过所述虚拟机监视器从所述物理地址空间确定所述代理系统的内存区域;

[0012] 通过所述加载器根据所述内存区域的区域信息进行页表构建,得到可信域页表,并通过预设的可信域扩展模块为所述可信域分配安全扩展页表,以初始化所述代理系统的运行环境;其中,所述代理系统和所述操作系统内核共享所述安全扩展页表。

[0013] 在一些实施例,所述基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,包括:

[0014] 基于所述独占模式,通过所述加载器向所述虚拟机监视器发送目标请求;

[0015] 基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理。

[0016] 在一些实施例,所述虚拟处理器包括代理虚拟处理器和其他虚拟处理器,所述内存资源包括所述代理系统的内存区域,所述基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理,包括:

[0017] 若所述目标请求用于请求所述虚拟机监视器将所述可信代理组件设置为休眠状态,则通过所述虚拟机监视器对所述内存区域进行内存阻断;

[0018] 将所述代理虚拟处理器和承载所述代理虚拟处理器的逻辑处理器解绑,并将所述其他虚拟处理器调度到对应的所述逻辑处理器。

[0019] 在一些实施例,所述基于所述目标请求,通过所述虚拟机监视器对所述虚拟处理器和所述内存资源进行管理,包括:

[0020] 若所述目标请求用于请求所述虚拟机监视器将所述可信代理组件设置为激活状态,则通过所述虚拟机监视器对所述内存区域进行阻断解除;

[0021] 将所述其他虚拟处理器与对应的所述逻辑处理器解绑,并将所述代理虚拟处理器调度到目标逻辑处理器。

[0022] 在一些实施例,所述可信域具有物理地址空间,所述基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对所述可信域的虚拟处理器和内存资源进行管理,包括:

[0023] 基于所述协作模式,通过所述虚拟机监视器从所述物理地址空间中确定所述代理系统的秘密工作区;

[0024] 根据所述秘密工作区更新安全扩展页表;

[0025] 通过所述加载器接收更新后的所述安全扩展页表,并屏蔽每个针对所述代理系统的外部中断,以启动所述代理系统;

[0026] 通过所述虚拟机监视器对启动后的所述代理系统的代理虚拟处理器和所述秘密工作区进行管理。

[0027] 在一些实施例,所述通过所述虚拟机监视器对启动后的所述代理系统的代理虚拟处理器和所述秘密工作区进行管理,包括:

[0028] 通过所述虚拟机监视器将启动后的所述代理系统的代理虚拟处理器下线,并解除所述秘密工作区;

[0029] 或者,

[0030] 通过所述虚拟机监视器将所述代理虚拟处理器调度到相应的逻辑处理器,并随机初始化所述秘密工作区。

[0031] 实现上述目的,本申请实施例的第二方面提出了一种虚拟机自省装置,所述装置包括:

[0032] 加载模块,用于将代理系统加载到可信域;其中,所述代理系统包括可信代理组件和加载器,所述可信域还包括虚拟机的操作系统内核,所述虚拟机运行在所述可信域内,所述虚拟机处于安全仲裁模式;

[0033] 初始化模块,用于通过所述加载器和虚拟机监视器初始化所述代理系统的运行环境;其中,所述虚拟机监视器处于非安全仲裁模式;

[0034] 确定模块,用于确定所述代理系统的运行模式;其中,所述运行模式为独占模式或者协作模式;

[0035] 管理模块,用于基于所述独占模式或者所述协作模式,通过所述虚拟机监视器对

所述可信域的虚拟处理器和内存资源进行管理,以使所述代理系统能够独立于所述操作系统内核运行。

[0036] 为实现上述目的,本申请实施例的第三方面提出了一种电子设备,所述电子设备包括存储器和处理器,所述存储器存储有计算机程序,所述处理器执行所述计算机程序时实现上述第一方面所述的方法。

[0037] 为实现上述目的,本申请实施例的第四方面提出了一种计算机可读存储介质,所述计算机可读存储介质存储有计算机程序,所述计算机程序被处理器执行时实现上述第一方面所述的方法。

[0038] 本申请实施例提出的虚拟机自省方法、虚拟机自省装置、电子设备及计算机可读存储介质,通过将代理系统加载到可信域,可信域还包括操作系统内核,以在不信任操作系统内核的前提下,在可信域创建一个安全的可信执行通道。代理系统包括可信代理组件和加载器,以确保可信执行通道的轻量化。可信域运行于虚拟机,虚拟机处于安全仲裁模式,从而为可信域提供一个高度安全的执行环境。为确保代理系统能够启动并正常运行,通过加载器和虚拟机监视器初始化代理系统的运行环境。虚拟机监视器运行于非安全仲裁模式,可以实现更高的性能和灵活性。确定代理系统的运行模式,运行模式为独占模式或者协作模式,基于独占模式或者协作模式,通过虚拟机监视器对可信域的虚拟处理器和内存资源进行管理,以使代理系统能够独立于操作系统内核运行。通过利用虚拟机监视器在平台层的资源管理能力,在不中断隔离保障的前提下,为可信域所有者提供一个安全、轻量、与虚拟机内核隔离的可信执行通道。

附图说明

[0039] 图1是本申请实施例提供的虚拟机自省方法的流程图;

[0040] 图2是图1中的步骤S120的流程图;

[0041] 图3是本申请实施例提供的虚拟机自省方法的架构图;

[0042] 图4是图1中的步骤S140的流程图;

[0043] 图5是图4中的步骤S420的流程图;

[0044] 图6是图4中的步骤S420的另一流程图;

[0045] 图7是本申请实施例提供的虚拟机自省方法的示意图;

[0046] 图8是图1中的步骤S140的另一流程图;

[0047] 图9是图8中的步骤S840的流程图;

[0048] 图10是本申请实施例提供的虚拟机自省方法的另一示意图;

[0049] 图11是本申请实施例提供的虚拟机自省装置的结构示意图;

[0050] 图12是本申请实施例提供的电子设备的硬件结构示意图。

具体实施方式

[0051] 为了使本申请的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本申请进行进一步详细说明。应当理解,此处所描述的具体实施例仅用以解释本申请,并不用于限定本申请。

[0052] 需要说明的是,虽然在装置示意图中进行了功能模块划分,在流程图中示出了逻

辑顺序,但是在某些情况下,可以以不同于装置中的模块划分,或流程图中的顺序执行所示出或描述的步骤。说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。

[0053] 除非另有定义,本文所使用的所有的技术和科学术语与属于本申请的技术领域的技术人员通常理解的含义相同。本文中所使用的术语只是为了描述本申请实施例的目的,不是旨在限制本申请。

[0054] 随着云计算的发展,越来越多的用户将敏感业务迁移至公有云平台,这带来了对虚拟机运行时安全性的防护需求,尤其是对主机操作系统(即虚拟机监视器)潜在恶意行为的防护。相关技术中,通过引入多种硬件级机密计算技术进行安全防护,主要包括可信域扩展、安全加密虚拟化-安全嵌套分页以及机密计算架构等。

[0055] 以可信域扩展为例,其通过引入可信域的概念,实现了对虚拟机私有内存和虚拟处理器状态的加密保护。该机制使用硬件级的“多密钥-全内存加密“技术对虚拟机私有内存进行加密,同时将虚拟机处理器的执行环境划分为安全仲裁模式与非安全仲裁模式,从而阻止虚拟机监视器访问可信域的明文内存数据与上下文。这些硬件机制本质上旨在保护虚拟机免受宿主机的恶意访问,因此禁止了传统的虚拟机外部自省工具对机密虚拟机内部状态的访问。然而,这种设计也带来了新的挑战,当虚拟机内部操作系统(即可信域内部的客户操作系统内核)自身被攻击或崩溃时,可信域所有者无法从内外部进行有效响应和修复,如执行内存抓取、日志提取等操作。可信域用户在面对内部攻击者时将失去对其虚拟机的控制能力。

[0056] 针对上述问题,相关技术采用特权分层方法在不信任虚拟机内核的前提下实现自省功能。在安全加密虚拟化-安全嵌套分页等架构中,硬件提供了虚拟机特权级别功能,允许在同一个机密虚拟机中运行多个具有不同权限等级的软件。通过将自省代理部署在比虚拟机内核更高权限的层,可以安全地访问内核空间的数据,实现对被攻击虚拟机的监测和控制。除设置虚拟机权限级别外,还有其他基于特权分层的实现,例如在机密虚拟机内嵌入飞地,作为可信子程序执行自省任务;又如将安全日志系统、虚拟可信平台模块封装至最高权限层,以增强可信性和抗篡改能力。

[0057] 然而,特权分层方法必须信任最高层或等效的特权软件模块,一旦特权软件模块所在的最高特权层遭受攻击,整个虚拟机的安全性将完全丧失。为支持多样化的安全功能,最高特权层常被引入大量代码,导致信任基增大,易受攻击面增加,安全性下降。且并非所有云用户都需要为其机密虚拟机增加额外的高权限层次,部署与运维特权层软件增加了系统复杂度和用户门槛,限制了实际应用场景。

[0058] 基于此,本申请实施例提供了一种虚拟机自省方法、虚拟机自省装置、电子设备及计算机可读存储介质,在不信任虚拟机内核的前提下,提供一种通用机制进行虚拟机控制,通过利用虚拟机监视器在云平台层的资源管理能力,如内存、虚拟处理器分配等,能够在不中断隔离保障的前提下,为可信域所有者提供一个安全、轻量、与虚拟机内核隔离的可信执行通道,解决了机密虚拟机内部特权分层机制所带来的问题。

[0059] 本申请实施例提供的虚拟机自省方法、虚拟机自省装置、电子设备及计算机可读存储介质,具体通过如下实施例进行说明,首先描述本申请实施例中的虚拟机自省方法。

[0060] 本申请实施例提供的虚拟机自省方法,涉及计算机技术领域。本申请实施例提供

的虚拟机自省方法可应用于终端中,也可应用于服务器端中,还可以是运行于终端或服务器端中的软件。在一些实施例中,终端可以是智能手机、平板电脑、笔记本电脑、台式计算机等;服务器端可以配置成独立的物理服务器,也可以配置成多个物理服务器构成的服务器集群或者分布式系统,还可以配置成提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、CDN以及大数据和人工智能平台等基础云计算服务的云服务器;软件可以是实现虚拟机自省方法的应用等,但并不局限于以上形式。

[0061] 本申请可用于众多通用或专用的计算机系统环境或配置中。例如:个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器系统、基于微处理器的系统、置顶盒、可编程的消费电子设备、网络PC、小型计算机、大型计算机、包括以上任何系统或设备的分布式计算环境等等。本申请可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本申请,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[0062] 图1是本申请实施例提供的虚拟机自省方法的一个可选的流程图,图1中的方法可以包括但不限于包括步骤S110至步骤S140。

[0063] 步骤S110,将代理系统加载到可信域;其中,代理系统包括可信代理组件和加载器,可信域还包括虚拟机的操作系统内核,虚拟机运行在可信域内,虚拟机处于安全仲裁模式;

[0064] 步骤S120,通过加载器和虚拟机监视器初始化代理系统的运行环境;其中,虚拟机监视器处于非安全仲裁模式;

[0065] 步骤S130,确定代理系统的运行模式;其中,运行模式为独占模式或者协作模式;

[0066] 步骤S140,基于独占模式或者协作模式,通过虚拟机监视器对可信域的虚拟处理器和内存资源进行管理,以使代理系统能够独立于操作系统内核运行。

[0067] 在一些实施例的步骤S110中,可信域是指在计算机系统如云平台中,具有高安全性和可信度的区域或者环境。将代理系统加载到可信域,可信域中还包含虚拟机的操作系统内核和各种应用程序,该虚拟机为机密虚拟机,机密虚拟机中的数据和应用程序在运行、静止和传输过程中始终保持加密状态。机密虚拟机运行在可信域内,通过可信域为机密虚拟机提供安全的执行环境。操作系统内核为操作系统的核心组件,用于管理虚拟机内部的虚拟资源,并为运行在其上的应用程序提供运行环境。虚拟机处于安全仲裁模式,在安全仲裁模式下,虚拟机的内存和处理器状态会受到保护,通过安全仲裁模式可确保虚拟机的机密性、完整性和可用性,以处理敏感数据和关键业务。当机密虚拟机内的操作系统内核遭遇攻击或者崩溃时,可信域所有者无法从内部或者外部对虚拟机进行响应和修复,可信域所有者将失去对虚拟机的控制能力。本申请实施例在不信任虚拟机操作系统内核的前提下,通过资源分离在可信域内部建立一个自治可信子系统即代理系统,来实现自省功能。

[0068] 可部署多个代理系统,每个代理系统均与操作系统内核一同内置于可信域镜像中。每个代理系统均包括可信代理组件和加载器,可信代理组件与加载器一起编译、链接,可信代理组件为可信域所有者在不信任操作系统内核的前提下运行的独立可执行程序,其功能和权限由可信域所有者在部署前预先选择和确定。假设在服务器运行一个敏感的财务

应用程序,该财务应用程序用于处理交易记录、客户信息等机密数据。然而,用户对服务器的操作系统内核的安全性存在疑虑,为了确保财务应用程序的安全运行,可部署可信代理组件。可信代理组件可具有数据加密、访问控制、日志记录等功能,并具有数据加密权限、数据访问权限等。加载器运行在特权级0,特权级0为最高特权级,用于向可信代理组件提供代理请求转发服务和运行时关键服务,代理请求转发服务指的是加载器代表可信代理组件向控制器发起操作请求的服务,如让代理系统进入休眠状态。运行时关键服务指的是代理系统运行过程中加载器提供的一系列关键服务,如准备代理系统的页表层次结构、在代理系统存在异常时保证优雅中止等。

[0069] 可信域具有物理地址空间,物理地址空间为可信域实际的、物理的内存地址范围如0-4GB,它与虚拟地址空间相对应,是操作系统和硬件直接管理内存资源的基础。

[0070] 请参阅图2,在一些实施例中,步骤S120可以包括但不限于包括步骤S210至步骤S230:

[0071] 步骤S210,通过加载器向虚拟机监视器发送初始化请求;

[0072] 步骤S220,基于初始化请求,通过虚拟机监视器从物理地址空间确定代理系统的内存区域;

[0073] 步骤S230,通过加载器根据内存区域的区域信息进行页表构建,得到可信域页表,并通过预设的可信域扩展模块为可信域分配安全扩展页表,以初始化代理系统的运行环境;其中,代理系统和操作系统内核共享安全扩展页表。

[0074] 在一些实施例的步骤S210中,请参阅图3,图3展示了一个支持本申请实施例的架构视图,系统架构包括带有控制器的虚拟机监视器、可信域扩展模块以及在可信域内安装并与操作系统内核隔离的两个代理系统,这两个代理系统均包括可信代理组件(代理1、代理2)和加载器。可信域和可信域扩展模块均处于安全仲裁模式,可信域扩展模块用于创建和管理可信域。虚拟机监视器是创建和管理虚拟机的软件层,并允许多个操作系统在同一物理硬件上运行,虚拟机监视器需要直接与硬件交互,为了提高虚拟机监视器的控制灵活性,将虚拟机监视器运行在非安全仲裁模式下。

[0075] 代理系统的引导启动分为两个阶段,其中第一个阶段为内核加载阶段,第二个阶段为运行环境初始化阶段。在可信域创建的内核加载阶段,将代理系统的镜像(包括加载器和可信代理组件)作为内核模块加载到可信域的私有内存中。随后加载器在控制器的协助下,初始化代理系统的运行时环境。代理系统的引导启动不会破坏现有的可信域认证流程,只需操作系统内核在代理系统加载时对代理系统的镜像做进一步度量,可将该镜像纳入度量链,并将度量结果存储在运行时度量寄存器中用于远程认证。度量结果用于指示已加载代理系统的镜像完整性。

[0076] 通过操作系统内核对代理系统进行完整性验证,若完整性验证通过,则通过加载器向虚拟机监视器发送初始化请求,初始化请求用于请求初始化代理系统的运行时环境。若完整性验证未通过,则不进行后续流程,并重新获取完整的代理系统。

[0077] 在一些实施例的步骤S220中,虚拟机监视器接收初始化请求,响应于初始化请求,虚拟机监视器的控制器从可信域的物理地址空间中为代理系统选定一段内存区域作为代理工作区,得到代理系统的内存区域。

[0078] 在一些实施例的步骤S230中,内存区域的区域信息为内存区域在物理地址空间的

物理地址,可称为驻留基址。虚拟机监视器的控制器将内存区域的区域信息传递给加载器。为使代理系统能够脱离可信域的操作系统内核独立执行,加载器使用与操作系统内核不共享的资源来搭建代理系统的运行环境,除了可信代理组件及加载器自身使用的内存页外,该运行环境还包括完整的分页层次结构以及一系列系统数据结构,如中断描述表和全局描述表。可通过加载器根据区域信息构建四级分页层次,将可信代理组件和加载器的代码(含中断/异常处理程序)、数据和栈等映射到所选定内存区域的物理地址,以建立虚拟地址和物理地址之间的映射关系,得到可信域页表。

[0079] 通过可信域扩展模块为可信域分配安全扩展页表,可信域扩展模块为每个可信域只分配一个安全扩展页表,操作系统内核、应用程序和代理系统共享该安全扩展页表,页表一旦就绪,运行环境初始化流程结束,加载器立即切换控制器生效,使得控制器可以控制和管理代理系统的运行。

[0080] 通过上述步骤S210至步骤S230,能够初始化代理系统的运行环境,使代理系统能够启动并运行在安全环境。

[0081] 在一些实施例的步骤S130中,确定代理系统的运行模式,运行模式为独占模式或者协作模式,在独占模式下仅存在代理线程独占式访问代理系统,在协作模式下可信域线程和代理线程能够并行运行。代理线程为代理系统创建的线程,可信域线程为操作系统内核或者应用程序创建的线程。

[0082] 本申请实施例利用虚拟机监视器进行资源隔离实现可信系统,并基于可信系统在机密虚拟机实现可信自省。基于独占模式或者协作模式,利用虚拟机监视器对虚拟机运行时资源(如虚拟处理器、物理地址映射)具有唯一控制权的特性,通过虚拟机监视器的控制器对可信域的虚拟处理器和内存资源进行管理,将一部分虚拟处理器和内存资源从被保护的虚拟机中逻辑隔离出来,以使代理系统能够独立于操作系统内核运行,由此构建一个与虚拟机的操作系统内核完全隔离的可信执行系统。该系统在虚拟机内部,但不受操作系统内核的控制,从而能在内核被攻陷的情况下独立执行敏感任务(如自省或响应操作)。从系统角度看,代理(可信代理组件)拥有一个专属可信系统,每个代理系统均具有一个专用的虚拟处理器和一组映射到一段可信域物理地址空间的私有物理页。除非专门设计,否则该代理的虚拟处理器与可信域物理地址页不与其他代理共享。

[0083] 请参阅图4,在一些实施例中,步骤S140可以包括但不限于包括步骤S410至步骤S420:

[0084] 步骤S410,基于独占模式,通过加载器向虚拟机监视器发送目标请求;

[0085] 步骤S420,基于目标请求,通过虚拟机监视器对虚拟处理器和内存资源进行管理。

[0086] 在一些实施例的步骤S410中,基于独占模式,在可信域启动过程中,代理系统被初始化并置于休眠状态,可通过加载器向虚拟机监视器发送目标请求,此时目标请求用于请求虚拟机监视器将可信代理组件设置为休眠状态。代理系统接收到任务请求后,可信代理组件被激活进入运行状态,可通过加载器向虚拟机监视器发送目标请求,此时目标请求用于请求虚拟机监视器将可信代理组件设置为激活状态。当代理系统完成工作负载后,即完成任务请求所指示的任务后,代理系统会回到休眠状态。可通过加载器发送目标请求将代理系统设置为休眠状态。

[0087] 需要说明的是,虽然代理系统的工作可能需要访问操作系统内核或应用程序的应

用数据,但代理系统的控制流完全限于自身的可信域物理地址区域内,不会执行该区域之外的指令。

[0088] 需要进一步说明的是,加载器通过虚拟机调用的方式向可信域扩展模块发送目标请求,并通过可信域扩展模块以安全仲裁调用的方式向虚拟机监视器的控制器转发该目标请求。

[0089] 在一些实施例的步骤S420中,虚拟机监视器的控制器在接收到来自加载器的目标请求,对可信域的虚拟处理器和内存资源进行管理。虚拟处理器包括代理系统的代理虚拟处理器和可信域内除代理系统之外的操作系统内核、应用程序等的其他虚拟处理器,内存资源为可信域的物理地址空间,包括代理系统的内存区域。

[0090] 通过上述步骤S410至步骤S420,控制器基于加载器发送的请求管理可信域的虚拟处理器和内存资源,以确保代理系统与操作系统内核之间可以相互独立运行,在可信域所有者不信任操作系统内核时,可通过代理系统实现任务的可信执行。

[0091] 请参阅图5,在一些实施例中,步骤S420可以包括但不限于包括步骤S510至步骤S520:

[0092] 步骤S510,若目标请求用于请求虚拟机监视器将可信代理组件设置为休眠状态,则通过虚拟机监视器对内存区域进行内存阻断;

[0093] 步骤S520,将代理虚拟处理器和承载代理虚拟处理器的逻辑处理器解绑,并将其他虚拟处理器调度到对应的逻辑处理器。

[0094] 在一些实施例的步骤S510中,若目标请求用于请求虚拟机监视器的控制器将可信代理组件设置为休眠状态,则控制器根据参数开启可信域物理地址的内存阻断,以对代理系统的内存区域进行内存阻断,并执行转译后备缓冲器跟踪以撤销之前用于代理系统的可信域物理地址映射,此时代理系统进入休眠状态。一个可信域具有一个安全扩展页,代理系统位于可信域内,因此可信域物理地址内存阻断对可信域与代理系统会产生相同的影响。

[0095] 在一些实施例的步骤S520中,将代理虚拟处理器和承载代理虚拟处理器的逻辑处理器解绑,并将其他虚拟处理器调度到各自对应的逻辑处理器。

[0096] 上述步骤S510至步骤S520,通过用户物理地址隔断技术阻断访问,保障了代理系统的静态安全性。当代理系统不处于活动状态时,虚拟机监视器对其内存施加物理地址限制,将其从可信域的有效地址空间中移除,使得具有高权限的操作系统内核也无法定位或访问该代理系统的内存区域,从而保障其静态安全性。

[0097] 请参阅图6,在一些实施例中,步骤S420还可以包括但不限于包括步骤S610至步骤S620:

[0098] 步骤S610,若目标请求用于请求虚拟机监视器将可信代理组件设置为激活状态,则通过虚拟机监视器对内存区域进行阻断解除;

[0099] 步骤S620,将其他虚拟处理器与对应的逻辑处理器解绑,并将代理虚拟处理器调度到目标逻辑处理器。

[0100] 在一些实施例的步骤S610中,若目标请求用于请求虚拟机监视器的控制器将可信代理组件设置为激活状态,则控制器通知可信域扩展模块解除对代理系统的内存区域(代理工作区)的阻断,使该可信域物理地址区域得以翻译到物理地址。

[0101] 在一些实施例的步骤S620中,控制器向可信域扩展模块发送处理器间中断指令暂

停可信域,可信域扩展模块基于处理器间中断指令将所有运行中的分配给该可信域的其他虚拟处理器从其承载的逻辑处理器上撤下,即将其他虚拟处理器与对应的逻辑处理器解绑,生成可信域退出指令。可信域扩展模块基于可信域退出指令将逻辑处理器的控制权交给控制器。控制器发起一次安全仲裁调用,使可信域扩展模块将代理虚拟处理器重新调度到目标逻辑处理器,以恢复代理虚拟处理器的运行。

[0102] 需要说明的是,虚拟处理器一旦被调度运行,就会从先前发出虚拟机调用的指令之后继续执行加载器的代码,加载器随即激活代理执行环境,并将控制权交给代理系统。

[0103] 上述步骤S610至步骤S620,通过虚拟处理器重调度实现代理系统在运行期间的动态隔离。在代理系统激活运行时,虚拟机监视器会通过调度控制暂停可信域所有其他线程,尤其是被攻击的内核线程,确保在该期间仅代理系统的代理虚拟处理器被调度执行,防止不可信代码干扰其运行过程,从而实现运行时的动态安全隔离。

[0104] 请参阅图7,图7为独占模式下的执行流程图。初始化代理系统的运行环境,在完成系统环境初始化后,通过控制器将代理虚拟处理器和承载代理虚拟处理器的逻辑处理器解绑,以将代理系统置于休眠状态,并通过操作系统内核控制其他虚拟处理器的运行。当有激活请求发往虚拟机监视器时,控制器会唤醒处于休眠状态的代理系统,并暂停其他所有可信域线程。代理系统完成任务后,加载器通过另一次虚拟机调用将代理系统置于休眠状态,控制器将可信域的其他虚拟处理器调度回各自的逻辑处理器。

[0105] 代理系统将其所有计算资源(即代理虚拟处理器和内存区域)与可信域的系统软件严格隔离,从而为代理系统的执行提供安全环境。该思想是基于虚拟机监视器对可信域的虚拟处理器和内存资源的管理来实现的,当代理系统需要休眠时,将其物理地址内存阻断,禁止可信域线程访问;当代理系统需要激活并解除内存阻断时,则通过虚拟处理器调度,暂停所有可信域线程的执行,使代理系统的内存区域和虚拟处理器上下文与其他任何可信域线程安全隔离,仅存在代理线程独占式访问代理系统。

[0106] 请参阅图8,在一些实施例中,步骤S140可以包括但不限于包括步骤S810至步骤S840:

[0107] 步骤S810,基于协作模式,通过虚拟机监视器从物理地址空间中确定代理系统的秘密工作区;

[0108] 步骤S820,根据秘密工作区更新安全扩展页表;

[0109] 步骤S830,通过加载器接收更新后的安全扩展页表,并屏蔽每个针对代理系统的外部中断,以启动代理系统;

[0110] 步骤S840,通过虚拟机监视器对启动后的代理系统的代理虚拟处理器和秘密工作区进行管理。

[0111] 在一些实施例的步骤S810中,在同一个可信域中可并行部署一个或多个代理系统,每个代理系统可在独占模式或者协作模式下运行。独占模式下,代理系统运行期间所有可信域线程被暂停,适用于可信域自省任务,此时冻结的可信域有助于保证采集数据的一致性。协作模式下,代理系统与其他不受信任的可信域线程可同时运行,并可通过数据交换与其他可信域线程交互,适用于诸如使用长期密钥进行数据签名等安全敏感任务。

[0112] 为减少代理系统对可信域正常运行的性能影响,本申请实施例用位置隐藏替代内存阻断,使可信域线程与代理线程能够并行运行。具体地,虚拟机监视器的控制器在可信域

的物理地址空间中,随机选择一段不在可信域常规范围内的、不公开的可信域物理地址作为代理系统的内存映射位置,从而为代理系统分配一个随机且保密的物理地址,得到代理系统的秘密工作区。

[0113] 由于可信域的物理地址阻断会对可信域中的所有虚拟处理器(包括代理虚拟处理器)生效,因此无法用于保护正在运行的代理系统,本申请实施例基于协作模式让代理系统在一个秘密工作区中运行。可信域攻击者必须先正确猜中该秘密工作区的可信域物理地址才能进行任何访问,鉴于广阔的52位可信域物理地址空间,其成功猜中的概率几乎可以忽略不计。通过物理地址空间随机化实现内存位置隐藏以支持并发运行,即使代理系统与操作系统内核并发运行,攻击者也难以定位并访问该代理系统,从而以更低开销实现安全隔离。

[0114] 更重要的是,任何错误访问都会立即触发可信域退出并被虚拟监视器捕获。为了主动保护代理系统,该可信域物理地址区域在检测到对可信域分配范围外的扩展页表违规时,会重新随机化。因此,协作模式下的代理系统不会遵循独占模式下固定的运行-休眠周期,其更像一个常驻守护线程。

[0115] 在一些实施例的步骤S820中,类似于独占模式,控制器通过安全仲裁调用向可信域扩展模块增加秘密工作区对应的页表项,根据该页表项更新安全扩展页表。

[0116] 在一些实施例的步骤S830中,通过加载器显式接收更新后的安全扩展页表,使该安全扩展页表指示的物理地址与虚拟地址之间新的映射关系生效。由于代理线程与信任域线程并发执行,因此需要防止信任域线程中断代理虚拟处理器。为此加载器会屏蔽所有外部中断,并在其专用的中断/异常处理程序中丢弃来自可信域线程的不可屏蔽处理期间中断,并屏蔽每个针对代理系统的外部中断,以启动代理系统。

[0117] 在一些实施例的步骤S840中,通过虚拟机监视器的控制器对启动后的代理系统的代理虚拟处理器和秘密工作区进行管理,以使代理系统能够独立于操作系统内核运行。

[0118] 上述步骤S810至步骤S840,通过物理地址空间随机化实现内存位置隐藏,实现以更低开销实现代理系统与操作系统内核的安全隔离。

[0119] 请参阅图9,在一些实施例,步骤S840可以包括但不限于包括步骤S910或者步骤S920:

[0120] 步骤S910,通过虚拟机监视器将启动后的代理系统的代理虚拟处理器下线,并解除秘密工作区;

[0121] 步骤S920,通过虚拟机监视器将代理虚拟处理器调度到相应的逻辑处理器,并随机初始化秘密工作区。

[0122] 在一些实施例的步骤S910中,代理系统启动完成后,加载器通过虚拟机调用向控制器发送休眠请求。基于休眠请求,通过控制器调度代理虚拟处理器下线,由可信域扩展模块保存上下文,并清除秘密工作区。需要说明的是,协作模式场景下,控制器不会进行可信域物理地址阻断。

[0123] 在一些实施例的步骤S920中,激活请求由可信域线程发起,通过可信域线程所在的逻辑处理器即其他虚拟处理器对应的逻辑处理器发起虚拟机调用,通知控制器唤醒处理休眠状态的代理系统,控制器在该逻辑处理器上立即返回给发起激活请求的可信域线程,并在另一逻辑处理器上唤醒代理系统。可通过控制器将代理虚拟处理器调度到该另一逻辑

处理器,并随机初始化秘密工作区,以重新确定代理系统的秘密工作区。

[0124] 通过上述步骤S910至步骤S920,能够实现代理系统和操作系统内核的并行运行。

[0125] 请参阅图10,图10为协作模式下的执行流程图。通过虚拟处理器0初始化代理系统的运行环境,并通过虚拟处理器0、虚拟处理器1和其他虚拟处理器并行运行内核代码。虚拟处理器0通知控制器唤醒休眠的代理系统,控制器立即返回给虚拟处理器0,并通过虚拟处理器1将代理系统重载入秘密工作区,通过虚拟器处理器1运行代理系统,并将运行结果写回到代理系统的驻留基址,并清除秘密工作区。

[0126] 本申请实施例在不修改可信域内核、不依赖特权分层的前提下实现自省与维护功能。相较于现有虚拟机特权分级或飞地等方案需引入额外高权限内核组件,本申请实施例无需修改虚拟机操作系统、无需虚拟机内嵌特权服务,即可实现与内核隔离的可信执行环境,能够简化部署并增强通用性,具有更强的安全性保障。

[0127] 传统的特权分层方案依赖于将安全代理部署在虚拟机中最具特权的软件中,一旦该层被攻击者控制,则所有安全功能将完全失效。本申请实施例通过虚拟机监视器进行资源分离,实现可信代理系统对可信域内核的完全物理隔离。即使可信域内核已被完全攻陷,攻击者仍无法访问可信代理系统的内存或虚拟处理器,从而有效对抗内核层面的内部威胁。此外,本申请实施例严格遵循最小特权原则,不提升任何软件特权水平,也不修改可信域的操作系统,降低了潜在的系统脆弱面。

[0128] 与需要更改可信域内部操作系统结构或插入特权服务的方案不同,本发明无需修改可信域的任何系统软件,即可构建可信执行环境。这种“非侵入式”的设计大大简化了部署过程,增强了技术的通用性和可推广性。虽然该方案需要云服务商在虚拟机监视器层配合部署,但这种合作模式已日益普遍,并不构成推广的主要障碍。

[0129] 特权分层方法由于功能集中在最上层权限代码中,限制了安全代理的复杂性和数量。本申请实施例将可信代理系统解耦于可信域内核之外,支持并发部署多个、复杂度各异的安全代理任务。此外,特权分层通常引入持久性性能开销(如高特权级别上的长期驻留任务),而本申请实施例的性能代价仅在启动/激活可信代理系统时出现,长期运行成本更低。

[0130] 综上所述,本申请实施例在保持较高安全强度的同时,兼顾了部署简易性、系统兼容性和功能可扩展性,能够有效弥补现有技术在可信虚拟机自省中的局限性。

[0131] 请参阅图11,本申请实施例还提供一种虚拟机自省装置,可以实现上述虚拟机自省方法,该虚拟机自省装置包括:

[0132] 加载模块1110,用于将代理系统加载到可信域;其中,代理系统包括可信代理组件和加载器,可信域还包括虚拟机的操作系统内核,虚拟机运行在可信域内,虚拟机处于安全仲裁模式;

[0133] 初始化模块1120,用于通过加载器和虚拟机监视器初始化代理系统的运行环境;其中,虚拟机监视器处于非安全仲裁模式;

[0134] 确定模块1130,用于确定代理系统的运行模式;其中,运行模式为独占模式或者协作模式;

[0135] 管理模块1140,用于基于独占模式或者协作模式,通过虚拟机监视器对可信域的虚拟处理器和内存资源进行管理,以使代理系统能够独立于操作系统内核运行。

[0136] 该虚拟机自省装置的具体实施方式与上述虚拟机自省方法的具体实施例基本相

同,在此不再赘述。

[0137] 本申请实施例还提供了一种电子设备,电子设备包括存储器和处理器,存储器存储有计算机程序,处理器执行计算机程序时实现上述虚拟机自省方法。该电子设备可以为包括平板电脑、车载电脑等任意智能终端。

[0138] 请参阅图12,图12示意了另一实施例的电子设备的硬件结构,电子设备包括:

[0139] 处理器1210,可以采用通用的中央处理器(Central Processing Unit,CPU)、微处理器、应用专用集成电路(Application Specific Integrated Circuit,ASIC)、或者一个或多个集成电路等方式实现,用于执行相关程序,以实现本申请实施例所提供的技术方案;

[0140] 存储器1220,可以采用只读存储器(Read Only Memory,ROM)、静态存储设备、动态存储设备或者随机存取存储器(Random Access Memory,RAM)等形式实现。存储器1220可以存储操作系统和其他应用程序,在通过软件或者固件来实现本说明书实施例所提供的技术方案时,相关的程序代码保存在存储器1220中,并由处理器1210来调用执行本申请实施例的虚拟机自省方法;

[0141] 输入/输出接口1230,用于实现信息输入及输出;

[0142] 通信接口1240,用于实现本设备与其他设备的通信交互,可以通过有线方式(例如USB、网线等)实现通信,也可以通过无线方式(例如移动网络、WIFI、蓝牙等)实现通信;

[0143] 总线1250,在设备的各个组件(例如处理器1210、存储器1220、输入/输出接口1230和通信接口1240)之间传输信息;

[0144] 其中处理器1210、存储器1220、输入/输出接口1230和通信接口1240通过总线1250实现彼此之间在设备内部的通信连接。

[0145] 本申请实施例还提供了一种计算机可读存储介质,该计算机可读存储介质存储有计算机程序,该计算机程序被处理器执行时实现上述虚拟机自省方法。

[0146] 存储器作为一种非暂态计算机可读存储介质,可用于存储非暂态软件程序以及非暂态性计算机可执行程序。此外,存储器可以包括高速随机存取存储器,还可以包括非暂态存储器,例如至少一个磁盘存储器件、闪存器件、或其他非暂态固态存储器件。在一些实施方式中,存储器可选包括相对于处理器远程设置的存储器,这些远程存储器可以通过网络连接至该处理器。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

[0147] 本申请实施例描述的实施例是为了更加清楚的说明本申请实施例的技术方案,并不构成对于本申请实施例提供的技术方案的限定,本领域技术人员可知,随着技术的演变和新应用场景的出现,本申请实施例提供的技术方案对于类似的技术问题,同样适用。

[0148] 本领域技术人员可以理解的是,图中示出的技术方案并不构成对本申请实施例的限定,可以包括比图示更多或更少的步骤,或者组合某些步骤,或者不同的步骤。

[0149] 以上所描述的装置实施例仅仅是示意性的,其中作为分离部件说明的单元可以是或者也可以不是物理上分开的,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。

[0150] 本领域普通技术人员可以理解,上文中所公开方法中的全部或某些步骤、系统、设备中的功能模块/单元可以被实施为软件、固件、硬件及其适当的组合。

[0151] 本申请的说明书及上述附图中的术语“第一”、“第二”、“第三”、“第四”等(如果存

在)是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的本申请的实施例能够以除了在这里图示或描述的那些以外的顺序实施。此外,术语“包括”和“具有”以及他们的任何变形,意图在于覆盖不排他的包含,例如,包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元,而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0152] 应当理解,在本申请中,“至少一个(项)”是指一个或者多个,“多个”是指两个或两个以上。“和/或”,用于描述关联对象的关联关系,表示可以存在三种关系,例如,“A和/或B”可以表示:只存在A,只存在B以及同时存在A和B三种情况,其中A,B可以是单数或者复数。字符“/”一般表示前后关联对象是一种“或”的关系。“以下至少一项(个)”或其类似表达,是指这些项中的任意组合,包括单项(个)或复数项(个)的任意组合。例如,a,b或c中的至少一项(个),可以表示:a,b,c,“a和b”,“a和c”,“b和c”,或“a和b和c”,其中a,b,c可以是单个,也可以是多个。

[0153] 在本申请所提供的几个实施例中,应该理解到,所揭露的装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,上述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0154] 上述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0155] 另外,在本申请各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0156] 集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时,可以存储在一个计算机可读取存储介质中。基于这样的理解,本申请的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括多指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本申请各个实施例的方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(Read-Only Memory,简称ROM)、随机存取存储器(Random Access Memory,简称RAM)、磁碟或者光盘等各种可以存储程序的介质。

[0157] 以上参照附图说明了本申请实施例的优选实施例,并非因此局限本申请实施例的权利范围。本领域技术人员不脱离本申请实施例的范围和实质内所作的任何修改、等同替换和改进,均应在本申请实施例的权利范围之内。

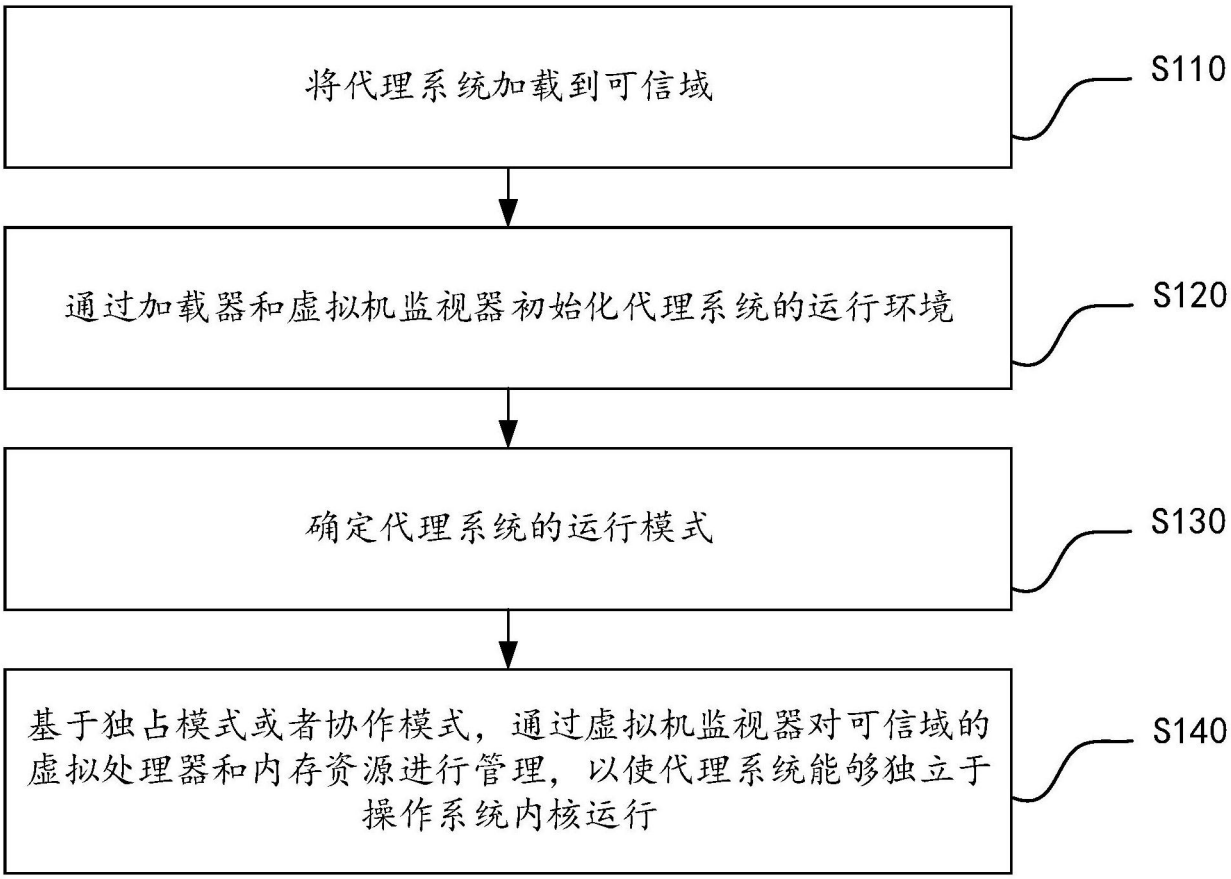

将代理系统加载到可信域
S110
通过加载器和虚拟机监视器初始化代理系统的运行环境
S120
确定代理系统的运行模式
S130
基于独占模式或者协作模式，通过虚拟机监视器对可信域的虚拟处理器和内存资源进行管理，以使代理系统能够独立于操作系统内核运行
S140

图1

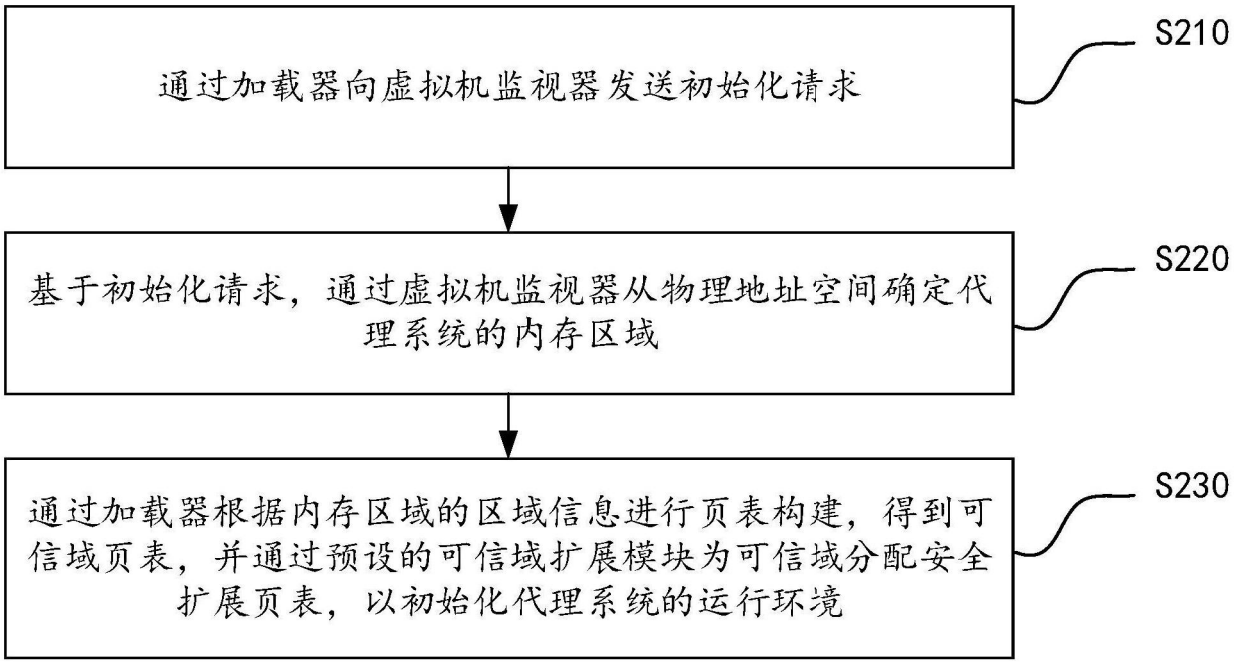

通过加载器向虚拟机监视器发送初始化请求
S210
基于初始化请求，通过虚拟机监视器从物理地址空间确定代理系统的内存区域
S220
通过加载器根据内存区域的区域信息进行页表构建，得到可信域页表，并通过预设的可信域扩展模块为可信域分配安全扩展页表，以初始化代理系统的运行环境
S230

图2

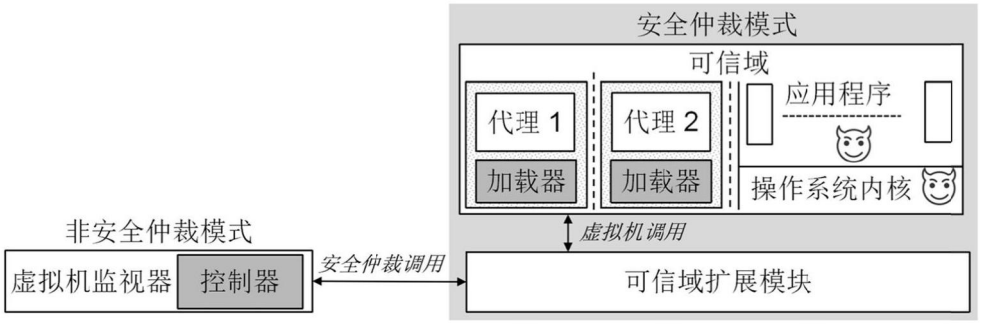

安全仲裁模式
可信域
代理 1
代理 2
应用程序
加载器
加载器
操作系统内核
虚拟机调用
非安全仲裁模式
虚拟机监视器
控制器
安全仲裁调用
可信域扩展模块

图3

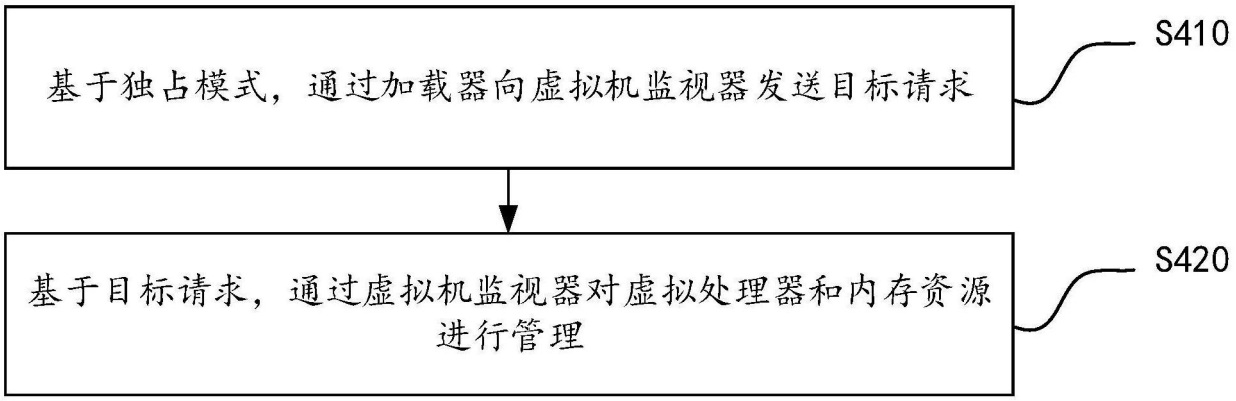

基于独占模式，通过加载器向虚拟机监视器发送目标请求
S410
基于目标请求，通过虚拟机监视器对虚拟处理器和内存资源进行管理
S420

图4

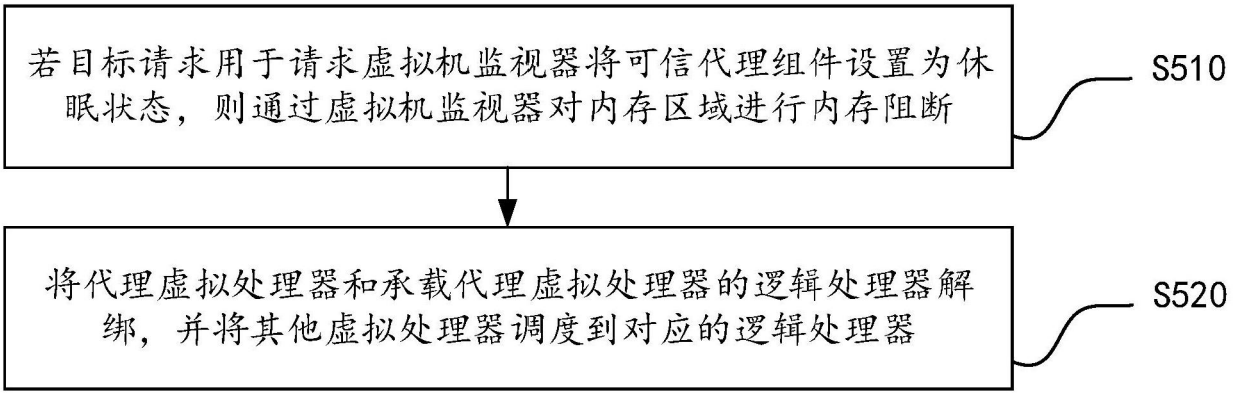

若目标请求用于请求虚拟机监视器将可信代理组件设置为休眠状态，则通过虚拟机监视器对内存区域进行内存阻断
S510
将代理虚拟处理器和承载代理虚拟处理器的逻辑处理器解绑，并将其他虚拟处理器调度到对应的逻辑处理器
S520

图5

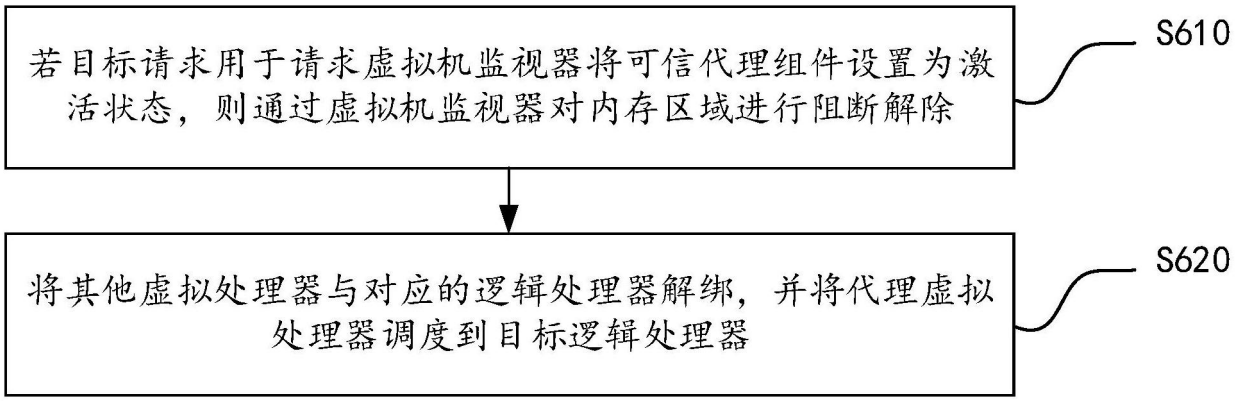

若目标请求用于请求虚拟机监视器将可信代理组件设置为激活状态，则通过虚拟机监视器对内存区域进行阻断解除
S610
将其他虚拟处理器与对应的逻辑处理器解绑，并将代理虚拟处理器调度到目标逻辑处理器
S620

图6

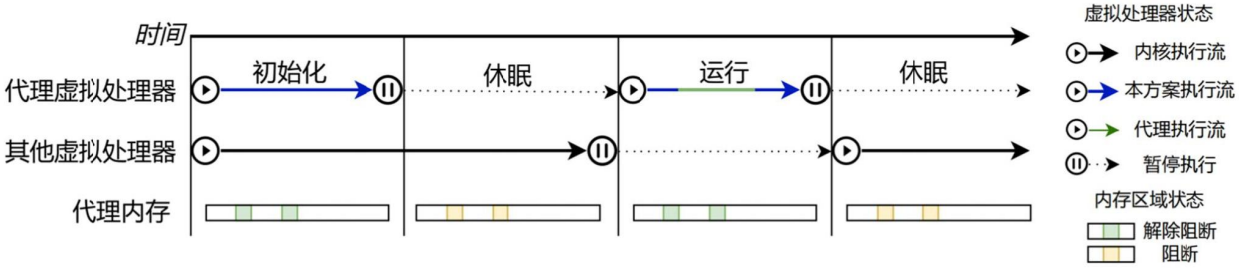

时间
代理虚拟处理器
其他虚拟处理器
代理内存
初始化
休眠
运行
休眠
虚拟处理器状态
内核执行流
本方案执行流
代理执行流
暂停执行
内存区域状态
解除阻断
阻断

图7

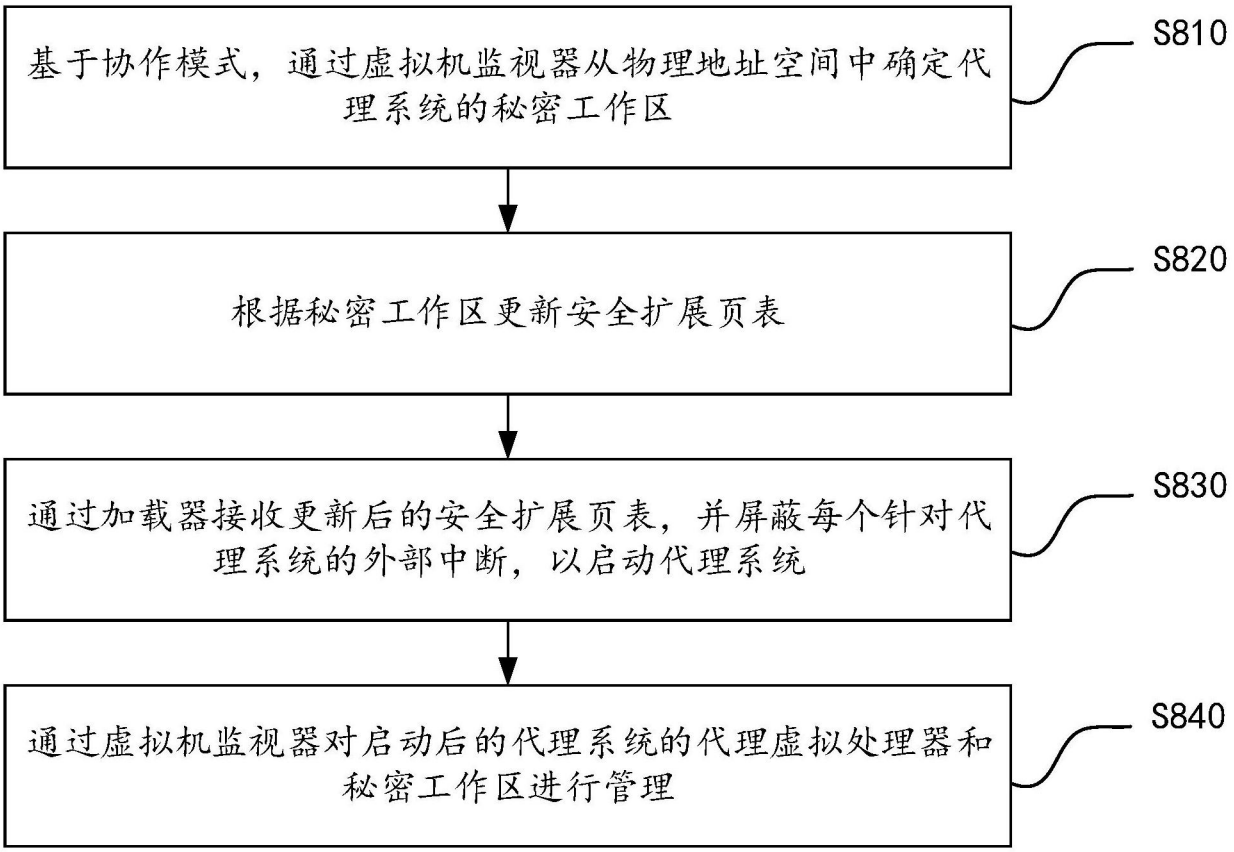

基于协作模式，通过虚拟机监视器从物理地址空间中确定代理系统的秘密工作区
S810
根据秘密工作区更新安全扩展页表
S820
通过加载器接收更新后的安全扩展页表，并屏蔽每个针对代理系统的外部中断，以启动代理系统
S830
通过虚拟机监视器对启动后的代理系统的代理虚拟处理器和秘密工作区进行管理
S840

图8

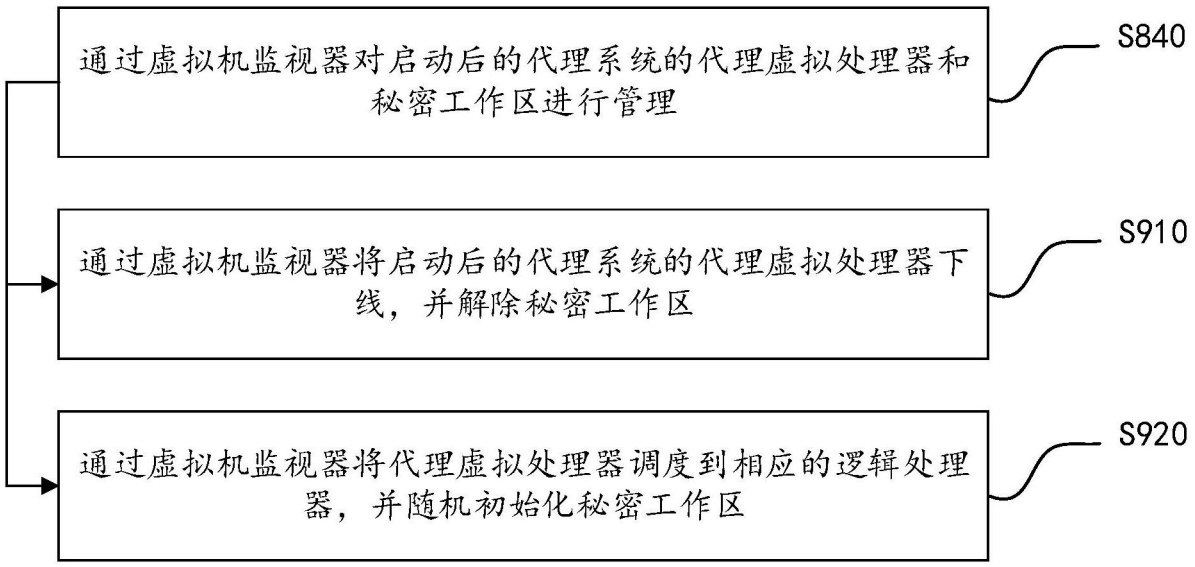

通过虚拟机监视器对启动后的代理系统的代理虚拟处理器和秘密工作区进行管理
S840
通过虚拟机监视器将启动后的代理系统的代理虚拟处理器下线，并解除秘密工作区
S910
通过虚拟机监视器将代理虚拟处理器调度到相应的逻辑处理器，并随机初始化秘密工作区
S920

图9

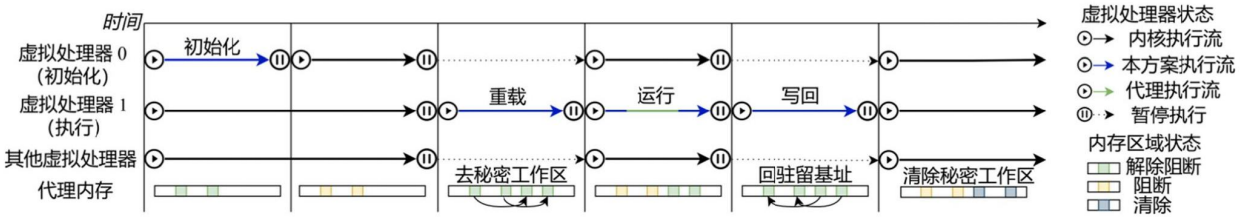

时间
虚拟处理器 0
(初始化)
虚拟处理器 1
(执行)
其他虚拟处理器
代理内存
初始化
重载
运行
写回
去秘密工作区
回驻留基址
清除秘密工作区
虚拟处理器状态
内核执行流
本方案执行流
代理执行流
暂停执行
内存区域状态
解除阻断
阻断
清除

图10

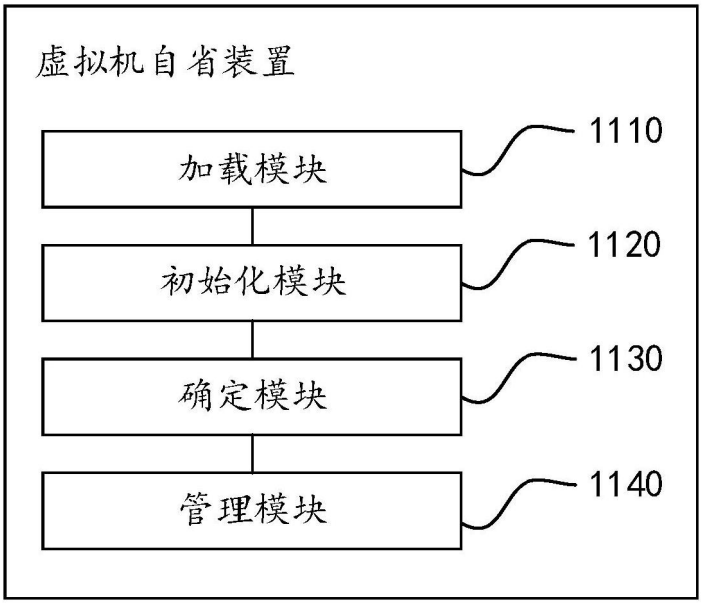

虚拟机自省装置
加载模块
1110
初始化模块
1120
确定模块
1130
管理模块
1140

图11

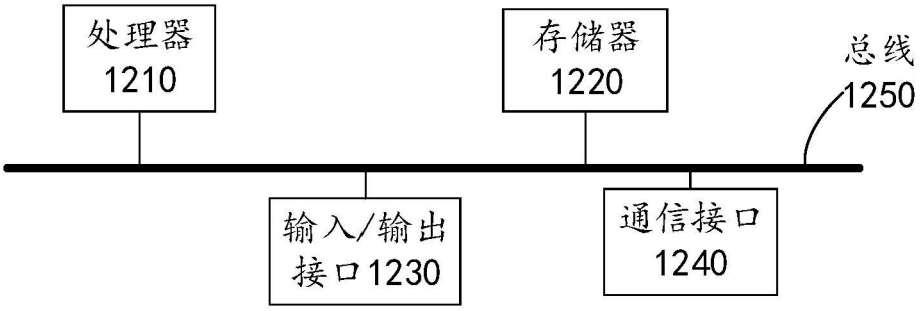

处理器
1210
存储器
1220
总线
1250
输入/输出
接口1230
通信接口
1240

图12