

(12) 发明专利申请

(10) 申请公布号 CN 102799817 A

(43) 申请公布日 2012. 11. 28

(21) 申请号 201210225723. 2

(22) 申请日 2012. 06. 29

(30) 优先权数据

13/174, 247 2011. 06. 30 US

(71) 申请人 卡巴斯基实验室封闭式股份公司

地址 俄罗斯莫斯科

(72) 发明人 维亚切斯拉夫 · E · 卢萨科夫

亚历山大 · V · 希里亚耶夫

(74) 专利代理机构 北京市磐华律师事务所

11336

代理人 徐丁峰 魏宁

(51) Int. Cl.

G06F 21/00 (2006. 01)

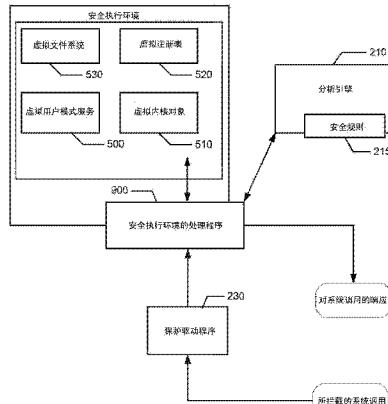
权利要求书 2 页 说明书 7 页 附图 10 页

(54) 发明名称

用于使用虚拟化技术进行恶意软件保护的系统和方法

(57) 摘要

本发明公开了用于使用虚拟化技术来保护部署于主机上的应用程序抵御恶意软件的系统、方法及计算机程序产品。一个示范性恶意软件保护系统可包括内核级驱动程序，其经配置以拦截寻址到受保护应用程序的对象的系统调用。该系统还包括分析引擎，其经配置以确定是否有与以下各项中的一个或多个相关联的安全规则：所拦截的系统调用、受保护应用程序的对象以及在受保护应用程序的对象上所允许的动作。安全规则指示是否允许或不允许所述系统调用在所述主机上执行。如果没有与系统调用相关联的安全规则，则使用受保护应用程序的对象的虚拟副本在主机的安全执行环境中执行系统调用。



1. 一种用于保护部署于主机上的应用程序的方法,所述方法包括 :

在所述主机的内核级,拦截寻址到部署于所述主机上受保护应用程序的对象的系统调用;

确定是否有与以下各项中的一个或多个相关联的安全规则:所拦截的系统调用、所述受保护应用程序的所述对象以及在所述受保护应用程序的所述对象上所允许的动作,其中,所述安全规则至少指示是否允许所述系统调用在所述主机上执行或者不允许所述系统调用在所述主机上执行;

如果有安全规则指示允许所述系统调用在所述主机上执行,则在所述主机上执行所述系统调用;

如果有安全规则指示不允许所述系统调用在所述主机上执行,则在所述主机上阻止所述系统调用的执行;

如果没有与所述系统调用相关联的安全规则,则使用所述受保护应用程序的所述对象的虚拟副本在安全执行环境中执行所述系统调用;

分析对所述受保护应用程序的所述对象的所述虚拟副本的改变是否表现出对所述应用程序、应用程序数据或所述主机的任何安全威胁;

如果对所述对象的所述虚拟副本的所述改变未表现出任何安全威胁,则对所述主机中的真实对象应用所述改变;以及

如果对所述对象的所述虚拟副本的所述改变表现出安全威胁,则在所述主机上阻止所述系统调用的执行。

2. 根据权利要求 1 所述的方法,其中,分析对所述对象的所述虚拟副本的改变是否表现出任何安全威胁包括:执行以下各项中的一个或多个:启发式分析、恶意软件签名匹配以及对所述对象的至少所述虚拟副本的行为分析。

3. 根据权利要求 1 所述的方法,其中,所述安全执行环境实施为所述主机上的用户模式服务。

4. 根据权利要求 1 所述的方法,其中,所述安全执行环境进一步包括:所述主机的一个或多个虚拟用户模式服务,一个或多个虚拟内核对象,虚拟注册表和虚拟文件系统。

5. 根据权利要求 1 所述的方法,其中,拦截系统调用进一步包括:通过内核级驱动程序拦截系统调用。

6. 根据权利要求 1 所述的方法,进一步包括:

拦截来自部署于所述主机上的应用程序的远程过程调用;以及

将被所述远程过程调用所寻址的对象的端口名称用在所述安全执行环境中的所述对象的虚拟副本的端口名称替换,从而将所述远程过程调用重定向到所述安全执行环境。

7. 一种用于保护部署于主机上的应用程序的系统,所述系统包括:

内核级驱动程序,存储在所述主机的存储器中并且由所述主机的处理器可执行,所述内核级驱动程序经配置以拦截寻址到受保护应用程序的对象的系统调用;和

分析引擎,由所述处理器可执行,所述分析引擎经配置以:

确定是否有与以下各项中的一个或多个相关联的安全规则:所拦截的系统调用、所述受保护应用程序的所述对象以及在所述受保护应用程序的所述对象上所允许的动作,其中,所述安全规则至少指示是否允许所述系统调用在所述主机上执行或者不允许所述系统

调用在所述主机上执行；

如果有安全规则指示允许所述系统调用在所述主机上执行，则指令所述主机执行所述系统调用；

如果有安全规则指示不允许所述系统调用在所述主机上执行，则指令所述主机阻止所述系统调用的执行；

如果没有与所述系统调用相关联的安全规则，则指令所述主机的安全执行环境的处理程序使用所述受保护应用程序的所述对象的虚拟副本在所述安全执行环境中执行所述系统调用；

分析对所述受保护应用程序的所述对象的所述虚拟副本的改变是否表现出对所述应用程序、应用程序数据、或所述主机的任何安全威胁；

如果对所述对象的所述虚拟副本的所述改变未表现出任何安全威胁，则指令所述主机对所述主机中的真实对象应用所述改变；以及

如果对所述对象的所述虚拟副本的所述改变表现出安全威胁，则指令所述主机在所述主机上阻止所述系统调用的执行。

8. 根据权利要求 7 所述的系统，其中，分析对所述对象的所述虚拟副本的改变是否表现出任何安全威胁包括：执行以下各项中的一个或多个：启发式分析、恶意软件签名匹配以及对所述对象的至少所述虚拟副本的行为分析。

9. 根据权利要求 7 所述的系统，其中，所述安全执行环境实施为所述主机上的用户模式服务。

10. 根据权利要求 7 所述的系统，其中，所述安全执行环境进一步包括：所述主机的一个或多个虚拟用户模式服务，一个或多个虚拟内核对象，虚拟注册表和虚拟文件系统。

11. 根据权利要求 7 所述的系统，其中，所述内核级驱动程序进一步经配置以：

拦截来自所述用户模式应用程序的远程过程调用；以及

将被所述远程过程调用所寻址的对象的端口名称用在所述安全执行环境中的所述对象的虚拟副本的端口名称替换，从而将所述远程过程调用重定向到所述安全执行环境。

## 用于使用虚拟化技术进行恶意软件保护的系统和方法

### 技术领域

[0001] 本发明总体上涉及计算机安全领域,且更具体而言,涉及用于使用虚拟化技术保护部署于主机(host computer)上的应用程序抵御恶意软件的系统、方法及计算机程序产品。

### 背景技术

[0002] 由于诸如病毒、蠕虫、木马、和其他类型计算机威胁的恶意软件的快速扩散和演变,计算机安全专家即使使用自动化的恶意软件检测手段也越来越难以跟踪新兴的威胁。当必须保护机密或者秘密的个人或公司信息的安全时,安全方面的考虑甚至更高。某些类型的恶意软件被专门设计以攻击计算机和部署于其上的应用程序,以收集机密或秘密的用户和系统信息。因此,诸如反病毒程序和防火墙的安全应用程序,必须被配置为保护关键的系统和应用程序对象免受未经授权的访问。

[0003] 一种用于保护系统抵御恶意软件的机制是“沙盒(sandboxing)”,其中,不受信任的程序在安全的虚拟环境中执行。可以限制程序的执行,以排除对主机系统的关键领域或进程或者对部署于其上的应用程序的访问。然而,已知的沙盒技术有局限性。例如,当主机系统已经被恶意软件所感染时,他们可能是无效的。而且,他们不允许基于读/写操作以外的操作类型对不受信任的程序做出的系统调用进行过滤,也不允许使用不同恶意软件检测算法进行请求分析。因此,需要改进沙盒机制,以保护主机和部署于其上的应用程序远离恶意软件。

### 发明内容

[0004] 本申请公开了用于使用虚拟化技术保护部署于主机上的应用程序抵御恶意软件的系统、方法及计算机程序产品。在一个示范性实施例中,恶意软件保护系统包括内核级驱动程序,所述内核级驱动程序经配置以拦截寻址到受保护应用程序的对象的系统调用。所述系统进一步包括分析引擎,所述分析引擎经配置以确定是否有与以下各项中的一个或多个相关联的安全规则:所拦截的系统调用、所述受保护应用程序的所述对象以及在所述受保护应用程序的所述对象上所允许的动作,其中所述安全规则至少指示是否允许所述系统调用在所述主机上执行或者不允许所述系统调用在所述主机上执行。如果有安全规则指示允许所述系统调用在所述主机上执行,则所述分析引擎指令所述主机执行所述系统调用。如果有安全规则指示不允许所述系统调用在所述主机上执行,则所述分析引擎指令所述主机阻止所述系统调用的执行。如果没有与所述系统调用相关联的安全规则,则所述分析引擎指令所述主机的安全执行环境的处理程序使用所述受保护应用程序的所述对象的虚拟副本在所述安全执行环境中执行所述系统调用。

[0005] 在一个示范性实施例中,用于保护部署于主机上的应用程序的方法包括:在所述主机的内核级,拦截寻址到部署于所述主机上受保护应用程序的对象的系统调用;确定是否有与以下各项中的一个或多个相关联的安全规则:所拦截的系统调用、所述受保护应用

程序的所述对象以及在所述受保护应用程序的所述对象上所允许的动作，其中所述安全规则至少指示是否允许所述系统调用在所述主机上执行或者不允许所述系统调用在所述主机上执行；如果有安全规则指示允许所述系统调用在所述主机上执行，则在所述主机上执行所述系统调用；如果有安全规则指示不允许所述系统调用在所述主机上执行，则在所述主机上阻止所述系统调用的执行；以及如果没有与所述系统调用相关联的安全规则，则使用所述受保护应用程序的所述对象的虚拟副本在安全执行环境中执行所述系统调用。

[0006] 以上对本发明示范性实施例的简要概括用于提供对这类实施例的基本理解。此概括并不是本发明设想的所有方面的宽泛概述，并且既不意图确定所有实施例的关键或决定性要素也不意图限制任何或所有实施例的范围。其唯一目的在于简要地提出一个或多个方面的一些构思，作为下面更为详细的描述的前序。为了实现前述的以及相关的目的，一个或多个实施例包括将在下面充分描述且在权利要求书中特别指出的特征。

## 附图说明

[0007] 附图包含于说明书中并构成说明书的一部分，示出了本发明的一个或多个示范性实施例，与详细描述一起用于解释本发明实施例的原理和实施方式。

[0008] 附图中：

[0009] 图 1 示出了根据一个示范性实施例的主机的原理图配置；

[0010] 图 2 示出了根据一个示范性实施例的部署于主机上的恶意软件保护系统的示范性实施方式；

[0011] 图 3 示出了根据一个示范性实施例的虚拟化主机 COM 子系统的示例；

[0012] 图 4 示出了根据一个示范性实施例的恶意软件保护系统的安全执行环境的操作的示意图；

[0013] 图 5 示出了根据一个示范性实施例的安全执行环境的虚拟组件的示意图；

[0014] 图 6 示出了根据一个示范性实施例的用于由恶意软件保护系统拦截本地过程调用的方法；

[0015] 图 7 示出了根据一个示范性实施例的恶意软件保护系统的操作的原理示意图；

[0016] 图 8 示出了根据一个示范性实施例的恶意软件保护系统的操作的算法；

[0017] 图 9 示出了根据一个示范性实施例的恶意软件保护系统的原理示意图；以及

[0018] 图 10 示出了根据一个示范性实施例的主机的原理示意图。

## 具体实施方式

[0019] 在本申请中，围绕用于使用虚拟化技术保护部署于主机上的应用程序抵御恶意软件的系统、方法及计算机程序，来描述本发明的示范性实施例。本领域普通技术人员应认识到，下面的描述仅仅是示例性的而并非意图以任何方式进行限定。受益于此公开内容的本领域技术人员将容易获得其他实施例的启示。现在，将更为详细地描述如图所示的示范性实施例的实施方式。贯穿全部附图以及下列描述，相同的附图标记将尽可能用于表示相同或相似的对象。

[0020] 图 1 示出了主机的示范性原理图配置，所述主机包括硬件 120、操作系统 170 和应用程序 / 程序 100。更详细的主机配置在本发明下面将更详细的讨论的图 10 中示出。一

般来说,操作系统(OS)支持用于在主机上执行程序和应用程序的几个级别的权限。例如,Windows®操作系统支持两种安全模式:内核模式 110 和用户模式 115。程序和应用程序 100 通常部署于用户模式 115 中,这提高了对于主机抵御严重错误的保护水平,尤其保护了系统服务 130、驱动程序(driver)140 和操作系统 170 的硬件抽象层 160 免受恶意软件的访问/修改。主机还可包括各种用户模式的服务 150,如操作系统提供的反病毒和其他安全服务,所述服务 150 控制用户模式应用程序 100 的执行,并监视主机的软件和硬件组件的状态。

[0021] 图 2 示出了部署于主机上的恶意软件保护系统的示范性实施例。该恶意软件保护系统包括内核级保护驱动程序 230、分析引擎 210 和安全执行环境(secure execution environment, SEE) 220。在一个示范性实施例中,内核级保护驱动程序 230 经配置以拦截指向用户模式应用程序 100 的系统调用(例如,读、写、加载、创建进程、开放网络连接等)并将它们重定向到安全执行环境(SEE) 220。在 SEE220 中,由分析引擎 210 分析请求以确定是否阻止系统调用、允许它由主机执行或使用 SEE220 的虚拟系统组件在 SEE220 中执行系统调用,而不会对应用程序 100 或主机造成任何伤害。

[0022] 在一个示范性实施例中,SEE220 可以实施为提供主机的组件的虚拟化的用户模式服务 150。图 3 示出了可以实施于 SEE220 的主机的 COM 子系统的虚拟化示例。一般地,该 COM 提供了用于系统组件的重用的模型,如可执行文件或动态库,它可以由支持该 COM 模型的任何程序调用。这种程序的示例是 Internet Explorer®和其他浏览器。COM 提供了对这种应用程序的启动副本的访问和控制。因此,在一个示范性实施例中,当在主机上所启动程序(这可能是恶意软件)寻址 COM 服务 300 时,系统调用可被重定向到 SEE220 中所创建的 COM 服务的虚拟副本 301。因此,对操作系统的 COM 对象和其它 OS 服务的访问不会直接进行。这种 COM 虚拟服务与原始服务没有区别。例如,当应用程序 100 做出系统调用时,主机系统启动 COM 服务器 310,它向应用程序 100 返回所请求的对象。当来自应用程序 100 的系统调用被重定向到 SEE220 时,虚拟的 COM 服务 301 创建虚拟服务器 311 并向应用程序 100 返回虚拟对象。

[0023] 图 4 示出了根据一个示范性实施例的恶意软件保护系统的安全执行环境(SEE)的操作的示意图。在主机上执行期间,应用程序 100 可能会发送系统调用到 OS400,在那里该系统调用必须由 OS 处理程序(handler)405 处理。系统调用可以被恶意软件保护系统的内核级保护驱动程序 230 拦截并且不会到达 OS 处理程序 405。例如,保护驱动程序 230 可配置为拦截以下系统调用:开放进程/输入;读取进程存储器;对存储器/磁盘的直接访问;对文件系统、注册表和 OS 内核 400 的对象的访问;请求获得权限;以及其他系统调用。在该拦截后,保护驱动程序 230 重定向系统调用到安全执行环境 220,在那里系统调用可以由 SEE 处理程序 410 使用主机的多个组件的虚拟副本加以执行。

[0024] 由 SEE 处理程序 410 执行系统调用的过程取决于主机在安全执行环境 220 中虚拟化的详细程度。为了无差错地有效操作,最好但不是必须,创建文件系统、注册表和一些 OS 服务的虚拟副本。图 5 示出了再现于 SEE220 中的主机的虚拟组件的示例。如图所示,SEE220 可包括虚拟用户模式服务 500 和虚拟 COM 子系统 505、虚拟内核对象 510(例如,端口、管道、事件、互斥量(mutex)、段(section)、旗语(semaphore))、虚拟注册表 520 和虚拟文件系统 530。系统组件的虚拟化可以即时(on-the-fly)执行而无需重新安装或重加载主

机,这是因为主机的操作系统内核中必然没有变化。

[0025] 虚拟化意味着,应用程序 100 在主机中所做的改变实际上并没有执行。换句话说,在主机系统(文件或注册表项)中的原始信息没有改变。进程的虚拟化被理解为是其对 OS 对象的访问的虚拟化,包括其文件和注册表。如果必要的话,对象的虚拟化以其虚拟副本的生成为先决条件(也提供对实际对象的访问)。在一个示范性实施例中,SEE 处理程序 410 确定需要在 SEE220 中进行虚拟化的软件和硬件组件,并使用这些虚拟化组件执行系统调用。作为一个规则,仅在读取操作期间才将真正的对象返回给应用程序 100。虚拟文件可以存储在专门的目录中,以及注册表项可以存储在注册表的专门分支中。其他 OS 对象可以存储在 RAM 中。

[0026] 在一个示范性实施例中,保护驱动程序 230 还可配置为拦截本地过程调用(local procedure call, LPC)。这使恶意软件保护系统可以通过改变端口名称来隔离应用程序 100。图 6 示出了这种拦截的示例。应用程序 100 发送创建对象的请求到对象链接和嵌入库 600(Windows® OS 中的 OLE32.DLL)。这个库调用建立连接到端口的函数 NtConnectPort (“\\RPC\_Control\epmapper”),其中“\\RPC\_Control\epmapper”是该端口的名称。这个请求被保护驱动程序 230 拦截并且该端口名称被更改。在替代该端口的名称后,请求不会跳转到用于(在 Windows® OS 中的)远程过程调用的服务 RPCSS610,而是跳转到由安全执行环境 220 所提供的用于远程过程调用的虚拟服务即虚拟 RPCSS615。虚拟服务 615 返回 epmapper 接口到应用程序 110,以与安全执行环境 220 进一步交互。

[0027] 前面的例子针对应用程序与 OS 的交互而加以考虑。这样的例子揭示了保护 OS 抵御部署于主机上的恶意软件的可能性。然而,在计算机上虚拟化所有应用程序是极为资源密集型的任务,这不是解决保护机密资料抵御恶意软件的问题的合宜之计。在计算机上,存储、传输、接收含有应予保护的机密数据的信息的应用程序组是有限的。这些信息包括个人通信、个人数据、密码、口令、信用卡号码和其他数据。在信息安全的风险不断增长的情况下,按照指定规则对应用程序和其与其他程序和 OS 的交互进行虚拟化,是必要的一层保护。因此,恶意软件保护系统允许在不丧失功能或无需重装的情形下,使包含机密信息的应用程序与恶意软件相隔离。

[0028] 由于根据本发明的一个示范性实施例的恶意软件保护系统使用对主机的对象和组件及其上所部署的应用程序进行虚拟化的技术,所以,对主机免受由于应用程序中的漏洞和错误所引起的感染的保护可以基于以下原则加以组织:(1)限制对受保护的对象的访问,所述受保护的对象诸如包含用户机密数据的对象;(2)保护主机免遭改变:在试图修改主机上的对象时,创建和制作在安全执行环境中启动的应用程序的对象的可用副本;和(3)保护主机上启动的关键进程和服务免受安全执行环境中启动的应用程序方的访问,以排除恶意进程进入主机的漏洞。

[0029] 在这种情况下,恶意软件保护系统被配置为,在安全执行环境中分析和过滤从应用程序到主机的对象(例如,存储器区域、文件、进程、执行线程)和部署于主机上的其他应用程序的系统调用。此外,恶意软件保护系统可分析和过滤从主机的对象到在安全执行环境中执行的应用程序的系统调用。

[0030] 此外,该恶意软件保护系统提供了以下优点:在应用程序访问因特网期间,保护主机免遭恶意软件的侵入;清空临时文件、访问日志、和其他存储应用程序的操作历史的数

据 ;保护由受保护应用程序所处理的用户数据,以免遭来自主机的进程的未经授权的访问 ;通过在安全执行环境中自动启动程序,确保从受保护应用程序启动这些程序。

[0031] 图 7 示出了根据一个示范性实施例的恶意软件保护系统的操作原理示意图。部署于主机 700 上的恶意软件 710 执行用于从用户模式应用程序 705 窃取机密信息的算法,例如从因特网浏览器、银行应用程序、通信应用程序或存储机密的用户或系统信息的其他应用程序。恶意软件 710 向操作系统发出系统调用,例如请求读取应用程序 705 的一个或多个对象(例如,存储器区域、文件、进程、执行线程等)的存储器。系统调用可被恶意软件保护系统的内核级保护驱动程序 230 拦截并重定向到恶意软件保护系统的安全执行环境 220。SEE220 可收集有关系统调用的所有信息,诸如发出调用的程序的名称、系统调用的类型(例如,读、写、加载、创建进程、开放网络连接等)、被调用的程序的名称、被系统调用所访问的进程和存储器区域以及其他相关信息。SEE220 发送收集到的信息到分析引擎 210,分析引擎 210 确定系统调用的性质,并得到是否执行系统调用、阻止系统调用、或使用 SEE220 中的虚拟系统组件来执行系统调用的结论。

[0032] 在一个示范性实施例中,分析引擎 210 是安全规则 720 的专家系统,它对以下内容进行分析 :所拦截的系统调用(例如,系统调用的类型)、系统调用的对象、来自恶意软件 710 的系统调用的统计数据、恶意软件 710 的属性和行为以及其他确定是否应由恶意软件保护系统执行或阻止系统调用的信息。安全规则的一个示例是根据由下面的例子中用尖括号所指定的对象类型来排除访问 :

[0033] 1f{access to<object of the OS kernel>} , then{block access}

[0034] 根据结论,系统调用或者被阻止,或者不经修改被发送到 OS 处理程序 405,或者被转发到 SEE 处理程序 410 以在 SEE220 中执行。对进程、数据和应用程序存储器的请求被根据这些规则进行过滤,这加强了对访问用户模式应用程序 705 的机密数据 730 的保护。

[0035] 图 8 示出了根据一个示范性实施例的恶意软件保护系统的操作的算法。在步骤 800,内核级保护驱动程序拦截对(或来自)用户模式应用程序(例如,应用程序的存储器、其进程、应用程序的文件)的系统调用。在步骤 810,由 SEE 的分析引擎检查系统调用是否符合安全规则,并在步骤 815 决定如何处理该系统调用 :如果它不符合安全规则,则在步骤 820 阻止系统调用(并且可生成表明禁止系统调用的错误消息);如果它不含有任何威胁,则在步骤 825 执行系统调用(发送到 OS 处理程序);或者如果没有发现相应的安全规则,则在步骤 830 在 SEE 中虚拟化系统调用的执行。特别是,在步骤 830,SEE 处理程序在 SEE 中创建系统调用的虚拟执行所必须的系统组件的所有必要虚拟副本,诸如虚拟系统服务、注册表、文件系统等。在 SEE 中所执行的动作可在步骤 840 存储在数据结构中,并在步骤 845 评估它们是否对系统调用所指向的应用程序(或其数据、进程等)的安全有威胁。在一个示范性实施例中,步骤 845 中的分析可包括对一个或多个所拦截的系统调用的相对于已知的恶意程序行为模式的启发式分析,例如共同所有的、标题为“用于检测多组件恶意软件的系统和方法”的第 7,614,084 号美国专利中所公开的,以援引的方式将该专利合并到本发明中。在其他的示范性实施例中,步骤 845 的分析可包括恶意软件签名匹配、行为分析或其他已知的恶意软件检测技术。如果没有检测到威胁,对 SEE 的对象的变化可在步骤 850 应用于主机。否则,SEE 的变化不会被应用于主机,并且在步骤 855 终止进程。

[0036] 图 9 示出了根据一个示范性实施例的恶意软件保护系统的示意图。内核级保护驱

动程序 230 拦截系统调用。可拦截的请求的示例如上所述。所拦截的系统调用被重定向到 SEE 900，在那里对受保护的应用程序对象和主机系统组件进行虚拟化。如果系统调用包括对受限访问的关键对象的请求，则可创建该对象的虚拟副本并返回给发起该请求的程序。如图所示，虚拟化的对象可包括但不限于用户模式服务 500、OS 内核 510、OS 注册表 520 和文件系统 530。除了由 SEE 强加到关键系统对象上的约束以外，SEE 还包括分析引擎 210，其经配置以如上所述地使用安全规则 215 分析系统调用是否存在安全威胁。在一个示范性实施例中，分析引擎 210 可实现安全评级算法，其中安全规则对所拦截的系统调用进行安全评级的评估和分配，如共同所有的、标题为“用于计算机进程安全评级的系统与方法”的第 7,530,106 号美国专利所具体公开的，该专利以援引的方式合并到本发明中。基于所分配安全评级，分析引擎 210 可以决定是否如以上参照图 7 和 8 所详细描述地那样，执行、阻止或虚拟化所拦截的系统调用。

[0037] 图 10 示出了适用于实现主机的诸如个人计算机 (PC) 或应用程序服务器的计算机系统 5 的一个示范性实施例。如图所示，计算机系统 5 可以包括通过系统总线 10 连接的一个或多个处理器 15、存储器 20、一个或多个硬盘驱动器 30、光驱 35、串行端口 40、图形卡 45、声卡 50 和网卡 55。系统总线 10 可以为几种类型的总线结构中的任何一种，包括使用各种已知总线架构中的任何一种的存储器总线或存储器控制器、外围总线和局部总线。处理器 15 可以包括一个或多个 Intel® Core 2Quad 2.33GHz 处理器或其他类型的通用微处理器。

[0038] 系统存储器 20 可以包括只读存储器 (ROM) 21 和随机存取存储器 (RAM) 23。存储器 20 可以实施为 DRAM (动态 RAM)、EPROM、EEPROM、闪存或其他类型的存储器架构。ROM 21 存储基本输入 / 输出系统 22 (BIOS)，该基本输入 / 输出系统 22 含有帮助在计算机系统 5 的组件之间传送信息的基本例程，诸如在启动期间。RAM 23 存储操作系统 24 (OS)，诸如 Windows® XP Professional 或其他类型操作系统，该操作系统 24 负责在计算机系统 5 中管理和协调进程以及分配和共享硬件资源。系统存储器 20 还存储应用程序和程序 25，诸如服务 306。系统存储器 20 还存储由程序 25 使用的各种运行时数据 26 以及关于已知恶意的和安全的对象的信息的各种数据库。

[0039] 计算机系统 5 可以进一步包括诸如 SATA 磁性硬盘驱动器 (HDD) 的硬盘驱动器 30 以及用于读或写诸如 CD-ROM、DVD-ROM 或其他光学介质的可移动光盘的光盘驱动器 35。驱动器 30 和 35 及其相关联的计算机可读介质提供了对计算机可读指令、数据结构、数据库、应用程序以及实施本发明所公开的算法和方法的程序模块 / 子例程的非易失性存储。尽管示例性的计算机系统 5 采用磁盘和光盘，本领域技术人员应当理解，在计算机系统的替代实施例中，还可以使用可存储计算机系统 5 可访问数据的其他类型的计算机可读介质，诸如磁带盒、闪存卡、数字视频盘、RAM、ROM、EPROM 以及其他类型的存储器。

[0040] 计算机系统 5 进一步包括用于连接诸如键盘、鼠标、触控板及其它类型的数据输入设备 75 的多个串行端口 40，诸如通用串行总线 (USB)。串行端口 40 还可以用来连接诸如打印机、扫描仪及其它类型的数据输出设备 80 以及诸如外部数据存储设备等的其它外围设备 85。系统 5 还可以包括诸如 nVidia® GeForce® GT240M 或其他视频卡的图形卡 45，用于与监视器 60 或其他视频再现设备接口。系统 5 还可以包括声卡 50，用于经由内部或外部扬声器 65 再现声音。此外，系统 5 可以包括网卡 55，诸如以太网、WiFi、GSM、蓝牙或其他

有线、无线或蜂窝网络接口，用于将计算机系统 5 连接至诸如因特网的网络 70。

[0041] 在各种实施例中，本发明所描述的算法和方法可以实现于硬件、软件、固件或者其任意组合中。如果在软件中实现，这些功能可以作为一个或者多个指令或者代码在非暂时性计算机可读介质上存储。计算机可读介质既包括计算机存储介质也包括通信介质，所述通信介质有助于从一个地方向另一个地方传送计算机程序。存储介质可以是能够被计算机访问的任何可用的介质。作为示例而非限制，这种计算机可读介质可包括 RAM、ROM、EEPROM、CD-ROM 或其它光盘存储器、磁盘存储器或其它磁性存储设备、或者任何能够用于以指令或者数据结构的形式承载或者存储所需的程序代码并且可被计算机访问的其它介质。而且，任何连接都可以被称为计算机可读介质。例如，从网站、服务器或者其它远程信源传输软件的同轴电缆、光缆、双绞线、数字用户线(DSL)、或者诸如红外、射频和微波这类无线技术包括在介质的定义中。

[0042] 为了清楚起见，本发明没有示出并描述实施例的所有常规特征。应该理解的是，在任何这种实际的实施方式的开发中，为了达到开发者的特定目标，必须做出大量特定于实施方式的决定，而且，这些特定目标会因实施方式的不同和开发者的不同而变化。应该理解的是，这种开发工作可能是复杂且费时的，但不论如何，对于受益于本申请的普通技术人员而言，都将是常规的工程任务。

[0043] 而且，可以理解的是，本发明使用的措辞和术语用于描述而非限制的目的，以使本说明书的术语或者措辞可由本领域技术人员在本发明提出的教导和指导下结合相关领域技术人员的知识做出解释。而且，除非像这样明确地予以阐述，否则说明书中或者权利要求中的任何术语都并非意图表示不常见的或者特殊的意思。

[0044] 本发明公开的各种实施例包括本发明通过图示方式提到的已知组件的现在和将来已知的等同物。而且，尽管已经示出和说明了实施例和应用程序，但对受益于本发明公开的内容的本领域技术人员来说显而易见的是，在不脱离本发明公开的发明构思的情况下，比上述提及到的更多的修改例都是可能的。

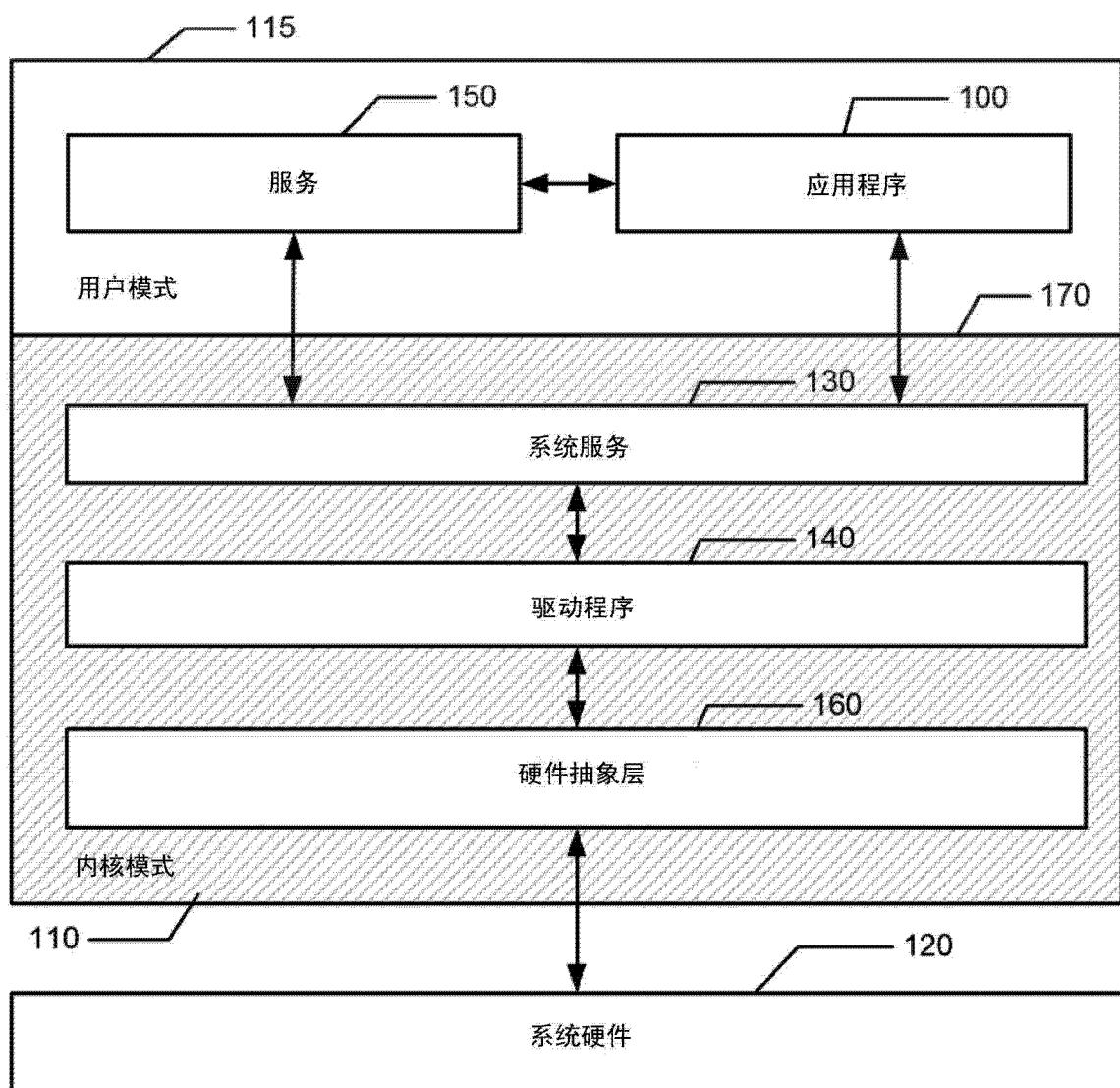


图 1

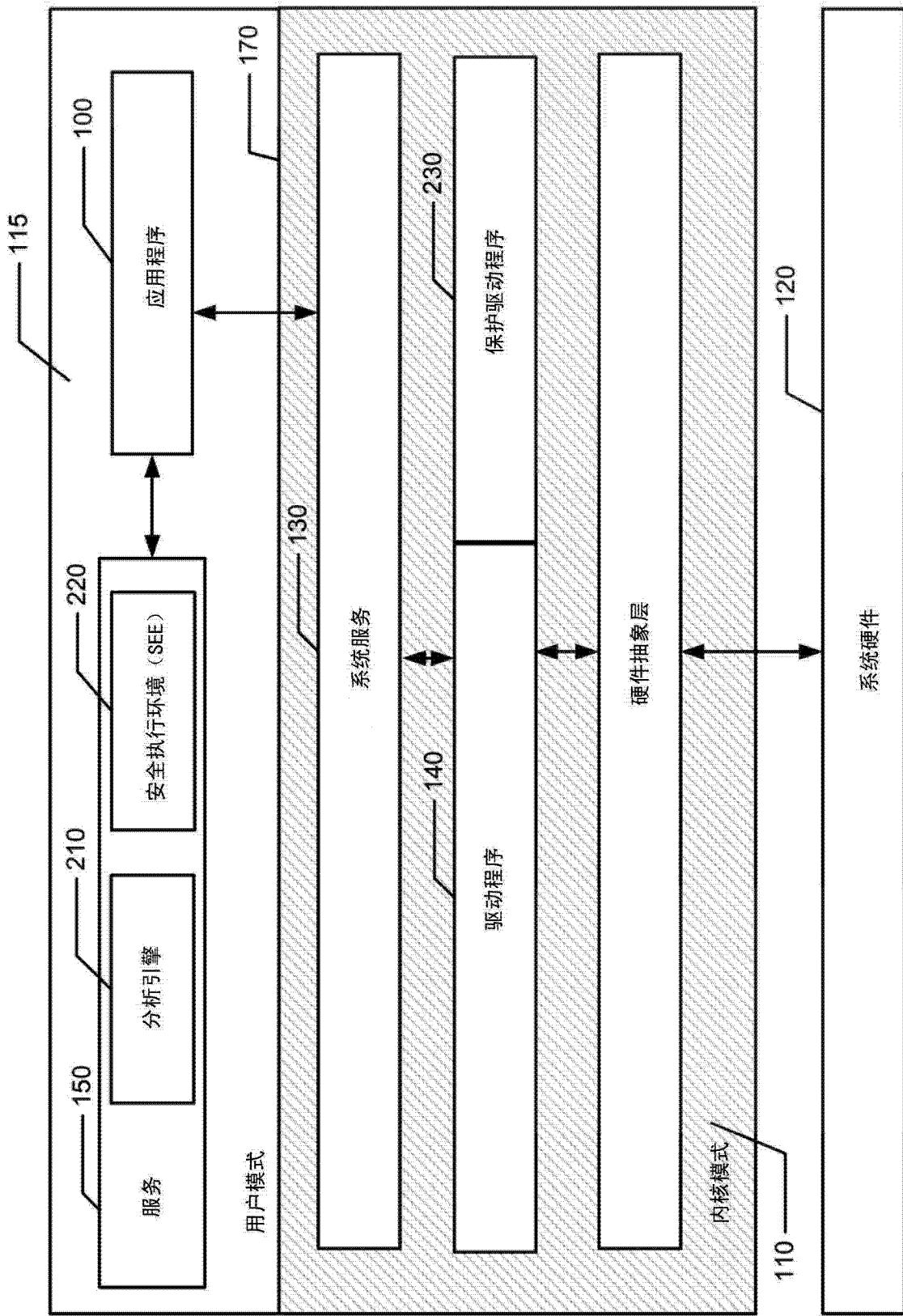


图 2

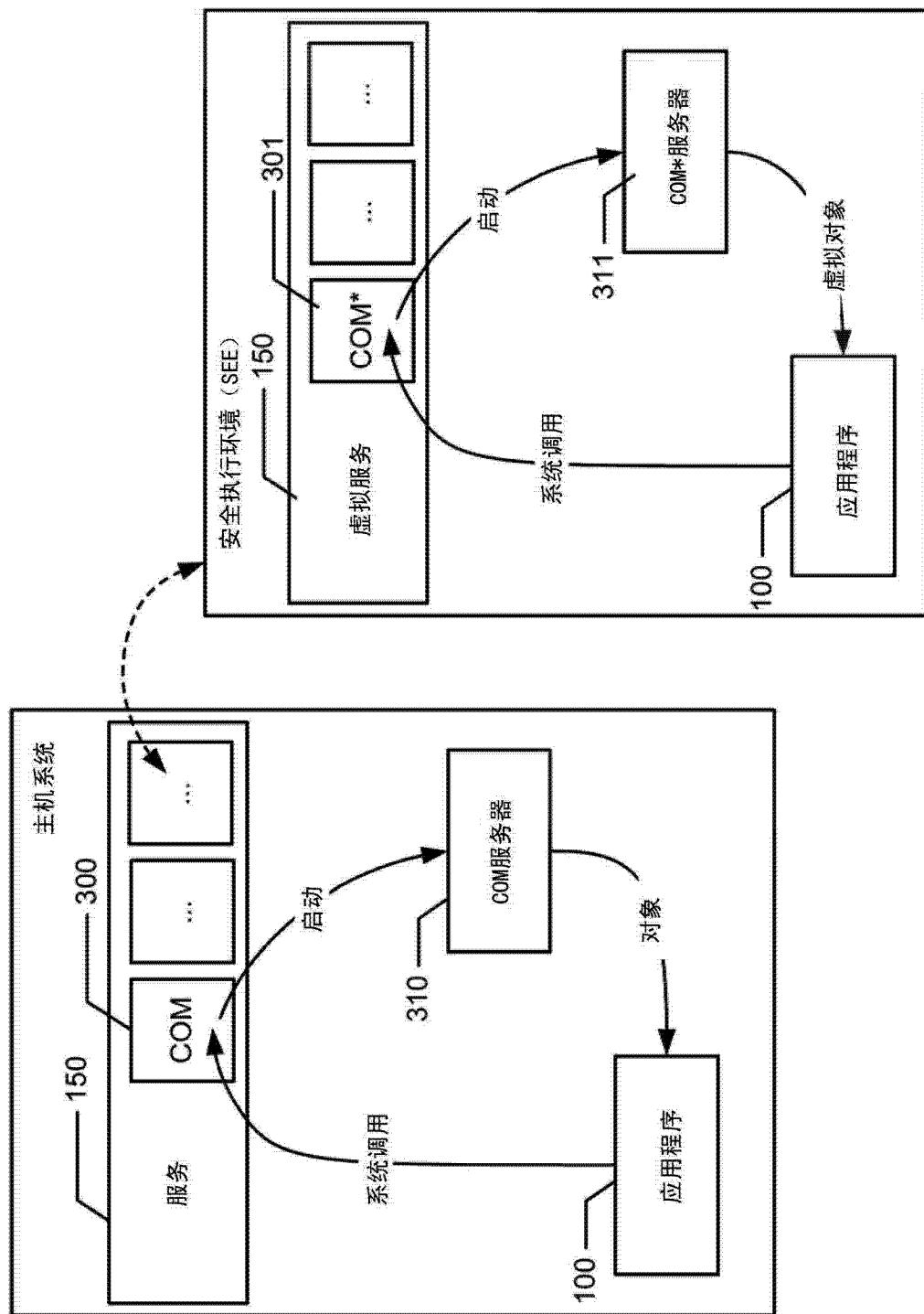


图 3

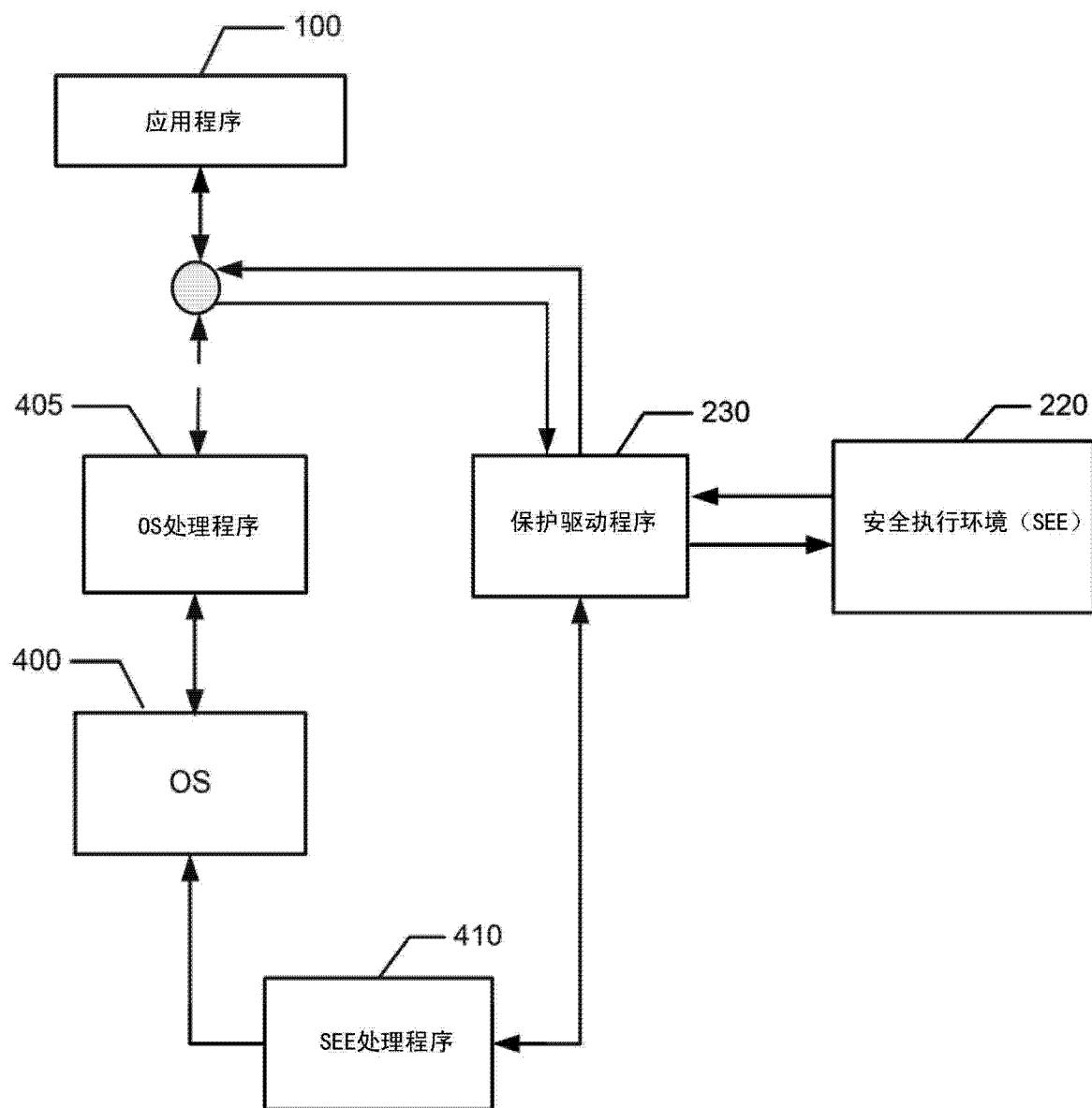


图 4

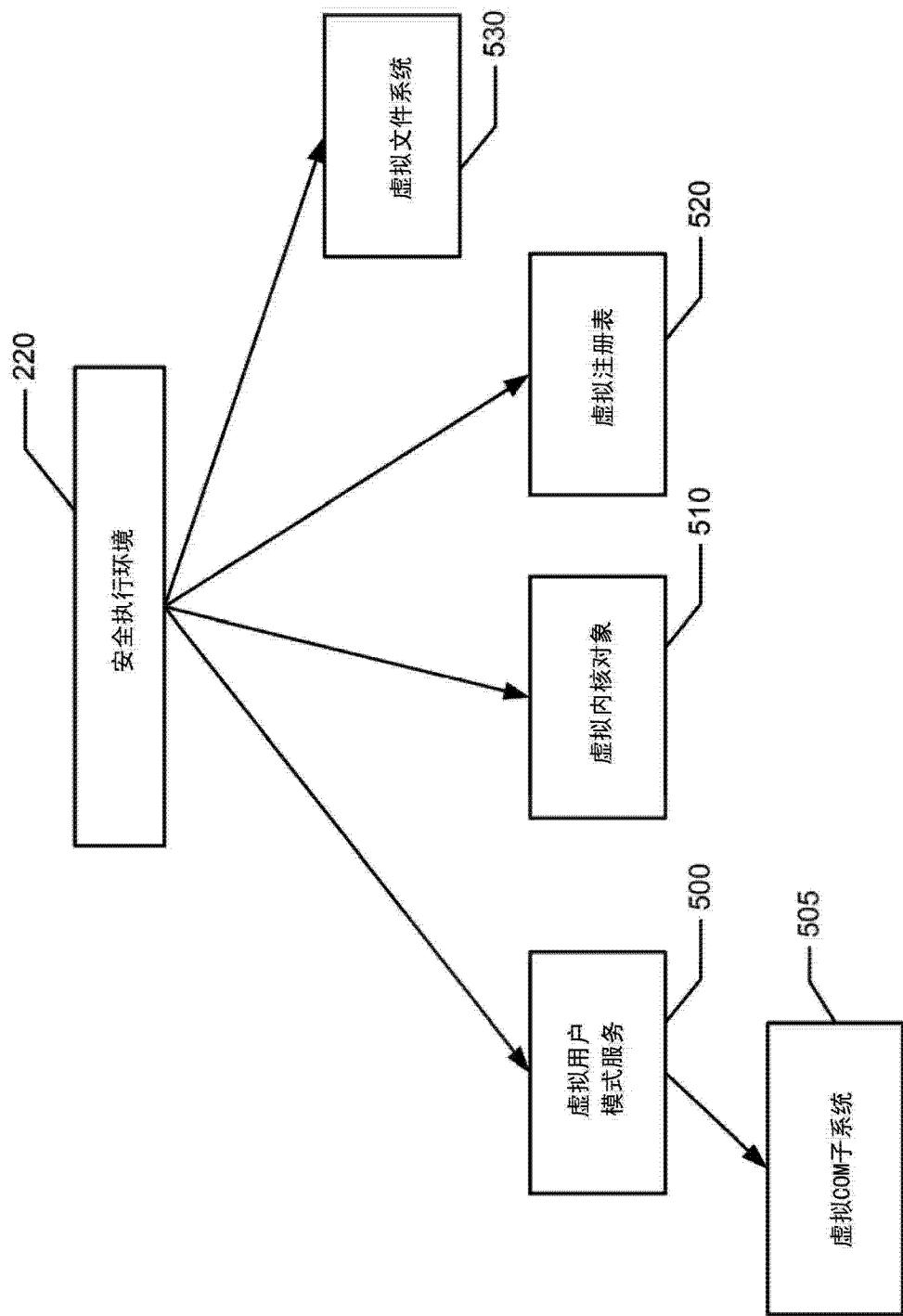


图 5

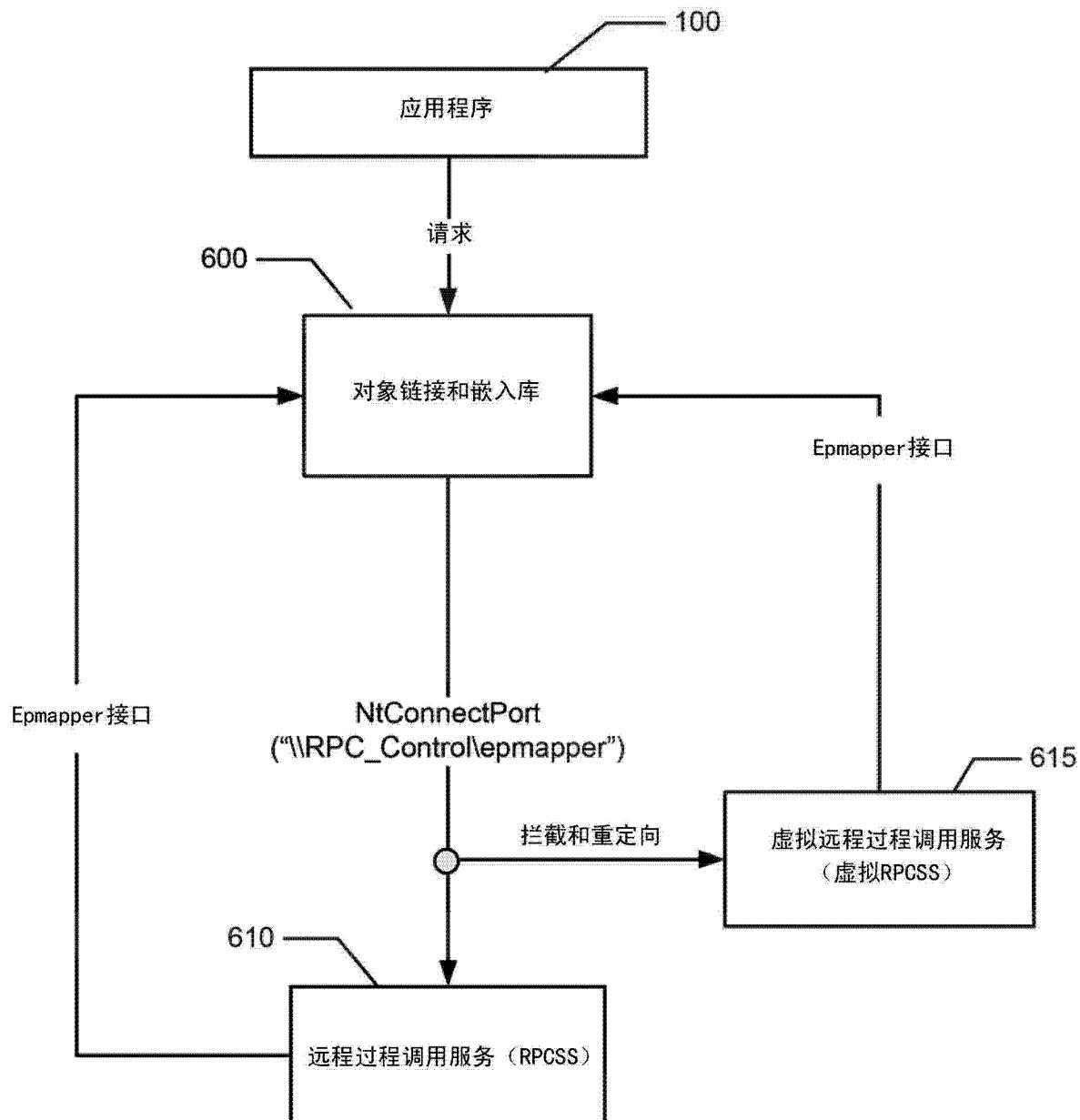


图 6

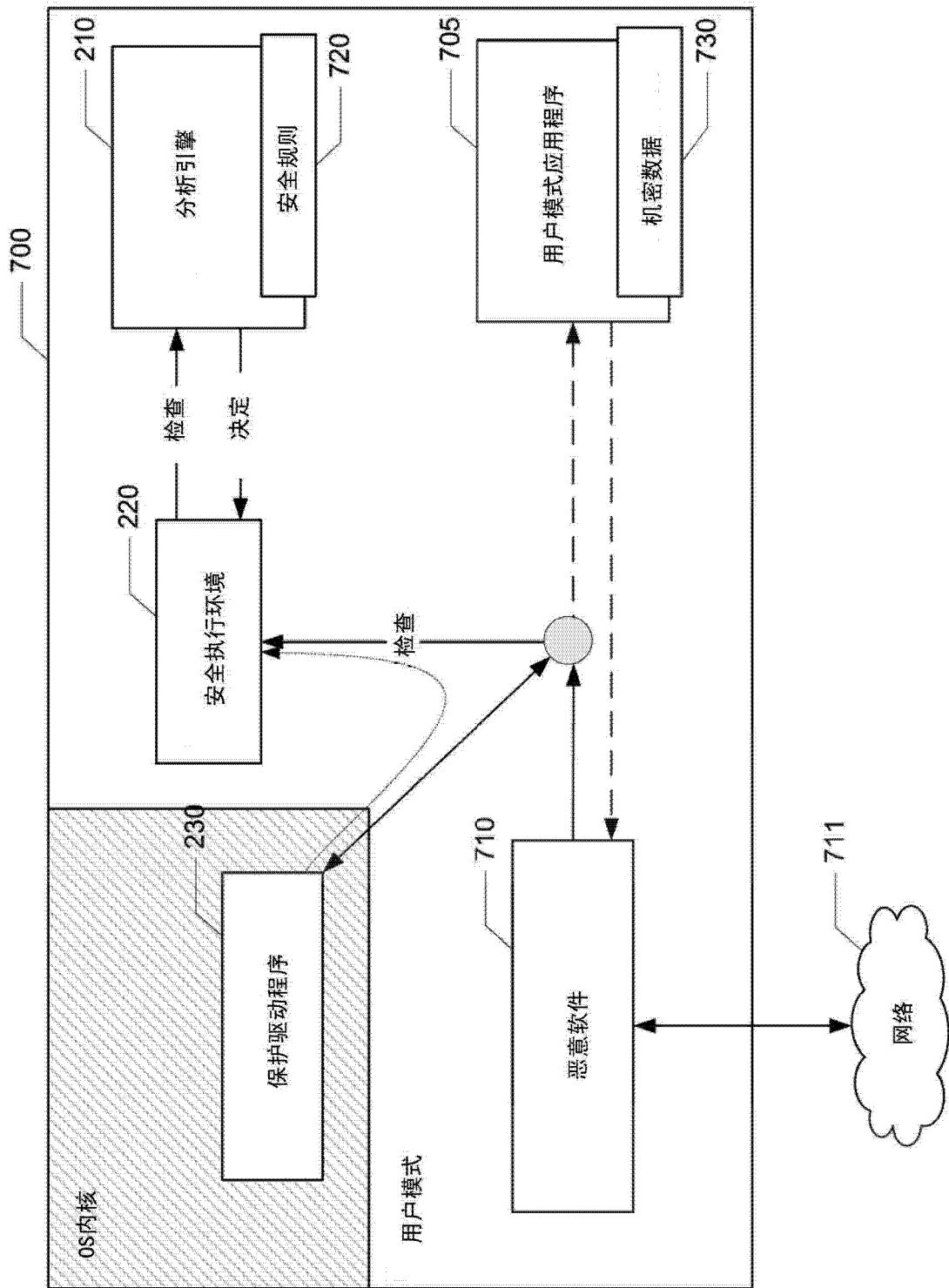


图 7

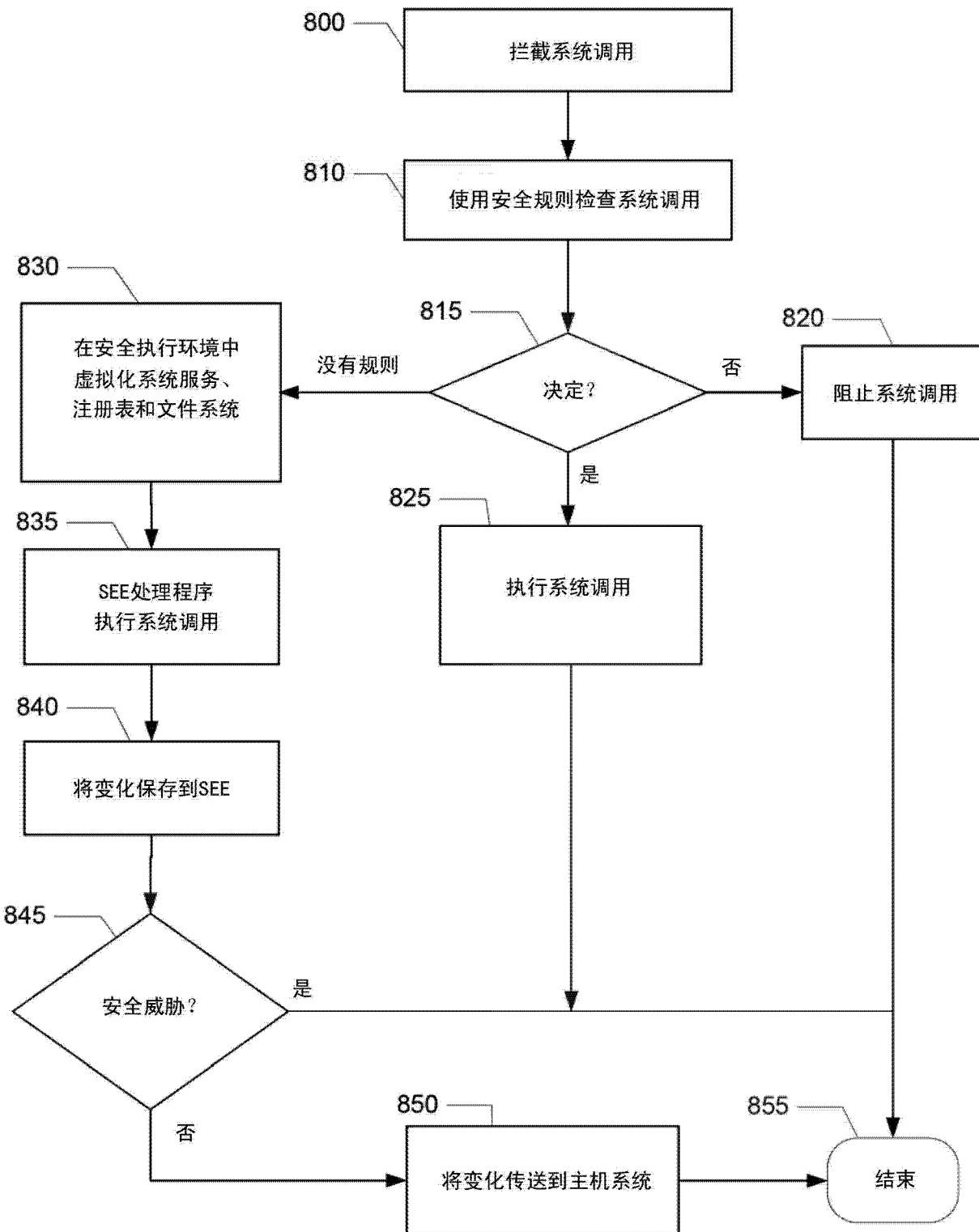


图 8

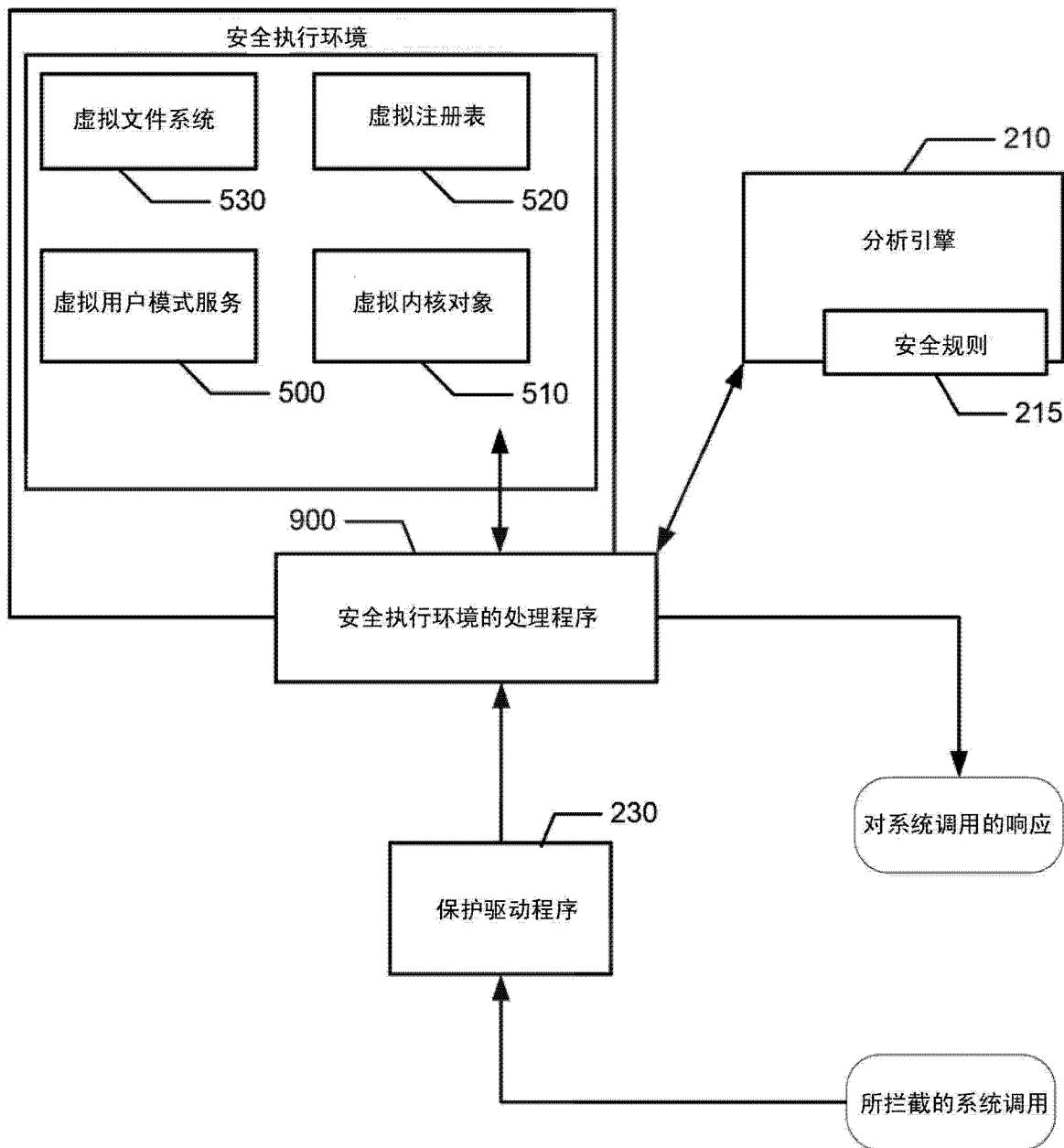


图 9

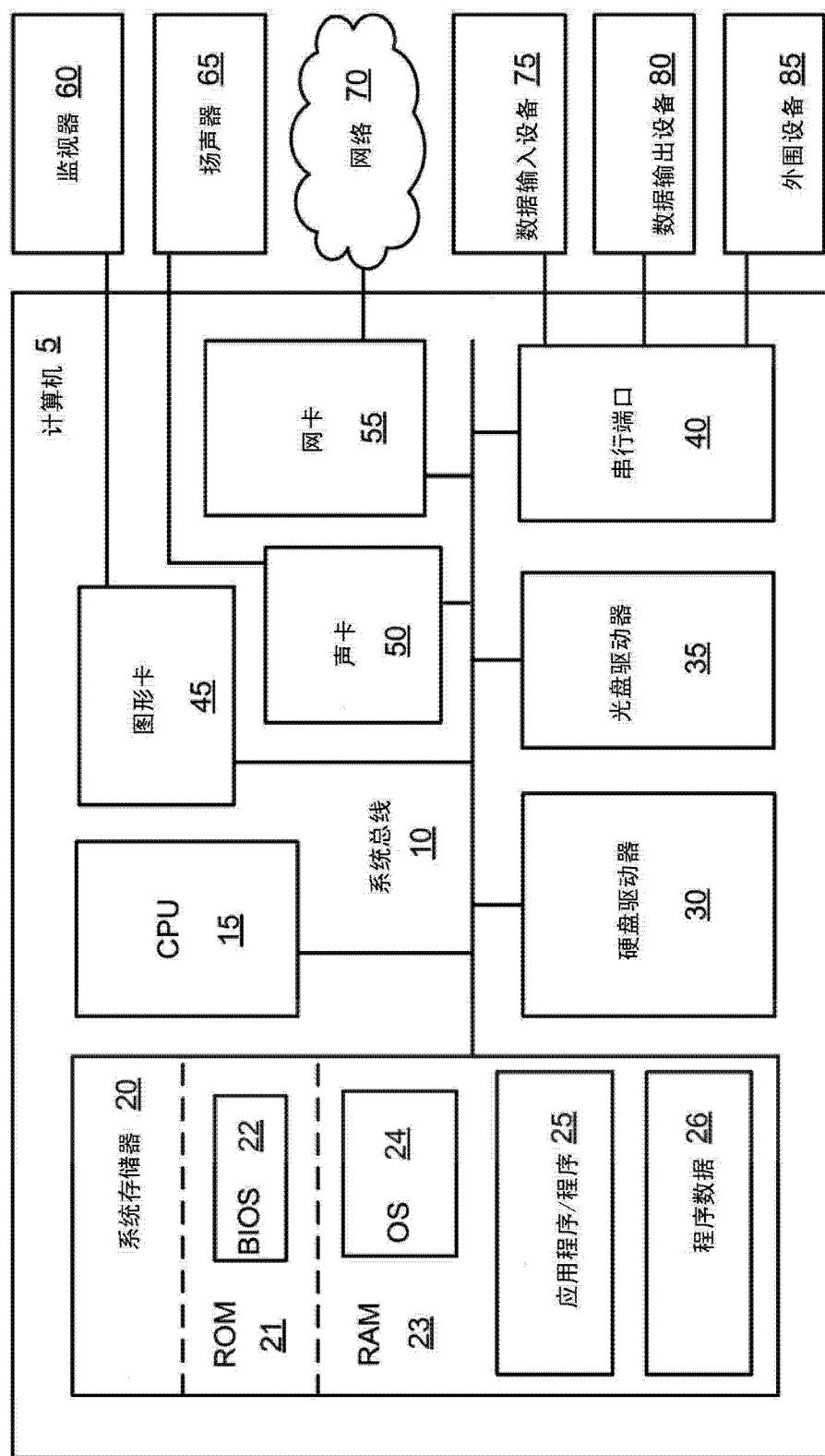


图 10