



(12)发明专利

(10)授权公告号 CN 105453097 B

(45)授权公告日 2018.05.29

(21)申请号 201380077101.2

(22)申请日 2013.09.20

(65)同一申请的已公布的文献号

申请公布号 CN 105453097 A

(43)申请公布日 2016.03.30

(30)优先权数据

13/906902 2013.05.31 US

(85)PCT国际申请进入国家阶段日

2015.11.30

(86)PCT国际申请的申请数据

PCT/US2013/060753 2013.09.20

(87)PCT国际申请的公布数据

W02014/193443 EN 2014.12.04

(73)专利权人 微软技术许可有限责任公司

地址 美国华盛顿州

(72)发明人 G.迪亚斯-奎利亚尔 D.K.古普塔

(74)专利代理机构 北京市金杜律师事务所

11256

代理人 王茂华

(51)Int.Cl.

G06F 21/12(2006.01)

G06F 21/53(2006.01)

(56)对比文件

US 2006242270 A1,2006.10.26,

US 2007088890 A1,2007.04.19,

US 2010082926 A1,2010.04.01,

US 2008005791 A1,2008.01.03,

CN 102037447 A,2011.04.27,

审查员 简文雨

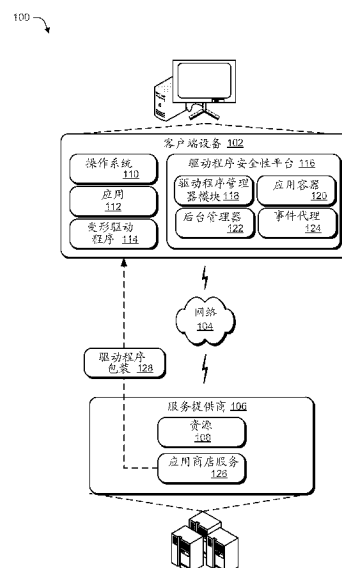
权利要求书1页 说明书11页 附图6页

(54)发明名称

受约束驱动程序平台在用户模式下在沙箱中运行驱动程序

(57)摘要

本文描述一种受约束变形驱动程序平台。在一个或多个实现方案中,提供使得能够针对虚拟私人网络(VPN)驱动程序和其它变形驱动程序实现受约束执行环境的平台。该平台可以实现为暴露接口的操作系统组件,驱动程序可以通过该接口向平台登记并且被调用以执行由平台支持的功能。受约束执行环境向经由平台操作的变形驱动程序放置一个或多个约束。例如,执行可以在每用户的基础上并且在沙箱内发生在用户模式下。另外,平台使得相关联的驱动程序作为后台进程以相对低特权运行。另外,平台可以暂停驱动程序并且通过后台任务的调度控制驱动程序的操作。因此,变形驱动程序向系统的暴露通过平台受到控制和限制。



1. 一种由计算设备实现的方法,该方法包括:

获得包含在具有与驱动程序安全性平台相关联的指定格式的驱动程序包装中的变形驱动程序;

至少部分地基于识别包括在驱动程序安装包中的数据,在安装之后识别驱动程序包装的指定格式;

响应于所述识别,向计算设备所实现的驱动程序安全性平台登记所述变形驱动程序;

经由驱动程序安全性平台针对变形驱动程序实例化受约束执行环境;以及

在受约束执行环境内执行变形驱动程序以在驱动程序安全性平台的指引下执行一个或多个任务。

2. 如权利要求1中所描述的方法,其中变形驱动程序在用户模式下执行。

3. 如权利要求1中所描述的方法,其中变形驱动程序在每用户的基础上执行。

4. 如权利要求1中所描述的方法,其中受约束执行环境建立被配置成隔离变形驱动程序以约束变形驱动程序对系统资源的访问的应用容器。

5. 如权利要求4中所描述的方法,其中应用容器还被配置成拒绝不与驱动程序安全性平台相关联的组件对变形驱动程序的访问。

6. 如权利要求1中所描述的方法,其中受约束执行环境包括用于变形驱动程序的后台进程,通过驱动程序安全性平台针对所述后台进程处置任务的调度。

7. 如权利要求1中所描述的方法,其中受约束执行环境被配置成在没有由驱动程序安全性平台对变形驱动程序的明确调用以执行任务的情况下暂停变形驱动程序。

8. 如权利要求1中所描述的方法,其中受约束执行环境被配置成依赖于用户同意而选择性地启用变形驱动程序的操作。

9. 如权利要求1中所描述的方法:

其中所述获得变形驱动程序是通过网络从服务提供商的应用商店执行的。

10. 如权利要求1中所描述的方法,其中变形驱动程序包括被配置成封装和解封装分组以用于通过网络的VPN通信的虚拟私人网络 (VPN) 驱动程序。

受约束驱动程序平台在用户模式下在沙箱中运行驱动程序

背景技术

[0001] 当前,实行底层数据从一个形式到另一个的变换的变形(transmogrifying)驱动程序,诸如虚拟私人网络(VPN)驱动程序,可以具有系统范围特权、高优先级和对敏感数据的访问并且因此可能呈现重大安全性风险。部分由于这些顾虑,用户可能不情愿在其设备上安装这样的驱动程序及相关联的软件。例如,用户可能由于这种类型产品的假定风险而不愿意从在线应用商店下载和安装VPN插件。因此,包含这样的变形驱动程序的产品的贸易和分发可能受阻。

发明内容

[0002] 本文描述受约束变形驱动程序平台。在一个或多个实现方案中,提供使得能够针对虚拟私人网络(VPN)驱动程序和其它变形驱动程序实现受约束执行环境的平台。平台可以实现为暴露接口的操作系统组件,驱动程序通过该接口可以向平台登记并且被调用以执行由平台支持的功能。受约束执行环境向经由平台操作的变形驱动程序放置一个或多个约束。例如,执行可以在每用户的基础上并且在沙箱内发生在用户模式下。另外,平台使相关联的驱动程序以相对低特权运行在后台进程中。另外,平台可以暂停驱动程序并且通过后台任务的调度控制驱动程序的操作。因此,变形驱动程序向系统的暴露通过平台受到控制和限制。

[0003] 提供本发明内容来以简化形式介绍以下在具体实施方式中进一步描述的概念的选择。本发明内容不意图标识所要求保护的的主题的关键特征或必要特征,其也不意图用作帮助确定所要求保护的的主题的范围。

附图说明

[0004] 参照附图描述具体实施方式。在图中,参考标号最左边的(多个)数位标识其中参考标号第一次出现的图。相同参考标号在描述和图中的不同实例中的使用可以指示相似或同样的项目。

[0005] 图1是依照受约束变形驱动程序平台技术的一个或多个实现方案的环境的图示。

[0006] 图2是依照一个或多个实现方案的示例场景的图示。

[0007] 图3是描绘出用于采用受约束执行环境以供变形驱动程序的执行的示例过程的流程图。

[0008] 图4是描绘出其中控制变形驱动程序的操作的示例过程的流程图。

[0009] 图5是描绘出其中经由在线应用商店分发变形驱动程序的示例过程的流程图。

[0010] 图6描绘出依照一个或多个实施例的示例计算系统。

具体实施方式

[0011] 概述

[0012] 部分由于对于虚拟私人网络(VPN)驱动程序和其它变形驱动程序的安全性顾虑,

用户可能不情愿在其设备上安装采用变形驱动程序的产品。因此,包含这样的变形驱动程序的产品的贸易和分发可能受阻。

[0013] 本文描述受约束变形驱动程序平台。在一个或多个实现方案中,提供使得能够针对私人虚拟网络 (VPN) 驱动程序和其它变形驱动程序实现受约束执行环境的平台。平台可以实现为暴露接口的操作系统组件,驱动程序通过该接口可以向平台登记并且被调用以执行由平台支持的功能。受约束执行环境在经由平台操作的变形驱动程序上放置一个或多个约束。例如,执行可以在每用户的基础上并且在沙箱内发生在用户模式下。另外,平台使相关联的驱动程序以相对低特权作为后台进程运行。另外,平台可以暂停驱动程序并且通过后台任务的调度控制驱动程序的操作。因此,变形驱动程序向系统的暴露通过平台受到控制和限制。

[0014] 在以下讨论中,首先描述可以采用本文描述的技术的示例操作环境。接下来,描述可以在示例环境以及其它环境中实现的示例细节和技术。因此,技术的执行不限于示例环境并且示例环境不限于示例技术的执行。最后,描述可以用于实现一个或多个实施例的示例系统和设备。

[0015] 示例操作环境

[0016] 图1是可操作来采用本文描述的技术的示例实现方案中的环境100的图示。所图示的环境100包括经由网络104通信耦合到服务提供商106的客户端设备102。服务提供商106可以被配置成使各种资源108(例如内容和服务)通过网络104可用于客户端设备102和其它客户端。一般地,使得由服务提供商106可访问的资源108可以包括典型地使得通过网络由一个或多个提供商可用的服务和/或内容的任何适当组合。服务的一些示例包括但不限于搜索服务、电子邮件服务、即时消息传送服务、在线生产力套件以及控制客户端对资源的访问的验证服务。内容可以包括文本、多媒体流、文档、应用文件、照片、音频/视频文件动画、图像、web页面、web应用、设备应用、用于由浏览器或其它客户端应用显示的内容等的各种组合。

[0017] 客户端设备102和服务提供商106可以由一个或多个计算设备实现并且还可以代表一个或多个实体。计算设备可以以各种方式配置。例如,计算设备可以配置为能够通过网络通信的计算机,诸如桌上型计算机、移动站、娱乐器具、通信耦合到显示设备的机顶盒、无线电话、游戏控制台等。因而,计算设备的范围可以从具有大量存储器和处理器资源的完整资源设备(例如个人计算机、游戏控制台)到具有有限存储器和/或处理资源的低资源设备(例如传统机顶盒、手持游戏控制台)。附加地,尽管在一些实例中示出单个计算设备,但是计算设备可以代表多个不同设备,诸如由服务提供商106利用的多个服务器。

[0018] 客户端设备102还被图示为包括操作系统110。操作系统110被配置成向在客户端设备102上可执行的应用112抽象底层硬件的底层功能性。例如,操作系统110可以抽象处理、存储器、网络和/或显示功能性,使得应用112可以在不知晓“如何”实现该底层功能性的情况下编写。应用112例如可以向操作系统110提供数据以供显示设备在不理解将如何执行再现的情况下进行该再现和显示。

[0019] 依照本文描述的技术,客户端设备102还被图示为包括变形驱动程序114和驱动程序安全性平台116,驱动程序安全性平台116可以以各种方式配置以向变形驱动程序114的操作强加约束。变形驱动程序114代表可以由本文描述的驱动程序安全性平台116管理和控

制的各个种类的应用模块、插件和/或其它第三方程序。一般地,变形驱动程序114实行底层数据从一个形式向另一个的变换。一种示例类型的变形驱动程序是虚拟私人网络(VPN)驱动程序,其将数据分组变换(例如封装/解封装)成用于VPN客户端与VPN服务器之间的安全VPN通信的专有形式。其它示例可以包括但不限于对数据分组/去分组的网络接口卡驱动程序、识别和处置恶意软件的抗病毒驱动程序、处理/转换图形的图形处理驱动程序、将应用数据翻译成用于打印机的可打印格式的打印机驱动程序等。本文讨论的技术还适用于VPN隧道和其它互联网协议(IP)隧道、数据管道、输运流、防火墙等。

[0020] 在传统模型中,变形驱动程序114可以具有系统范围特权、高优先级和对敏感数据的访问。一般地,变形驱动程序紧密耦合到操作系统并且在内核模式下运行。因而,变形驱动程序114可以包括和/或被称为内核模式驱动程序、系统驱动程序和/或类驱动程序。因为变形驱动程序114典型地具有提升的特权和/或在内核模式下的完全系统访问,所以这些种类的驱动程序呈现重大安全性风险并且可能变成由恶意方出于非法目的而采用的目标/手段。以不受约束的特权和访问自由操作的变形驱动程序114还可能不利地影响处理资源和电池寿命。此外,变形驱动程序经由应用商店或其它在线信道的分发可能至少部分地由于刚刚列举的顾虑而受阻。

[0021] 然而,依照本文描述的创新原理,可以建立受约束执行环境以使得变形驱动程序114能够以缓解以上顾虑的受约束方式在用户模式下执行。例如,通过控制如上文和下文讨论的受约束执行环境内的变形驱动程序114,向驱动程序的操作强加安全性,所述安全性有效地使系统更为可靠、降低驱动程序操作的资源消耗、防止经由驱动程序对系统的开发利用、使得能够实现每用户安装,并且因此使驱动程序通过互联网信道的分发成为可行选项。

[0022] 特别地,驱动程序安全性平台116表示实例化用于变形驱动程序114的受约束执行环境的客户端设备102的功能性。由驱动程序安全性平台116表示的功能性可以以各种方式实现。在一些实现方案中,驱动程序安全性平台116可以提供为操作系统组件,然而,驱动程序安全性平台116也可以实现为如所图示的独立式组件。如所描绘的,驱动程序安全性平台116可以包括驱动程序管理器模块118、应用容器120、后台管理器122和事件代理124或者以其它方式由它们形成以实现平台的各方面。

[0023] 驱动程序管理器模块118表示识别和管理与平台相关联的变形驱动程序114的平台的功能性。驱动程序管理器模块118可以在每用户的基础上通过应用容器120或针对每一个驱动程序所建立的“沙箱”来控制变形驱动程序114在用户模式下的执行。应用容器120限定可以针对各个驱动程序执行的有限任务集并且约束对所限定的任务集的操作。应用容器120还提供双壁沙箱隔离,这意味着从容器内到系统中和从外部实体到容器中两者来约束访问。这防止外部实体劫持应用容器内的沙箱驱动程序和/或获得对其的控制。

[0024] 后台管理器122可操作来使用后台进程管理应用容器120(及相关联的驱动程序)。这可以包括但不限于实例化用于容器的后台进程以及处置针对后台进程的任务调度。后台进程可以分派有低特权令牌或者传达相对少的特权和访问权限的相当的控制数据。换言之,驱动程序安全性平台116使相对低的优先级与限制访问和/或操作自由的变形驱动程序114相关联。后台管理器122还可以导致暂停针对未使用时的变形驱动程序114的后台进程,其进一步约束变形驱动程序除了在用户和/或驱动程序安全性平台116的具体指引之外执行工作的能力。例如,除了正在驱动程序管理器模块118的指引下执行针对驱动程序或驱动

程序类型所具体限定的任务集时之外,可以暂停变形驱动程序114。

[0025] 后台管理器122可以实现为如所图示的驱动程序安全性平台116的组件。可替换地,后台管理器122可以配置为操作系统(OS)组件,其被设计成使用包括变形驱动程序114的后台进程管理和协调多个应用112的操作。在该方案中,驱动程序安全性平台116可以配置成与后台管理器122交互以利用OS的集成功能性来支持应用的后台操作。

[0026] 事件代理124表示生成事件来代理驱动程序与系统服务之间的交互以执行关于变形驱动程序和对应应用容器120所启用的指定任务的功能性。特别地,事件代理124可以操作来产生使得执行由后台管理器122调度的任务的事件。在一些实现方案中,这涉及生成适当呼叫以编码进入点,诸如系统应用编程接口(API)或其它适当接口。事件代理124还可以配置成生成事件以代理与计算设备所暴露的用户接口元件(诸如与OS相关联的显示屏和UI、浏览器、VPN客户端应用和/或其它应用)的交互。以所描述的方式“沙箱化”的变形驱动程序可以受约束以使其不能直接操控UI组件和元件。因此,事件代理124可以作为代理项操作,其使得能够代表变形驱动程序114与UI组件进行交互和协调而没有过度的安全性暴露。

[0027] 如图1中进一步图示的,服务提供商106可以配置成管理和提供对应用商店服务126的客户端访问。应用商店服务126表示可以由服务提供商126通过网络104向客户端提供的资源108中的特定一个。应用商店服务126被配置成使得用户能够对在线应用数据库(例如应用市场)进行访问以浏览、选择、购买和/或下载应用。可以通过应用商店服务126使来自各种开发者的应用对客户端可用。

[0028] 依照本文讨论的技术,应用商店服务126可以选择为用于经由驱动程序安全性平台116控制的变形驱动程序114的分发机构。由于使用经由驱动程序安全性平台116而针对驱动程序实例化的(多个)受约束执行环境所获得的附加安全性措施、约束和性能增益,使得这成为可能。在一个方案中,变形驱动程序114可以被包装以用于以与驱动程序安全性平台116相关联的指定已知格式包括在应用市场中。包含指定格式的对应变形驱动程序114的驱动程序包装128可以经由应用商店服务126而使得可用并且可以通过网络104下载以用于由客户端设备102使用,如图1中所表示的。在一个方案中,驱动程序包装128可以采用与应用商店服务126相关联的专有格式(诸如.appx或另一相当的商店特定或提供商特定格式)。此外或者可替换地,包装可以配置成包含标识符、密钥、代码、文件扩展、或者促进将驱动程序包装128识别为包含由驱动程序安全性平台116支持的变形驱动程序的其它适当标识数据。

[0029] 在安装时,驱动程序安全性平台116可以基于利用包装所包括的适当识别数据而通过驱动程序管理器模块118的方式或者以其它方式将驱动程序包装128和/或格式识别为与驱动程序安全性平台116相关联。因此,响应于以所识别的格式的驱动程序包装128的安装,驱动程序管理器模块118可以操作来向平台登记对应的变形驱动程序并且由此实行针对通过驱动程序安全性平台116所启用的驱动程序的约束和附加安全性。

[0030] 相对于图2中所描绘(一般地在200处)的示例使用场景讨论关于受约束变形驱动程序平台的这些和其它方面的附加细节。在该示例中,应用容器120或沙箱被图示为包含虚拟私人网络(VPN)驱动程序202,尽管还构想如本文所指出的其它驱动程序、插件和代码。应用容器120配置成使得能够实现用于所包含的驱动程序的有限任务集,所述驱动程序在该情况下恰巧为VPN驱动程序202。在该VPN场景中,所启用的任务可以约束成与通过网络和

VPN服务器的VPN通信相关联的连接、断开、封装和/或解封装操作。其它类型的驱动程序和代码可以与平台所启用的对应操作相关联以支持其核心功能性,同时最小化访问和性能影响。可准许的任务集可以与每一个类型的驱动程序和/或在单独的逐驱动程序的基础上相关联。如所指出的,应用容器120有效地将驱动程序约束到通过平台针对驱动程序所启用的特定任务集。

[0031] 图2还表示各种组件在用户模式和内核模式之间的划分。一般地,计算设备的处理系统可以取决于所执行的代码的类型而在用户模式和内核模式之间切换。典型地,桌上型应用在用户模式下运行,并且核心操作系统组件在内核模式下运行。在内核模式下操作的驱动程序/代码可以共享共同存储器分配并且不与其它驱动程序/代码或OS资源隔离。如所指出的,变形驱动程序传统地实现为内核模式驱动程序,其具有对OS的基本上不受约束的访问和引起系统范围崩溃的潜力。但是,依照本文描述的技术,变形驱动程序可以替代地经由适当的受约束执行环境而在用户模式下执行。

[0032] 特别地,应用容器120和包含在其中的VPN驱动程序202在图2中被描绘为在用户模式下执行。VPN驱动程序202可以经由驱动程序接口204与内核206所提供的操作系统110的核心服务和功能性交互。内核206一般实现底层硬件208的软件抽象以促进通过软件组件对硬件的操作。驱动程序接口204可以表示一个或多个应用编程接口(API)或者与操作系统110相关联的其它适当接口,由内核206表示的核心服务和功能性通过该接口而使得对于在用户模式下执行的代码可访问。

[0033] 在特定示例中,核心服务和功能性包括至少VPN服务210以促进VPN通信。VPN服务210可以对应于针对VPN驱动程序202所启用的任务集,其可以配置成经由驱动程序接口所支持的适当呼叫和回叫而调用服务。驱动程序接口204可以在应用容器120内实例化为所描绘的那样或者实例化为独立式组件。由驱动程序接口204表示的功能性还可以在用户模式组件与在内核模式下操作的内核206的对应组件之间划分。经由驱动程序接口204的交互可以经由驱动程序安全性平台、经由驱动程序管理器模块118或者以其它方式控制。

[0034] 在操作中,后台管理器122可以设立用于应用容器120的后台进程211并且使用后台进程211管理VPN驱动程序202的执行。任务可以在平台的引导下调度。一般地,这在用户的明确指引下(例如用户选择或用户同意)或者依照基于用户输入所建立的简档而发生。简档可以限定对应用容器120中的VPN驱动程序202可用的任务集并且相应地设定相关联的特权和访问权限。平台然后操作来使用应用容器120施行简档所指示的约束。简档例如可以指示用户同意在每一次执行特定操作时(例如操作的每一个实例)被获得,在该情况下可以生成用于用户同意的通知和提示以选择性地控制特定操作。简档还可以指示全局或默认用户同意以允许一些所选操作自动发生而没有针对每一个实例的同意的进一步指示。在没有用户同意的情况下,应用容器120操作来防止驱动程序执行受约束的任务。换言之,受约束执行环境配置成依赖于用户同意而选择性地启用变形驱动程序的操作。因而,为用户提供针对经由平台所管理的变形驱动程序114所启用的交互范围的终极控制。

[0035] 图2进一步图示了应用容器120与事件代理124的交互。特别地,事件代理124可以操作来代表VPN驱动程序产生VPN事件212和/或用户接口事件214,VPN驱动程序使得执行经由后台管理器122所调度的任务。例如,VPN事件212可以包括定向到驱动程序接口204以调用VPN服务210的呼叫。附加地,事件代理124可以产生用户接口事件214以操控用于操作系

统110、应用112或其它UI的UI组件。这可以包括但不限于输出通知以获得针对适当情况下的所调度任务的用户同意。

[0036] 这样,可以实现受约束执行环境来控制VPN驱动程序202和/或传统地在内核模式下执行的其它变形驱动程序的操作。这样的驱动程序一般接受特定形式的输入数据216并且处理该数据来产生如图2中所表示的不同形式的经变换的数据218。例如,图2的VPN驱动程序202可以操作来将数据分组封装成用于到企业服务器的安全VPN通信的专有格式。VPN驱动程序202还可以配置成将从企业服务器所获得的专有格式的数据分组解封装成由客户端设备102可消耗的数据。其他类型的驱动程序被配置成执行与驱动程序所意图的功能性绑定的相当的数据变换。例如,打印机驱动程序转换数据以用于打印,网络接口卡驱动程序操作来分组和去分组数据,抗病毒驱动程序可以扫描文件并且将文件与已知签名相比较等等。

[0037] 因而,可以提供驱动程序安全性平台,其使得能够实现被配置成针对VPN驱动程序以及其它变形驱动程序施行各种约束的受约束执行环境。平台可以实现为暴露接口的操作系统组件,驱动程序可以通过该接口向平台登记并且被调用以执行由平台支持和/或启用的任务。变形驱动程序可以经由平台在用户模式下、在每用户的基础上并且在约束变形驱动程序的特权和系统访问的“沙箱”应用容器内执行。

[0038] 已经考虑到示例操作环境的前述讨论,现在考虑关于相对于以下示例过程所描述的受约束变形驱动程序平台的技术的细节。

[0039] 示例过程

[0040] 该部分参照图3-5的示例过程讨论受约束变形驱动程序平台的技术的细节。在以下讨论的部分中,可以参照其中可以实现各种方面的图1的示例操作环境。以下描述的每一个过程的方面可以在硬件、固件或软件或其组合中实现。过程被示出为一组框,其指定由一个或多个设备执行的操作并且未必限于所示出的用于由相应框执行操作的顺序。在至少一些实现方案中,过程可以由适当配置的计算设备执行,所述计算设备诸如图1的示例客户端设备102,其包括或者利用驱动程序安全性平台116或相当的功能性。

[0041] 图3是描绘出用于采用受约束执行环境以供变形驱动程序的执行的示例过程300的流程图。向驱动程序安全性平台登记变形驱动程序(框302)。例如,驱动程序管理器模块118可以配置成识别以指定格式包装的驱动程序。包装例如可以包括合适的标识符以向驱动程序安全性平台指示驱动程序被设计用于该平台并且要经由该平台来控制。因而,在安装驱动程序之后,驱动程序管理器模块118检测包装和/或标识符并且登记驱动程序以用于经由系统处置。随后,通过具有对应约束的驱动程序安全性平台管理驱动程序的操作。

[0042] 针对变形驱动程序实例化受约束执行环境(框304)并且调用变形驱动程序以执行一个或多个任务(框306)。然后,经由受约束执行环境执行变形驱动程序以执行任务(框308)。在此,驱动管理器模块118可以在执行驱动程序时建立用于特定驱动程序的受约束执行环境。受约束执行环境可以包括如之前讨论的应用容器120(例如“沙箱”)。受约束执行环境在执行期间所应用的变形驱动程序的操作上强加各种约束。例如,受约束执行环境隔离对应的驱动程序以约束对系统资源的访问。特别地,访问可以约束到与驱动程序的功能性相关联的指定任务集。驱动程序由系统调用以执行任务并且可能不能够自己发起任务。例如,受约束执行环境可以防止驱动程序自己创建对象。替代地,用于特定任务的对象由平台

创建并且传递到受约束执行环境以由驱动程序使用。

[0043] 驱动程序还可以经由受约束执行环境作为后台任务而执行并且可以分派有传达有限特权和访问权限的特权令牌。与受约束执行环境相关联的后台进程可以在没有平台和/或执行工作的用户的具体指引的情况下暂停。附加地,受约束执行环境拒绝不与驱动程序安全性平台相关联的应用和其它组件对变形驱动程序的访问以防止对驱动的篡改和/或劫持。

[0044] 附加地,驱动程序如之前所讨论的那样在用户模式下执行并且还可以在每用户的基础上安装。因此,不同用户可以安装不同驱动程序以用于在同一客户端设备上使用和/或用于相同目的。这可以促进依照用户偏好对不同硬件或不同软件版本的使用。由于驱动程序在用户模式下被沙箱化,因此一个驱动程序的安装和卸装不会影响其他用户的体验。参照例如VPN技术,同一设备的不同用户可以采用不同VPN驱动程序来连接到不同雇主的相应VPN服务器而不会不利地相互影响。

[0045] 另外,使用所描述的技术实现的驱动程序可以容易地并且完整地移除而不会留下可能导致非稳定性或非意图行为的残余文件、数据、设置和性质。这是可能的,因为受约束执行环境提供存储装置的单独化分配以供相应驱动程序的排他性使用。驱动程序可以不具有对在别处写入数据或设置的访问。因此,通过删除与驱动程序相关联的所分配存储装置中包含的数据而容易地达成清除驱动程序和状态。

[0046] 图4是描绘出其中控制变形驱动程序的操作的示例过程的流程图。识别变形驱动程序与驱动程序安全性平台的关联(框402)。例如,可以如之前所讨论的那样向驱动程序安全性平台116登记变形驱动程序。驱动程序的操作随后通过驱动程序安全性平台116以受控制的方式处置。关联可以在安装时基于驱动程序以指定格式或包装进行配置而建立。如所指出的,.appx格式是可以用于使驱动程序与平台相关联的一种适当格式,尽管还构想其它文件类型、名称和标识信息。

[0047] 确定针对变形驱动程序所启用的任务集(框404)。所启用的任务集可以基于作为特定类型驱动程序(例如VPN对比打印机对比抗病毒程序等)的驱动程序的标识和/或每一个特定驱动程序的单独标识(例如区分与不同制造商/提供商相关联的相同类型的驱动程序)。平台因此可以针对每一个单独的驱动程序和/或在驱动程序类型的基础上支持和启用指定任务集。任务集可以是操作系统暴露的并且使得驱动程序履行的特定订约。因而,在标识给定驱动程序之后,平台知晓驱动程序能够执行的任务种类,启用那些任务,并且可以防止驱动程序执行其它任务。

[0048] 特别地,建立将变形驱动程序的操作约束到所启用的任务集的执行环境(框406)。例如,受约束执行环境可以以之前所描述的方式建立。受约束执行环境启用某些任务或订约,但是以其它方式防止驱动程序访问操作系统资源或执行没有由驱动程序安全性平台明确授权执行的任务。

[0049] 然后,通过执行环境控制变形驱动程序的操作,包括调用变形驱动程序以执行所启用的任务集并且在没有由驱动程序安全性平台进行的调用的情况下暂停变形驱动程序(框408)。再次,经由受约束执行环境执行的变形驱动程序由平台调用而不是能够自己发起任务。驱动程序可以在其没有被主动地调用以执行任务时保持在暂停状态中。这可以通过后台进程211的方式发生,该后台进程211被解暂停以使得能够工作并且然后在完成工作后

返回到暂停状态。因而,由驱动程序执行的工作在平台和/或用户的指引下发生。驱动程序一般不能够执行除由平台针对驱动程序所指定的那些之外的任务。驱动程序也不能够操作,除非由平台要求,这防止驱动程序自由地消耗系统资源和电池寿命。

[0050] 图5是描绘出其中经由在线应用商店分发变形驱动程序的示例过程的流程图。限定驱动程序安全性平台可识别的应用包装格式以用于将插件包括在在线应用商店中(框502)。例如,诸如变形驱动程序114和来自第三方提供商的其它代码之类的插件可以编码在采用针对驱动程序安全性平台116所限定的指定格式的驱动程序包装128中。开发者可以使用指定格式来利用驱动程序安全性平台116并且信号通知平台以相应地经由平台处置其代码/驱动程序。驱动程序包装128可以配置成促进包括在应用市场中,诸如使用.appx格式或其它指定格式经由应用商店服务126。

[0051] 启用访问以经由在线应用商店下载插件(框504)。然后,以限定格式将插件分发给客户端,以有效地使客户端向驱动程序安全性平台登记插件并且实现受约束执行环境来控制插件的操作(框506)。将变形驱动程序114作为供应品包括在应用市场中由于经由驱动程序安全性平台116可获得的附加安全性措施、约束和性能增益而是部分可行的,其使得更可能的是用户将从在线商店下载和安装这样的驱动程序。因而,包含指定格式的对应变形驱动程序114的驱动程序包装128可以经由应用商店服务126而使得可用。驱动程序包装128可以包含适当的标识数据以促进驱动程序安全性平台116对包装的识别。驱动程序安全性平台116配置成在安装时识别驱动程序包装128并且执行如本文所描述的操作以登记驱动程序,创建受约束执行环境,施行放置在驱动程序上的各种约束等。

[0052] 已经考虑到一些示例过程,现在考虑实现依照一个或多个实施例的各种方面的示例系统和设备的讨论。

[0053] 示例系统和设备

[0054] 图6图示了包括代表可以实现本文描述的各种技术的一个或多个计算系统和/或设备的示例计算设备602的示例系统600。计算设备602可以例如是服务提供商的服务器、与客户端相关联的设备(例如客户端设备)、片上系统和/或任何其它适当的计算设备或计算系统。

[0055] 如所图示的示例计算设备602包括通信耦合到彼此的处理系统604、一个或多个计算机可读介质606和一个或多个I/O接口608。尽管未示出,但是计算设备602还可以包括系统总线或者将各种组件彼此耦合的其它数据和命令传递系统。系统总线可以包括不同总线结构中的任何一个或组合,所述总线结构诸如存储器总线或存储器控制器、外围总线、通用串行总线和/或利用任何各种总线架构的处理器或本地总线。还构想各种其它示例,诸如控制和数据线。

[0056] 处理系统604代表使用硬件执行一个或多个操作的功能性。因此,处理系统604被图示为包括硬件元件610,硬件元件610可以配置成处理器、功能块等。这可以包括作为专用集成电路或使用一个或多个半导体形成的其它逻辑设备的硬件中的实现方案。硬件元件610不受形成它们的材料或其中所采用的处理机制的限制。例如,处理器可以包括(多个)半导体和/或晶体管(例如电子集成电路(IC))。在这样的上下文中,处理器可执行指令可以是电子可执行指令。

[0057] 计算机可读介质606被图示为包括存储器/存储装置612。存储器/存储装置612表

示与一个或多个计算机可读介质相关联的存储器/存储容量。存储器/存储装置612可以包括易失性介质(诸如随机存取存储器(RAM))和/或非易失性介质(诸如只读存储器(ROM)、闪存存储器、光盘、磁盘等)。存储器/存储装置612可以包括固定介质(例如RAM、ROM、固定硬盘驱动器等)以及可移除介质(例如闪存存储器、可移除硬盘驱动器、光盘等)。计算机可读介质606可以以各种其它方式配置,如下文进一步描述的。

[0058] (多个)输入/输出接口608代表允许用户向计算设备602录入命令和信息,并且还允许使用各种输入/输出设备将信息呈现给用户和/或其它组件或设备的功能性。输入设备的示例包括键盘、光标控制设备(例如鼠标)、用于语音操作的麦克风、扫描仪、触摸功能性(例如配置成检测物理触摸的电容或其它传感器)、相机(例如其可以采用诸如红外频率之类的可见或不可见波长以检测不涉及作为手势的触摸的运动)等。输出设备的示例包括显示设备(例如监视器或投影仪)、扬声器、打印机、网络卡、触觉响应设备等。因而,计算设备602可以以如下文进一步描述的各种方式配置以支持用户交互。

[0059] 本文在软件、硬件元件或程序模块的一般上下文中描述各种技术。一般地,这样的模块包括例程、程序、对象、元件、组件、数据结构等,其执行特定任务或实现特定抽象数据类型。如本文使用的术语“模块”、“功能性”和“组件”一般表示软件、固件、硬件或其组合。本文描述的技术的特征是独立于平台的,这意味着技术可以在具有各种处理器的各种商用计算平台上实现。

[0060] 所描述的模块和技术的实现方案可以存储在某种形式的计算机可读介质上或者跨某种形式的计算机可读介质传输。计算机可读介质可以包括可以由计算设备602访问的各种介质。作为示例而非限制,计算机可读介质可以包括“计算机可读存储介质”和“通信介质”。

[0061] “计算机可读存储介质”是指相对于单纯信号传输、载波或信号本身使得能够存储信息的介质和/或设备。因而,计算机可读存储介质不包括信号承载介质或信号本身。计算机可读存储介质包括诸如易失性和非易失性、可移除和不可移除介质之类的硬件和/或在适用于存储诸如计算机可读指令、数据结构、程序模块、逻辑元件/电路或其它数据之类的信息的方法或技术中实现的存储设备。计算机可读存储介质的示例可以包括但不限于RAM、ROM、EEPROM、闪存存储器或其它存储器技术、CD-ROM、数字多用盘(DVD)或其它光学存储装置、硬盘、盒式磁带、磁带、磁盘存储装置或其它磁存储设备、或者适用于存储期望的信息并且可以由计算机访问的其它存储设备、有形介质或制造品。

[0062] “通信介质”是指配置成诸如经由网络将指令传输到计算设备602的硬件的信号承载介质。通信介质典型地可以在诸如载波、数据信号或其它输运机制之类的调制数据信号中体现计算机可读指令、数据结构、程序模块或其它数据。通信介质还包括任何信息递送介质。术语“调制数据信号”是指这样的信号:使其特性中的一个或多个以这样的方式设定或改变以将信息编码在信号中。作为示例而非限制,通信介质包括诸如有线网络或直接有线连接之类的有线介质,以及诸如声学、RF、红外和其它无线介质之类的无线介质。

[0063] 如之前描述的,硬件元件610和计算机可读介质606代表以可以在一些实施例中用于实现本文描述的技术的至少一些方面的硬件形式实现的指令、模块、可编程设备逻辑和/或固定设备逻辑。硬件元件可以包括集成电路或片上系统的组件、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、复杂可编程逻辑器件(CPLD)以及硅或其它硬件设备中的其它实现

方案。在该上下文中，硬件元件可以操作为执行由指令、模块和/或硬件元件所体现的逻辑限定的程序任务的处理设备以及用于存储指令以供执行的硬件设备，例如之前所描述的计算机可读介质。

[0064] 以上的组合也可以用于实现本文描述的各种技术和模块。因此，软件、硬件或程序模块，包括操作系统110、应用112、驱动程序安全性平台116和其它程序模块，可以实现为体现在某种形式的计算机可读存储介质上和/或由一个或多个硬件元件610体现的一个或多个指令和/或逻辑。计算设备602可以配置成实现对应于软件和/或硬件模块的特定指令和/或功能。因此，将模块实现为计算设备602作为软件可执行的模块可以至少部分地在硬件中实现，例如通过使用计算机可读介质和/或处理系统的硬件元件610。指令和/或功能可以由一个或多个制造品可执行/可操作（例如一个或多个计算设备602和/或处理系统604）以实现本文描述的技术、模块和示例。

[0065] 如图6中进一步图示的，示例系统600使得能够在个人计算机(PC)、电视设备和/或移动设备上运行应用时实现用于无缝用户体验的普适环境。当在利用应用、播放视频游戏、观看视频等时从一个设备转换到下一个时，服务和应用在所有三个环境中基本上类似地运行以用于共同的用户体验。

[0066] 在示例系统600中，多个设备通过中央计算设备互连。中央计算设备可以在多个设备本地或者可以远离多个设备定位。在一个实施例中，中央计算设备可以通过网络、互联网或其它数据通信链路连接到多个设备的一个或多个服务器计算机的云。

[0067] 在一个实施例中，该互连架构使得能够跨多个设备递送功能性以向多个设备的用户提供共同且无缝的体验。多个设备中的每一个可以具有不同物理要求和能力，并且中央计算设备使用平台来使得能够向设备递送既针对设备定制又是所有设备所共同的体验。在一个实施例中，创建目标设备类并且针对通用类设备定制体验。设备类可以由设备的物理特征、使用类型或其它共同特性限定。

[0068] 在各种实现方案中，计算设备602可以采取各种不同配置，诸如用于计算机614、移动装置616和电视618用途。这些配置中的每一个包括可以具有大体不同构造和能力的设备，并且因而计算设备602可以根据不同设备类中的一个或多个来配置。例如，计算设备602可以实现为计算机614类设备，其包括个人计算机、桌上型计算机、多屏幕计算机、膝上型计算机、上网本等。

[0069] 计算设备602还可以实现为移动装置616类设备，其包括移动设备，诸如移动电话、便携式音乐播放器、便携式游戏设备、平板计算机、多屏幕计算机等。计算设备602还可以实现为电视618类设备，其包括具有非正式观看环境中的大体更大屏幕或者连接到所述屏幕的设备。这些设备包括电视、机顶盒、游戏控制台等。

[0070] 本文描述的技术可以由计算设备602的这些各种配置支持并且不限于本文描述的技术的具体示例。这通过在计算设备602上包括驱动程序安全性平台116来图示。驱动程序安全性平台116和其它模块的功能性还可以全部或部分通过使用分布式系统来实现，诸如通过经由平台622的“云”620，如下文描述的。

[0071] 云620包括和/或代表用于资源624的平台622。平台622抽象云620的硬件（例如服务器）和软件资源的底层功能性。资源624可以包括在远离计算设备602的服务器上执行计算机处理时可以利用的应用和/或数据。资源624还可以包括通过互联网和/或通过诸如蜂

窝或Wi-Fi网络之类的订户网络所提供的服务。

[0072] 平台622可以抽象资源和功能以将计算设备602与其它计算设备连接。平台622还可以服务于抽象资源的规模以向遭遇的对于经由平台622所实现的资源624的需求提供对应水平的规模。相应地,在互连设备实施例中,本文描述的功能性的实现可以遍及系统600分布。例如,功能性可以部分地实现在计算设备602上以及经由抽象云620的功能性的平台622而实现。

[0073] 结论

[0074] 尽管已经以具体到结构特征和/或方法动作的语言描述了本发明,但是要理解到,在所附权利要求中限定的本发明未必限于所描述的具体特征或动作。相反,具体特征和动作是作为实现所要求保护的发明的示例形式公开的。

100

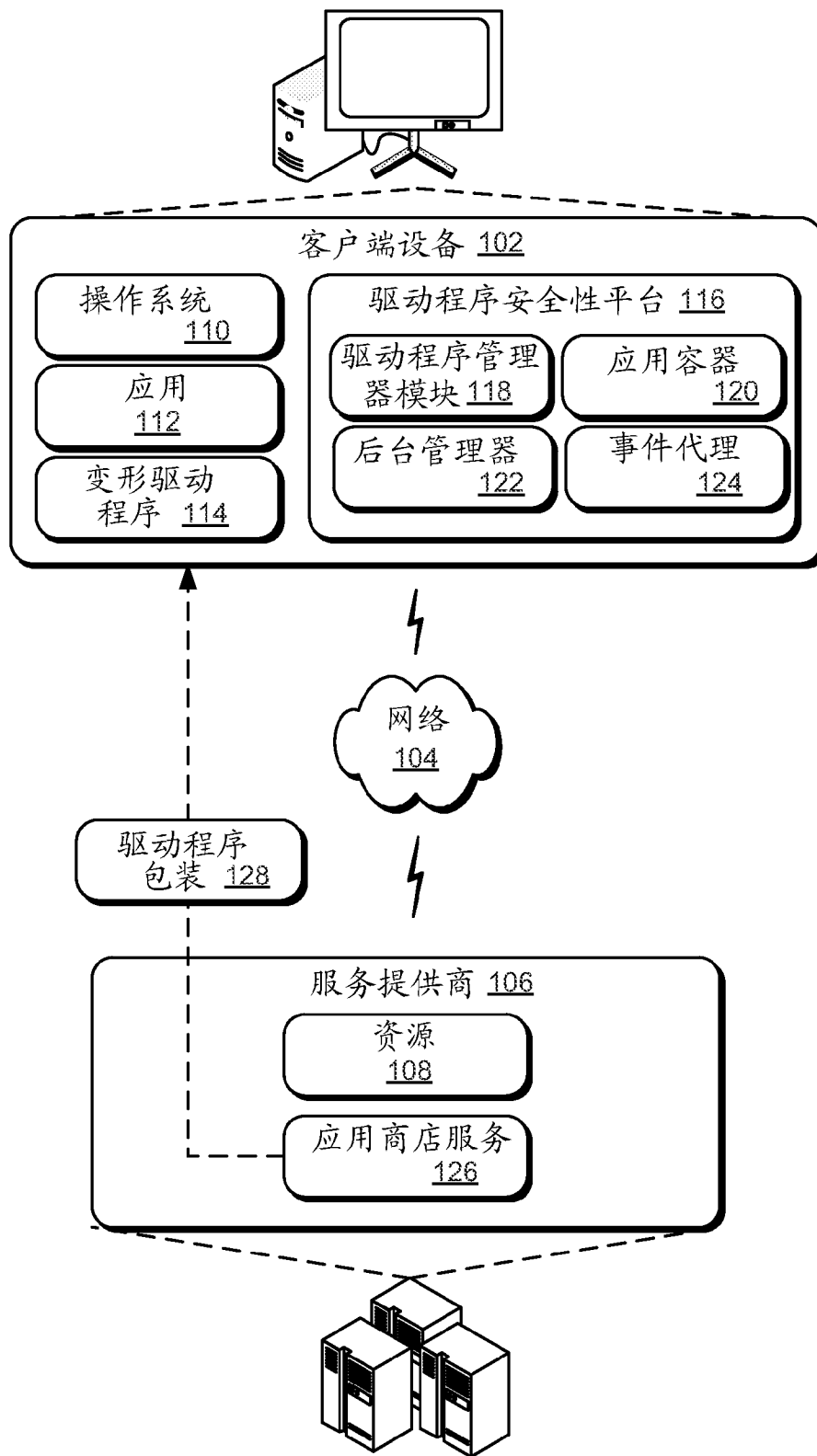


图 1

200

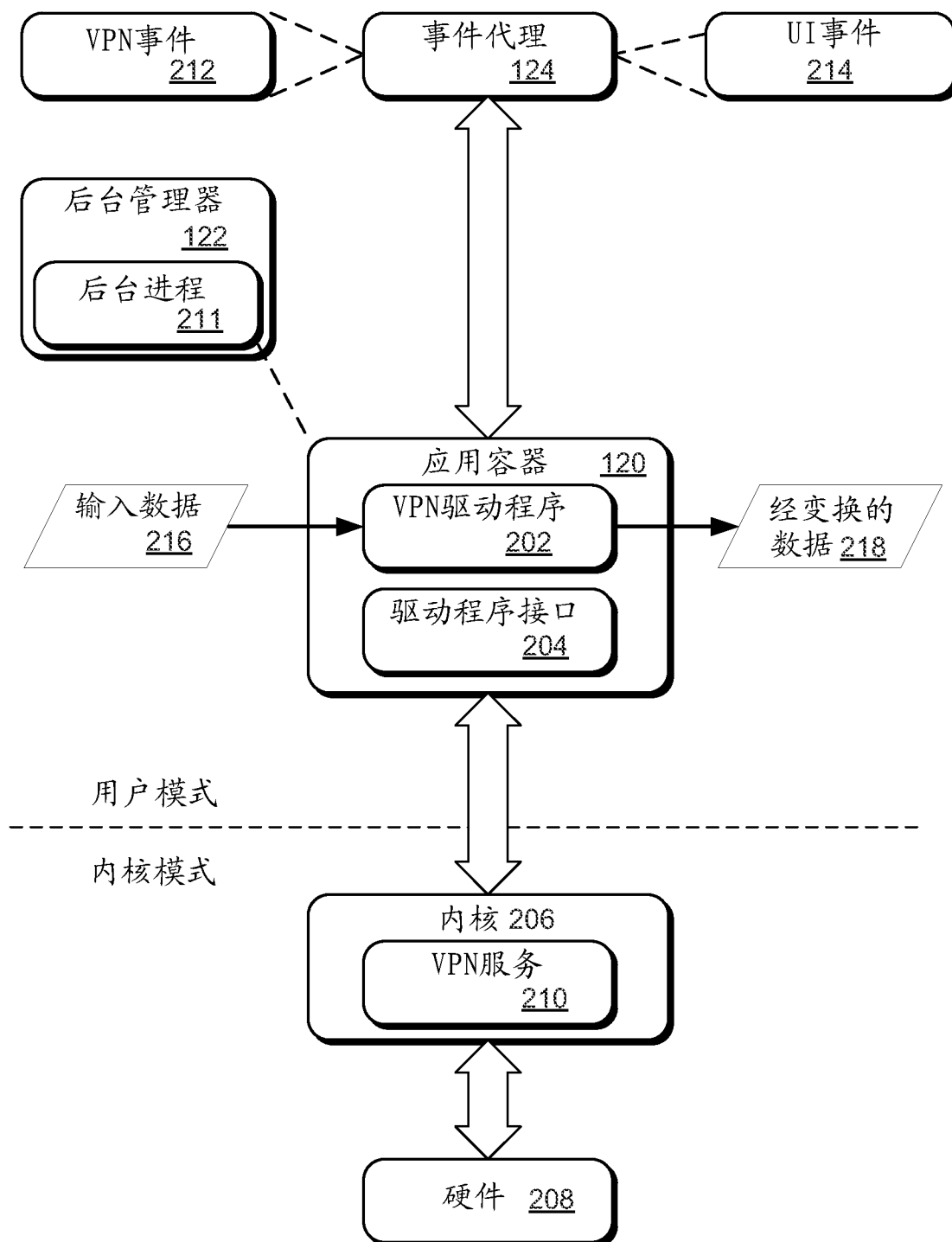


图 2

300

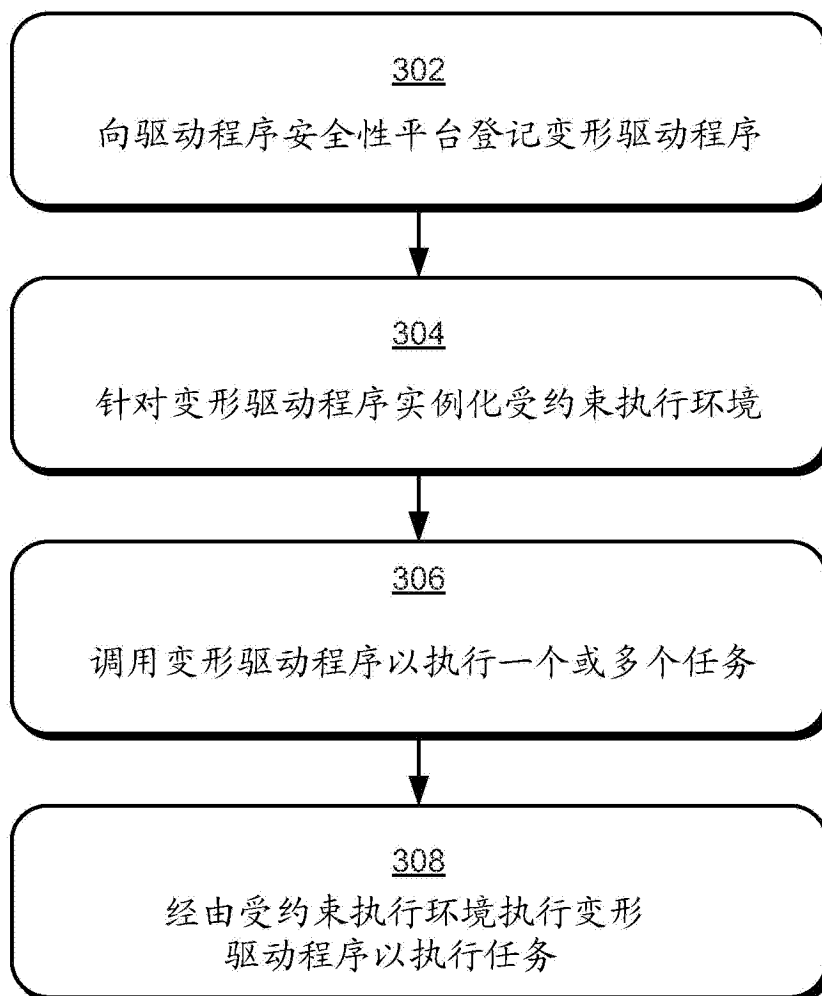


图 3

400

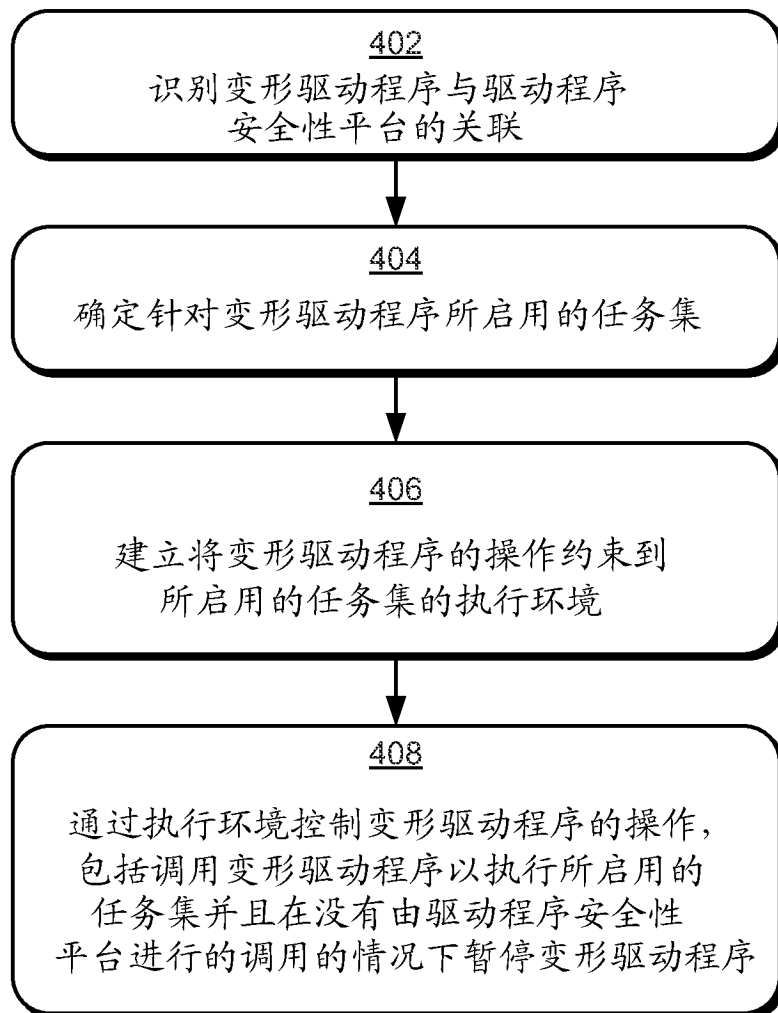


图 4

500

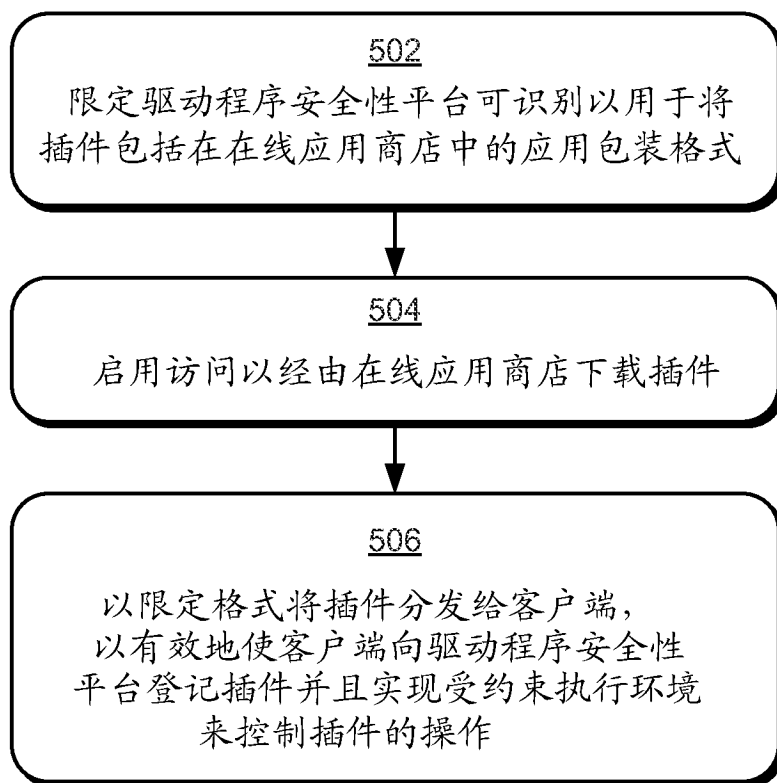


图 5

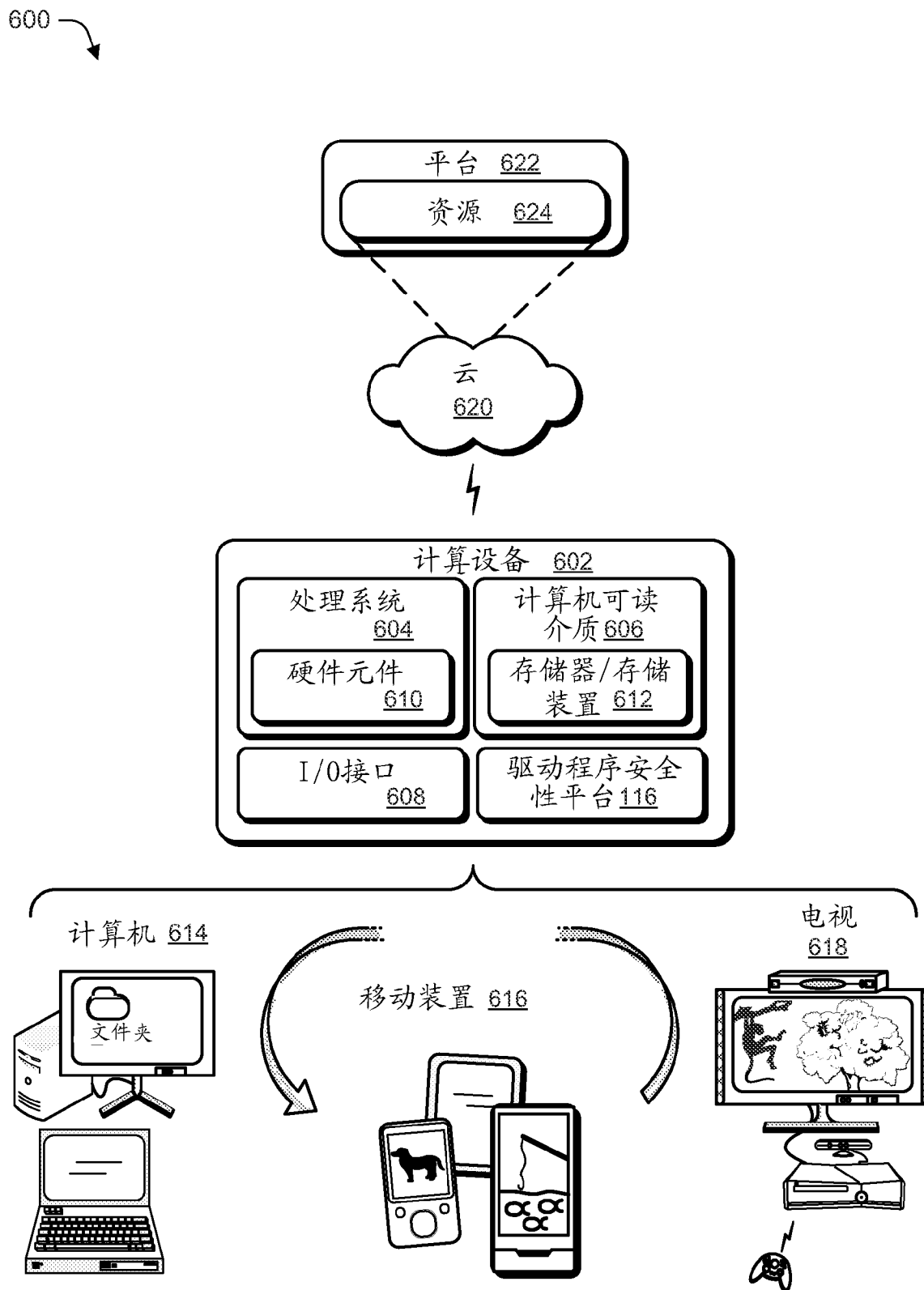


图 6