



(12)发明专利

(10)授权公告号 CN 106415577 B

(45)授权公告日 2020.06.16

(21)申请号 201580015751.3

(22)申请日 2015.03.03

(65)同一申请的已公布的文献号

申请公布号 CN 106415577 A

(43)申请公布日 2017.02.15

(30)优先权数据

14/231409 2014.03.31 US

(85)PCT国际申请进入国家阶段日

2016.09.22

(86)PCT国际申请的申请数据

PCT/US2015/018412 2015.03.03

(87)PCT国际申请的公布数据

W02015/153037 EN 2015.10.08

(73)专利权人 CA公司

地址 美国加利福尼亚州

(72)发明人 S·佩雷拉

(74)专利代理机构 北京律盟知识产权代理有限公司 11287

代理人 章蕾

(51)Int.Cl.

G06F 21/55(2006.01)

(56)对比文件

CN 103064770 A, 2013.04.24,

CN 103632088 A, 2014.03.12,

US 8205257 B1, 2012.06.19,

US 2014068774 A1, 2014.03.06,

审查员 郑婷

权利要求书2页 说明书8页 附图7页

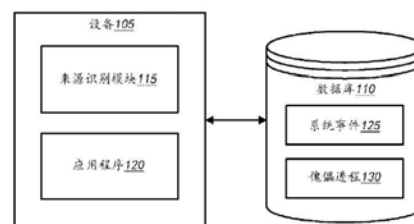
(54)发明名称

用于识别可疑事件来源的系统和方法

(57)摘要

本发明描述了一种用于识别可疑事件来源的计算机实施的方法。在一个实施例中,可在数据库中登记系统事件。可检测与第一进程相关联的可疑事件,并且可将所述第一进程识别为多个潜在傀儡进程中的一个。可查询所述数据库中已登记的系统事件以识别第二进程,其中所述第二进程被检测为启动所述第一进程。

100



1. 一种用于识别可疑事件来源的计算机实施的方法,包括:
在数据库中登记系统事件,其中所述系统事件包括检测到的进程启动事件;
检测第一进程被启动,所述第一进程是所述检测到的进程启动事件中的一个;
检测与所述第一进程相关联的可疑事件;
查询所述数据库中的已登记的系统事件;
分析所述查询的结果,所述查询的结果指示第二进程和所述第一进程之间的关联,所述第二进程不同于所述第一进程;
至少部分基于对所述查询的结果的分析,确定所述第二进程启动所述第一进程;以及
至少部分基于确定所述第二进程启动所述第一进程,将所述第一进程识别为多个潜在傀儡进程中的一个。
2. 根据权利要求1所述的方法,还包括:
生成通知,所述通知将所述第二进程识别为所述可疑事件的来源。
3. 根据权利要求1所述的方法,还包括:
确定所述第二进程在发起所述第一进程后关闭。
4. 根据权利要求1所述的方法,还包括:
在识别发起所述第一进程的所述第二进程后,将所述第一进程识别为傀儡进程。
5. 根据权利要求1所述的方法,其中经由所述第二进程直接在命令行接口上提供的指令发起所述第一进程。
6. 根据权利要求1所述的方法,其中经由来自至少一个文件的指令发起所述第一进程,到达所述至少一个文件的路径由所述第二进程在命令行接口上提供。
7. 根据权利要求1所述的方法,还包括:
维护所述潜在傀儡进程的列表;
经由已登记的系统事件检测新的傀儡进程,其中所述新的傀儡进程是所述检测到的进程启动事件中的一个;以及
将所述新的傀儡进程添加到所述潜在傀儡进程的列表。
8. 根据权利要求7所述的方法,其中所述潜在傀儡进程的列表包括具有命令行接口的至少一个进程。
9. 根据权利要求8所述的方法,其中所述潜在傀儡进程的列表包括cmd.exe、rundll.exe、rundll32.exe、regsvr32.exe、dllhost.exe、regedit.exe、taskhost.exe、cscript、wscript、vbscript、perlscript、bash、ldconfig、terminal.app和x-code.app中的至少一者。
10. 根据权利要求1所述的方法,其中经由内核模式驱动程序登记所述系统事件。
11. 一种被配置成识别可疑事件来源的计算设备,包括:
处理器;
与所述处理器进行电子通信的存储器;
存储在所述存储器中的指令,所述指令可由所述处理器执行以:
在数据库中登记系统事件,其中所述系统事件包括检测到的进程启动事件;
检测第一进程被启动,所述第一进程是所述检测到的进程启动事件中的一个;
检测与所述第一进程相关联的可疑事件;

查询所述数据库中的已登记的系统事件；

分析所述查询的结果，所述查询的结果指示第二进程和所述第一进程之间的关联，所述第二进程不同于所述第一进程；

至少部分基于对所述查询的结果的分析，确定所述第二进程启动所述第一进程；以及

至少部分基于确定所述第二进程启动所述第一进程，将所述第一进程识别为多个潜在傀儡进程中的一个。

12. 根据权利要求11所述的计算设备，其中所述指令可由所述处理器执行以：

生成通知，所述通知将所述第二进程识别为所述可疑事件的来源。

13. 根据权利要求11所述的计算设备，其中所述指令可由所述处理器执行以：

确定所述第二进程在发起所述第一进程后关闭。

14. 根据权利要求11所述的计算设备，其中所述指令可由所述处理器执行以：

在识别发起所述第一进程的所述第二进程后，将所述第一进程识别为傀儡进程。

15. 一种用于由处理器识别可疑事件来源的非暂态性计算机可读介质，所述非暂态性计算机可读介质上存储有指令，所述指令可由所述处理器执行以：

在数据库中登记系统事件，其中所述系统事件包括检测到的进程启动事件；

检测第一进程被启动，所述第一进程是所述检测到的进程启动事件中的一个；

检测与所述第一进程相关联的可疑事件；

查询所述数据库中的已登记的系统事件；

分析所述查询的结果，所述查询的结果指示第二进程和所述第一进程之间的关联，所述第二进程不同于所述第一进程；

至少部分基于对所述查询的结果的分析，确定所述第二进程启动所述第一进程；以及

至少部分基于确定所述第二进程启动所述第一进程，将所述第一进程识别为多个潜在傀儡进程的一个。

用于识别可疑事件来源的系统和方法

背景技术

[0001] 计算机系统和计算机相关技术的使用继续高速增长。计算机系统使用的这种增长影响了计算机相关技术的进展。计算机系统的确已日益成为商业领域和个人消费者活动密不可分的一部分。计算机系统可用于执行若干商业、工业和学术活动。包括互联网的计算机网络的使用增长加快了计算机的广泛使用。

[0002] 许多企业使用一个或多个计算机网络来在连接到这些网络的各计算机之间传送和共享数据。雇员的生产力和效率经常需要人与计算机交互。计算机技术的用户继续需要这些技术的效率得到提高。使用并依赖于计算机的任何人总是希望计算机技术的效率得到提升。

[0003] 计算机和移动设备的广泛使用已经导致存在更多的恶意程序,诸如隐匿程序、间谍软件、特洛伊木马等。恶意软件程序设计者不断适应于恶意软件检测技术的进步,这形成了恶意软件技术适应恶意软件检测技术进步的持续循环。

发明内容

[0004] 根据至少一个实施例,描述了一种识别可疑事件来源的计算机实施的方法。在一个实施例中,可在数据库中登记系统事件。在一些情况下,经由内核模式驱动程序登记系统事件。系统事件可包括计算机系统中发生的任何可检测的事件。在一些情况下,系统事件可包括进程启动事件、文件系统事件等。可检测与第一进程相关联的可疑事件,并且可将该第一进程识别为多个潜在傀儡进程中的一个。可查询数据库中已登记的系统事件以识别第二进程,其中该第二进程被检测为启动第一进程。

[0005] 在一些实施例中,可生成通知。该通知可将第二进程识别为可疑事件的来源,并且/或者指示第二进程是潜在的恶意程序。可确定第二进程在启动第一进程后关闭。在识别到启动第一进程的第二进程后,可将第一进程识别为傀儡进程。

[0006] 在一个实施例中,可经由第二进程在命令行接口上直接提供的指令启动第一进程。除此之外或作为另外一种选择,可经由来自至少一个文件的指令启动第一进程,到达所述至少一个文件的路径由第二进程在命令行接口上提供。可例如在数据库中维护潜在傀儡进程的列表。傀儡进程可为可由另一个单独进程配置成执行一组指令的任何进程。可经由已登记的系统事件检测新的傀儡进程,并且可将新的傀儡进程添加到潜在傀儡进程的列表。潜在傀儡进程的列表可包括具有命令行接口的至少一个进程。在一些情况下,潜在傀儡进程的列表可包括cmd.exe、rundll.exe、rundll32.exe、regsvr32.exe、dllhost.exe、regedit.exe、taskhost.exe、cscript、wscript、vbscript、perlscript、bash、ldconfig、terminal.app和x-code.app中的至少一者。

[0007] 还描述了一种被配置成识别可疑事件来源的计算设备。该设备可包括处理器和与处理器电子通信的存储器。存储器可存储能够由处理器执行以进行以下操作的指令:在数据库中登记系统事件,检测与第一进程相关联的可疑事件,将第一进程识别为多个潜在傀儡进程中的一个,以及查询数据库中已登记的系统事件以识别第二进程,该第二进程启动

第一进程。

[0008] 还描述了一种用于识别可疑事件来源的计算机程序产品。计算机程序产品可包括存储指令的非暂态性计算机可读介质。这些指令可由处理器执行以在数据库中登记系统事件,检测与第一进程相关联的可疑事件,将第一进程识别为多个潜在傀儡进程中的一个,以及查询数据库中已登记的系统事件以识别第二进程,该第二进程启动第一进程。

[0009] 来自上述实施例中的任何一者的特征可根据本文所述的一般原理彼此结合地使用。通过结合附图和权利要求书阅读下面的详细描述,将会更充分地理解这些和其他实施例、特征和优点。

附图说明

[0010] 附图示出了多个示例性实施例并且为说明书的一部分。这些附图结合下面的描述展示并且说明本发明的各种原理。

[0011] 图1是示出环境的一个实施例的框图,在该环境中可实施本发明的系统和方法;

[0012] 图2是示出环境的另一个实施例的框图,在该环境中可实施本发明的系统和方法;

[0013] 图3是示出来源识别模块的一个例子的框图;

[0014] 图4是示出用于识别可疑事件来源的方法的一个实施例的流程图;

[0015] 图5是示出用于维护潜在傀儡进程列表的方法的一个实施例的流程图;

[0016] 图6是示出用于识别可疑事件来源的方法的一个实施例的流程图;以及

[0017] 图7示出了适于实施本发明的系统和方法的计算机系统的框图。

[0018] 虽然本文所述的实施例易有各种修改和替代形式,但在附图中以举例的方式示出了特定实施例并且将在本文中详细描述。然而,本文所述的示例性实施例并非旨在限于所公开的特定形式。相反,本发明涵盖落在所附权利要求书范围内的所有修改形式、等同形式和替代形式。

具体实施方式

[0019] 本文所述的系统和方法涉及识别可疑事件的来源。更具体地讲,本文所述的系统和方法涉及检测执行导致可疑事件的潜在恶意指令的傀儡进程,以及识别潜在恶意指令的来源。

[0020] 恶意软件的创建者不断地使其技术进化以适应恶意软件检测系统。当前的基于文件的恶意软件检测系统在着手检测适应性恶意软件(诸如多态性恶意软件)时经常落后于形势。因此,恶意软件检测系统可采用行为学习引擎来识别逃避检测的多态性恶意软件。此类行为引擎可完全基于恶意软件如何运行来检测新的未分类恶意软件。通过观察一个或多个进程的行为,本文所述的系统和方法能够将可疑事件与根进程(例如,恶意进程、恶意可执行等)相关联,此根进程即对所述可疑事件最终负责的进程。例如,假如行为引擎看到对Run或RunOnce注册表项进行写入的试图,则该行为引擎可确定哪个进程最终负责该写命令,从而使得行为引擎对该负责进程采取措施,诸如终止该进程以及/或者删除与该进程相关联的文件。

[0021] 在一些情况下,等到当前系统检测到恶意软件事件时,恶意软件事件的根进程已经终止其自身以便掩盖其曾参与恶意软件事件。恶意软件作者可能采用Windows签署进程

来执行其恶意攻击,因为恶意软件创建者认识到典型的行为引擎被配置成对合法的Windows进程不采取补救措施,因为这样做可能会影响机器的稳定性和/或破坏关键的功能。因此,在一些情况下,恶意软件作者可能使用“傀儡”进程来执行其恶意攻击。傀儡进程可为由主进程用来通过代理完成一个或多个动作的任何次进程。这些进程在运行时不表现任何动作,它们只是退出。次进程执行经由命令行等被告知的任务,因此称为“傀儡”。例如,恶意软件作者可创建恶意进程,即主进程,其被配置成采用Windows进程,诸如cmd.exe或rundll32.exe,即次或傀儡进程。在一些情况下,恶意进程可利用具有命令行接口的进程。恶意进程可直接在命令行上提供恶意指令作为一条或多条命令行指令。除此之外或作为另外一种选择,恶意进程可在文件中提供恶意指令,其中到达该文件的路径在命令行上传递。因此,恶意进程可将Windows进程配置成代表恶意进程执行一条或多条恶意指令。一旦恶意进程将次进程配置成执行恶意指令,恶意进程就可能试图通过关闭来掩盖其踪迹,仅留下次进程可被发现。一旦恶意进程已经用恶意指令配置次进程,恶意进程便关闭,留下次进程执行这些指令,这时恶意进程关闭并且看起来无处可见。因此,在一些实施例,确定第一进程配置第二进程,并且检测到来自次进程的可疑事件可指示可疑事件和启动进程是恶意的。在一些情况下,确定第二进程包括命令行接口可指示可疑事件和启动进程是恶意的。除此之外或作为另外一种选择,确定启动进程在启动傀儡进程后关闭可指示可疑事件和启动进程是恶意的。

[0022] 在一个实施例中,本文所述的系统和方法可保持并维护潜在傀儡进程的列表(例如,通过可执行程序名称、图像名称、进程名称、进程特征、应用程序描述等)。该列表可包括已知为已经在过去用作傀儡进程的那些进程和/或已知为能够用作傀儡进程的那些进程。已知windows傀儡进程的列表可包括一个或多个命令行接口。例如,该列表可包括rundll32.exe、regsvr32.exe、cscript、wscript、dllhost.exe、taskhost.exe、cmd.exe、terminal.app、x-code.app、bash等。

[0023] 在一个例子中,傀儡进程可直接在命令行上接收恶意指令。所述系统和方法可使用内核模式驱动程序跟踪此类进程启动,并且在此类傀儡进程启动后,可识别启动傀儡进程的所识别进程并且使其为由傀儡进程引起的任何可疑事件负责。在一些情况下,傀儡进程可间接经由在命令行上提供的到达文件(例如,.BAT文件、.CMD文件、.BTM文件、.MSC文件、.JOB文件、.IDL文件、.TLB文件、.EXE文件、.DLL文件、.APP文件等)的路径接收指令。该文件可包含另外的指令(例如,rundll32.exe)和/或到达特殊格式指令文件(诸如,在regedit.exe用作傀儡进程的情况下,.REG文件)的路径。在一些情况下,可指示傀儡进程在注册表中查找组件对象模型(COM)对象全局唯一标识符(GUID)以便定位到达可从中检索指令的COM动态链接库(DLL)的路径。此类恶意文件和/或恶意指令可被称为“死亡投放”文件和/或“死亡投放”指令,因为负责的恶意进程“投放”其程序包并且等到程序包被执行的时候已经“死亡”(即,关闭)。恶意软件可创建这些死亡投放文件,并且指示操作系统启动傀儡进程以将这些死亡投放文件用作其执行什么的指令源。因此,本文所述的系统和方法可采用内核模式驱动程序来将死亡投放文件的创建追溯到创建它们的进程。在一些情况下,这些系统和方法可监测所有进程启动。识别到傀儡进程启动后,经由直接/间距接收这些死亡投放文件的命令行,这些系统和方法可识别最终负责死亡投放文件的进程(例如,恶意进程、恶意可执行等)并且相应地采取补救措施。

[0024] 图1是示出环境100的一个实施例的框图,在该环境中可实施本发明的系统和方法。在一些实施例中,本文所述的系统和方法可在单个设备(例如,设备105)上执行。例如,来源识别模块115可位于设备105上。设备105的例子包括移动设备、智能电话、个人计算设备、计算机、服务器等。

[0025] 在一些配置中,设备105可包括来源识别模块115和应用程序120。应用程序120可允许设备105与来源识别模块115介接。在一个例子中,设备105可连接到数据库110。在一个实施例中,数据库110可在设备105内部。在另一个实施例中,数据库110可在设备105外部。在一些配置中,数据库110可包括系统事件125和傀儡进程130。系统事件125可包括进程启动事件、文件系统事件等。在一些情况下,傀儡进程130可包括一个或多个潜在傀儡进程的列表。该列表可包括已知为已经用作傀儡进程的进程和/或有可能用作傀儡进程的进程。例如,进程可由于该进程的一个或多个特征而被认为有可能用作傀儡进程。例如,在一些情况下,包括命令行接口的进程可比不包括命令行接口的进程更有可能用作傀儡进程。

[0026] 来源识别模块115可通过以下方式识别可疑事件的来源:检测可疑事件,确定某一进程是否在潜在傀儡进程的列表(例如,傀儡进程130)上,查询已存储的系统事件125,以及/或者生成关于所检测到的可疑事件的通知。下文论述关于来源识别模块115的进一步的细节。

[0027] 图2是示出环境200的另一个实施例的框图,在该环境中可实施本发明的系统和方法。在一个实施例中,设备105-a可经由网络205与服务器210通信以及/或者访问数据库110-a。数据库110-a可为图1中的数据库110的一个例子。网络205的例子包括云网络、局域网(LAN)、广域网(WAN)、虚拟专用网(VPN)、无线网络(例如,使用802.11)、蜂窝网络(例如,使用3G和/或LTE)等。在一些情况下,网络205包括互联网。

[0028] 在一些配置中,设备105-a可为图1所示的设备105的一个例子。如图所示,设备105-a可不包括来源识别模块115。例如,设备105-a可包括应用程序120,该应用程序允许设备105-a与位于服务器210上的来源识别模块115介接。因此,设备105-a可被配置成经由应用程序120、网络205和服务器210执行欺诈检测模块115的一个或多个功能。在一些实施例中,设备105-a和服务器210两者都可包括来源识别模块115,其中在设备105-a和服务器210上单独地和/或同时地执行来源识别模块115的功能的至少一部分。

[0029] 在一些实施例中,服务器210可包括来源识别模块115,并且可连接到数据库110-a。例如,来源识别模块115可经由网络205和服务器210访问数据库110-a中的系统事件125和/或傀儡进程130。数据库110-a可在服务器210内部或外部。

[0030] 在一些配置中,应用程序120可使得设备105-a与来源识别模块115介接以:通过搜索预先过滤的网页来识别潜在欺诈活动,在预先过滤的网页当中检测样式,并且将用户输入与所检测到的样式进行比较以便向用户实时通知潜在欺诈。因此,经由来源识别模块115,应用程序120可识别潜在欺诈活动并警告用户此类活动。

[0031] 图3是示出来源识别模块115-a的一个例子的框图。来源识别模块115-a可为图1和/或图2所示的来源识别模块115的一个例子。如图所示,来源识别模块115-a可包括监测模块305、比较模块310、识别模块315、分类模块320和通知模块325。

[0032] 如上所述,恶意进程可以另外的、无害的合法进程为目标,使其代表恶意进程执行一个或多个动作。在一些情况下,恶意进程可以受信任的进程(诸如cmd.exe)为目标,并将

受信任的进程配置成代表其执行恶意任务。恶意进程可将目标进程配置成执行恶意任务以躲避检测,从而允许恶意进程在配置目标进程之后关闭或终止其自身。因此,恶意进程试图掩盖其在执行恶意指令过程中的任何参与,从而允许恶意进程在常规的恶意软件检测系统下不被检测到。因而,恶意进程可被称为启动进程或来源进程,即最终负责由目标进程引起的任何恶意事件的进程。目标进程可被称为傀儡进程,因为它是最终控制一切的恶意进程。因此,为了防止启动进程不被检测到,来源识别模块115-a可被配置成根据本文所述的系统和方法来识别可疑事件的来源。

[0033] 在一个实施例中,监测模块305可在数据库中登记系统事件。在一些情况下,经由内核模式驱动程序登记系统事件。监测模块305可被配置成检测与第一进程相关联的可疑事件。比较模块310可确定第一进程(例如,占用进程)是否是潜在傀儡进程列表中的一个进程。例如,可参考已知已经被用作傀儡进程的进程的列表和/或有可能被用作傀儡进程的进程的列表来确定第一进程是否与列表上的进程相关联。除此之外或作为另外一种选择,可基于进程的一个或多个特征来将进程包括在潜在傀儡进程的列表中。在一些情况下,潜在傀儡进程的列表包括具有命令行接口的至少一个进程。例如,具有命令行接口特征的进程可被认为比没有命令行接口特征的进程更有可能被恶意实体用作傀儡进程。因此,具有命令行接口特征的进程可被包括在潜在傀儡进程的列表中。除此之外或作为另外一种选择,潜在傀儡进程的列表可包括以下中的至少一者:cmd.exe、rundll.exe、rundll32.exe、regsvr32.exe、dllhost.exe、regedit.exe、taskhost.exe、cscript、wscript、vbscript、perlscript、bash、ldconfig、terminal.app和x-code.app。

[0034] 在一个实施例中,分类模块325可维护潜在傀儡进程的列表。分类模块325可结合已登记的系统事件检测新的傀儡进程。例如,基于来源识别模块115-a执行的分析,该分析可至少部分地基于已登记的系统事件,可确定进程X是傀儡进程。然而,分类模块325可确定进程X未出现在潜在傀儡进程的列表上。因此,在确定进程X未被包括在潜在傀儡进程的列表中后,分类模块325可将进程X添加到潜在傀儡进程的列表中。因此,分类模块325可通过确定第二进程启动第一进程来识别新的傀儡进程,该第二进程将第一进程配置成执行导致可疑事件的某些操作,这可触发加强对可疑事件的审查。因此,在确定第二进程创建并启动生成可疑事件的第一进程并且将第二进程识别为实际上负责可疑事件的进程后,可将第一进程指定为新的傀儡进程并且作为潜在傀儡进程添加在潜在傀儡进程数据库中。

[0035] 恶意进程可由于目标进程的某些特征而以特定进程为目标。例如,目标进程可包括命令行接口,该命令行接口使得恶意进程能够有效地远程控制目标进程,即使在恶意进程已经终止或甚至删除其自身之后也如此。因此,目标进程可被称为傀儡进程。在识别启动该进程的启动进程后,识别模块315可将目标进程指定为傀儡进程。在一些情况下,监测模块305可确定启动进程在启动该进程后关闭。在一些情况下,确定启动进程在发起傀儡进程后关闭可提供进一步指示并且增加可疑事件和启动进程是恶意的可能性。

[0036] 在一些情况下,监测模块305可确定傀儡进程是由启动进程经由直接在命令行接口上提供的指令所发起。除此之外或作为另外一种选择,监测模块305可确定傀儡进程是经由包括在文件中的指令所发起。例如,监测模块305可确定启动进程负责在傀儡进程的命令行接口上提供到达一个或多个文件的路径。因此,在检测到由于在命令行接口上提供的指令的执行而产生的可疑事件后,识别模块315可将启动进程识别为最终负责该可疑事件的

进程。

[0037] 发起傀儡进程的进程可被指定为启动进程或来源进程。在确定生成可疑事件的进程是潜在傀儡进程后,识别模块315可识别发起该傀儡进程的启动进程。在一些实施例中,识别模块315可通过查询存储在数据库中的已登记的系统事件(例如,系统事件125)来识别启动进程。系统事件可包括进程启动事件、文件系统事件等。因此,登记在数据库中的系统事件可指示傀儡进程由特定进程启动。识别模块315可将这个识别出的特定进程指定为启动进程。在一些实施例中,系统事件可指示启动进程在发起傀儡进程之后关闭。因此,所获得的关于启动进程和/或傀儡进程的动作、方面或特征的信息可提供关于可疑事件来源的指示。在一个实施例中,通知模块320可在检测到可疑事件和/或识别出启动进程时生成通知。该通知可将启动进程识别为可疑事件的来源。因此,如上所述,来源识别模块115可使用文件系统驱动程序来将死亡投放文件的创建追溯到创建它们的进程。来源识别模块115可监测所有进程启动。因而,当来源识别模块115检测到经由直接/间接指向一个或多个文件(例如,死亡投放文件)的命令行指令的进程启动(例如,傀儡进程启动)时,来源识别模块115将该一个或多个文件的创建者作为对该傀儡进程所做的任何事情的责任进程。

[0038] 图4是示出用于识别可疑事件来源的方法400的一个实施例的流程图。在一些配置中,方法400可通过图1、图2和/或图3中示出的来源识别模块115实施。在一些配置中,方法400可通过图1和/或图2中示出的应用程序120实施。

[0039] 在框405处,可在数据库中登记系统事件。在一些情况下,可经由内核模式驱动程序登记系统事件。在框410处,可检测与第一进程相关联的可疑事件。在框415处,可将第一进程识别为多个潜在傀儡进程中的一个。在框420处,在确定第一进程是潜在傀儡进程后,可查询数据库中已登记的系统事件以识别被检测为启动第一进程的第二进程。

[0040] 图5是示出用于维护潜在傀儡进程列表的方法500的一个实施例的流程图。在一些配置中,方法500可通过图1、图2和/或图3中示出的来源识别模块115实施。在一些配置中,方法500可通过图1和/或图2中示出的应用程序120实施。

[0041] 在框505处,维护潜在傀儡进程的列表。在框510处,可经由已登记的系统事件检测新的傀儡进程。例如,可采用内核模式驱动程序来检测并在数据库中登记系统事件。在框515处,可将新的傀儡进程添加到潜在傀儡进程的列表。

[0042] 图6是示出用于识别可疑事件来源的方法600的一个实施例的流程图。在一些配置中,方法600可通过图1、图2和/或图3中示出的来源识别模块115实施。在一些配置中,方法600可通过图1和/或图2中示出的应用程序120实施。

[0043] 在框605处,可检测到第一进程执行。如上所述,可在数据库中登记并存储一个或多个系统事件。因此,可将第一进程和其他进程的操作存储在这样的数据库中。在框610处,可检测到第一进程发起第二进程。在框615处,可检测到第二进程生成可疑事件。在框620处,可获得关于第一进程的信息。例如,所登记的系统事件可包括关于发起第二进程的第一进程的数据。在一些情况下,所登记的系统事件可指示第一进程在发起第二进程之后关闭。在框625处,可将第二进程与潜在傀儡进程的列表进行比较。此类进程的例子包括从命令行接口发起的进程,诸如cmd.exe、rundll32.exe等。在框630处,如果发现匹配项,则可将第二进程指定为傀儡进程。在框635处,如果没有匹配项,则在确定第一进程发起了第二进程后,可将第二进程添加到所述列表。例如,在确定第一进程发起了第二进程并且第二进程生成

了可疑事件后,那么可将第一进程添加到潜在傀儡进程的列表。在框640处,可将第一进程识别为可疑事件的来源。如上所述,在一些情况下,可经由通过内核模式驱动程序登记的系统事件来识别第一进程。在一些实施例中,在将第一进程识别为可疑事件的来源后,可生成通知。在一些情况下,该通知可将启动进程识别为可疑事件的来源。除此之外或作为另外一种选择,该通知可将启动进程识别为潜在恶意进程。

[0044] 图7示出了适于实施本发明的系统和方法的计算机系统700的框图。计算机系统700包括使计算机系统700的主要子系统互连的总线705,所述子系统诸如中央处理器710、系统存储器715(通常为RAM,但是也可包括ROM、闪存RAM等)、输入/输出控制器720、外部音频设备(诸如经由音频输出接口730的扬声器系统725)、外部设备(诸如经由显示适配器740的显示屏735)、键盘745(与键盘控制器750连接)(或其他输入设备)、多个USB设备765(与USB控制器770连接),以及存储接口780。此外,还包括通过串行端口760连接至总线705的鼠标755(或其他点击设备)以及网络接口785(直接连接到总线705)。

[0045] 总线705允许在中央处理器710和系统存储器715之间进行数据通信,如前文提及,系统存储器可以包括只读存储器(ROM)或闪存存储器(均未示出)以及随机存取存储器(RAM)(未示出)。RAM通常是将操作系统和应用程序装载到其中的主存储器。除了其他代码,ROM或闪存存储器可以包含控制基本硬件操作(如与外围组件或设备的交互)的基本输入输出系统(BIOS)。例如,用以实施本发明的系统和方法的来源识别模块115-b可存储在系统存储器715中。与计算机系统700驻存在一起的应用程序(例如,应用程序120)通常存储在非暂态性计算机可读介质上并通过它进行访问,所述非暂态性计算机可读介质诸如硬盘驱动器(例如固定磁盘775)或其他存储介质。此外,通过接口785进行访问时,可以根据应用程序和数据通信技术以电子信号的形式对应用程序进行调制。

[0046] 存储接口780与计算机系统700的其他存储接口一样可以连接到标准计算机可读介质(如固定磁盘驱动器775)以用于存储和/或检索信息。固定磁盘驱动器775可以是计算机系统700的一部分,或者可以是独立的,并且可以通过其他接口系统进行访问。网络接口785可以经由直接网络链路提供到远程服务器的直接连接,或经由POP(入网点)提供到互联网的直接连接。网络接口785可以使用无线技术提供此类连接,包括数字蜂窝电话连接、蜂窝数字分组数据(CDPD)连接、数字卫星数据连接等。

[0047] 很多其他设备或子系统(未示出)可以通过相似的方式进行连接(例如文档扫描仪、数码相机等)。相反,不需要提供图7中示出的所有设备亦可实施本发明的系统和方法。可以使用与图7中所示方式不同的方式来使设备和子系统互连。计算机系统的操作(如图7中所示的操作)是本领域中易知的,在本申请中不进行详细讨论。用以实施本发明的代码可存储于非暂态性计算机可读介质中,诸如一个或多个系统存储器715或固定磁盘775中。计算机系统700上提供的操作系统可为iOS®、MS-DOS®、MS-

WINDOWS®、OS/2®、UNIX®、LINUX®或另外的已知操作系统。

[0048] 此外,关于本文所述的信号,本领域的技术人员将会认识到,可以将信号从第一块直接传输到第二块,或者可以在块之间修改信号(例如,放大、衰减、延迟、锁存、缓冲、反转、过滤或其他修改方式)。尽管上述实施例的信号被特性化为从一个块传输到下一个块,但本发明的系统和方法的其他实施例可以包括经过修改的信号来代替这些直接传输的信号,只

要信号的信息和/或功能方面在块之间传输。在某种程度上,由于所涉及的电路的物理限制(例如,不可避免地存在一些衰减和延迟),第二块上的信号输入可以概念化为得自第一块输出的第一信号的第二信号。因此,如本文所用,得自第一信号的第二信号包括第一信号或对第一信号所做的任何修改,无论是由于电路限制还是由于通过不会改变第一信号的信息和/或最终功能方面的其他电路元件。

[0049] 虽然上述公开内容使用特定框图、流程图和例子阐述了各种实施例,但每个框图组成部分、流程图步骤、操作和/或本文描述和/或示出的组件可使用多种硬件、软件或固件(或其任何组合)配置单独和/或共同地实施。此外,包含在其他组件内的部件的任何公开内容应当被视为在本质上是示例性的,因为可实施许多其他体系结构来实现相同功能。

[0050] 本文描述和/或示出的过程参数和步骤顺序仅通过举例的方式给出并且可根据需要改变。例如,虽然本文示出和/或描述的步骤可以特定顺序示出或讨论,但这些步骤不必按示出或讨论的顺序来执行。本文描述和/或示出的各种示例性方法也可省略本文描述或示出的步骤中的一者或多者,或除了所公开的那些步骤之外还包括附加步骤。

[0051] 此外,虽然本文已经在全功能计算系统的背景中描述和/或示出了各种实施例,但这些示例性实施例中的一者或多者可作为各种形式的程序产品来分发,而不考虑用于实际进行分发的计算机可读介质的特定类型。本文所公开的实施例也可使用执行某些任务的软件模块来实施。这些软件模块可包括脚本、批文件或可存储在计算机可读存储介质上或计算系统中的其他可执行文件。在一些实施例中,这些软件模块可将计算系统配置为执行本文所公开的示例性实施例中的一者或多者。

[0052] 出于阐释目的,已关于特定实施例进行了以上描述。然而,以上示例性讨论并非旨在是穷举的或将本发明限制为所公开的精确形式。鉴于上述教导,许多修改形式和变型形式都是可能的。为了最好地解释本发明的系统和方法的原理及其实际应用,选择并描述了实施例,由此使得其他本领域的技术人员能最好地利用本发明的系统和方法以及具有可适合于所设想的特定用途的各种修改形式的各种实施例。

[0053] 除非另有说明,否则在本说明书和权利要求书中使用的术语“一”或“一个”应解释为意指“…中的至少一个”。此外,为了易于使用,在本说明书和权利要求书中使用的词语“包括”和“具有”与词语“包含”可互换并且与词语“包含”具有相同含义。此外,将在说明书和权利要求书中使用的术语“基于”理解为“至少基于”的含义。

100

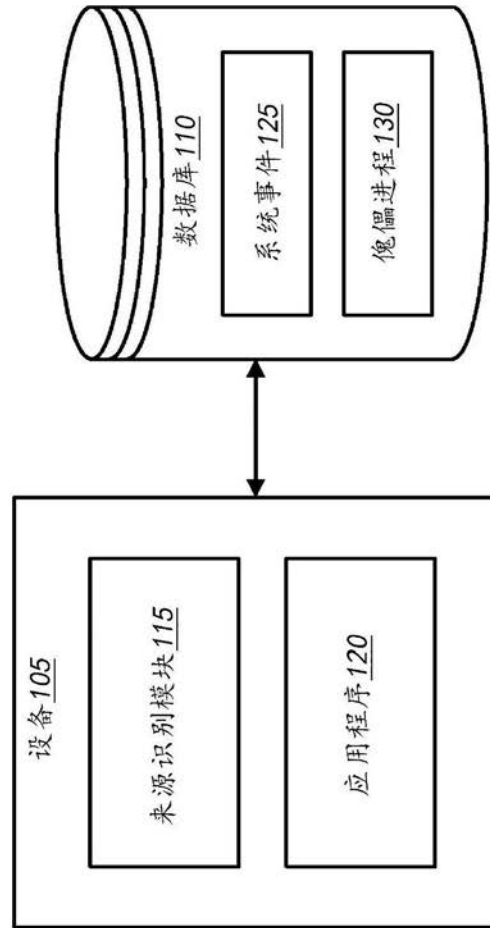


图1

200

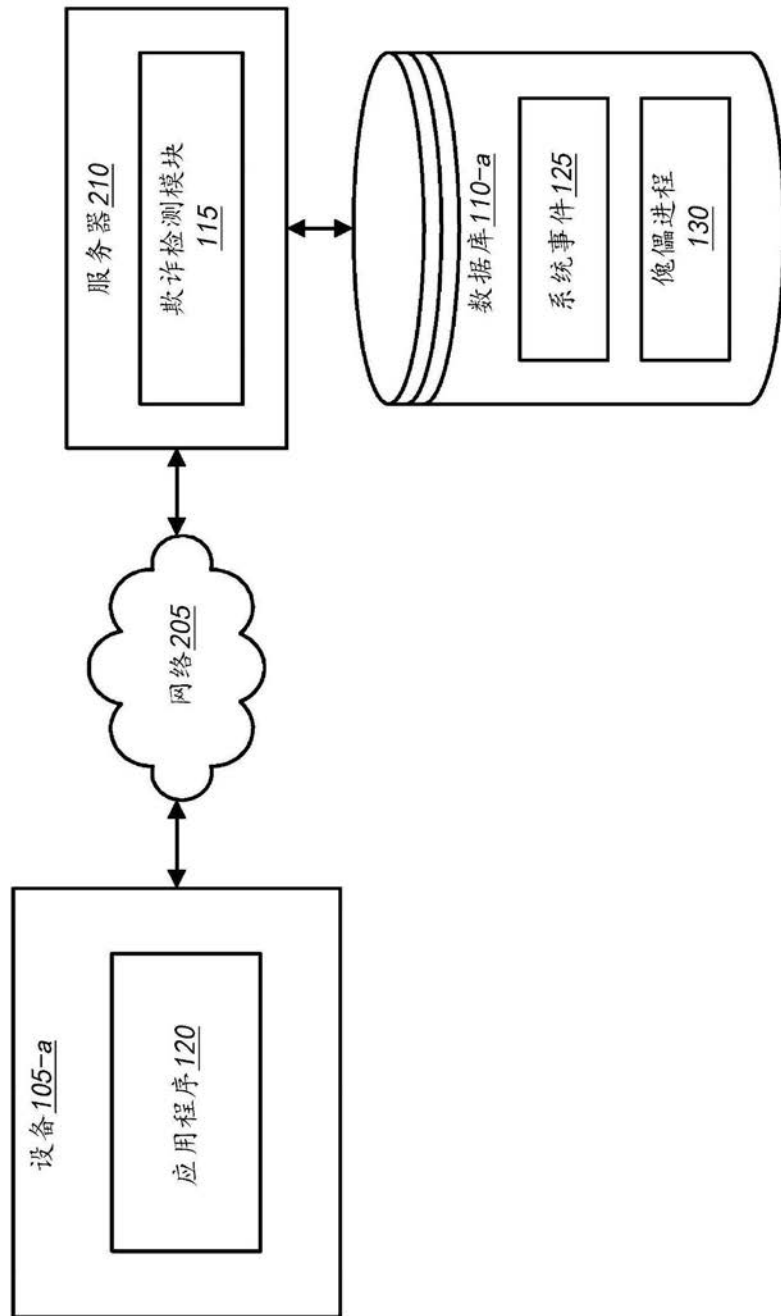


图2

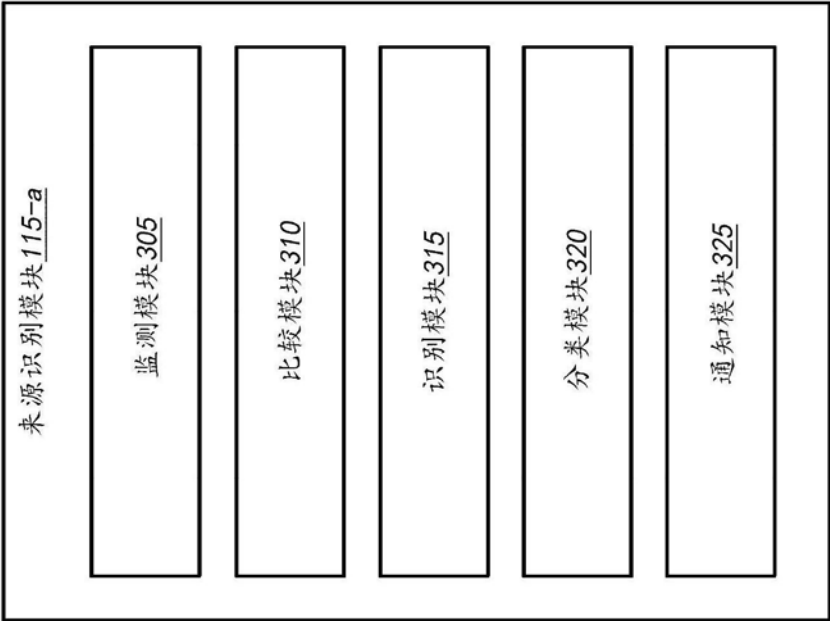


图3

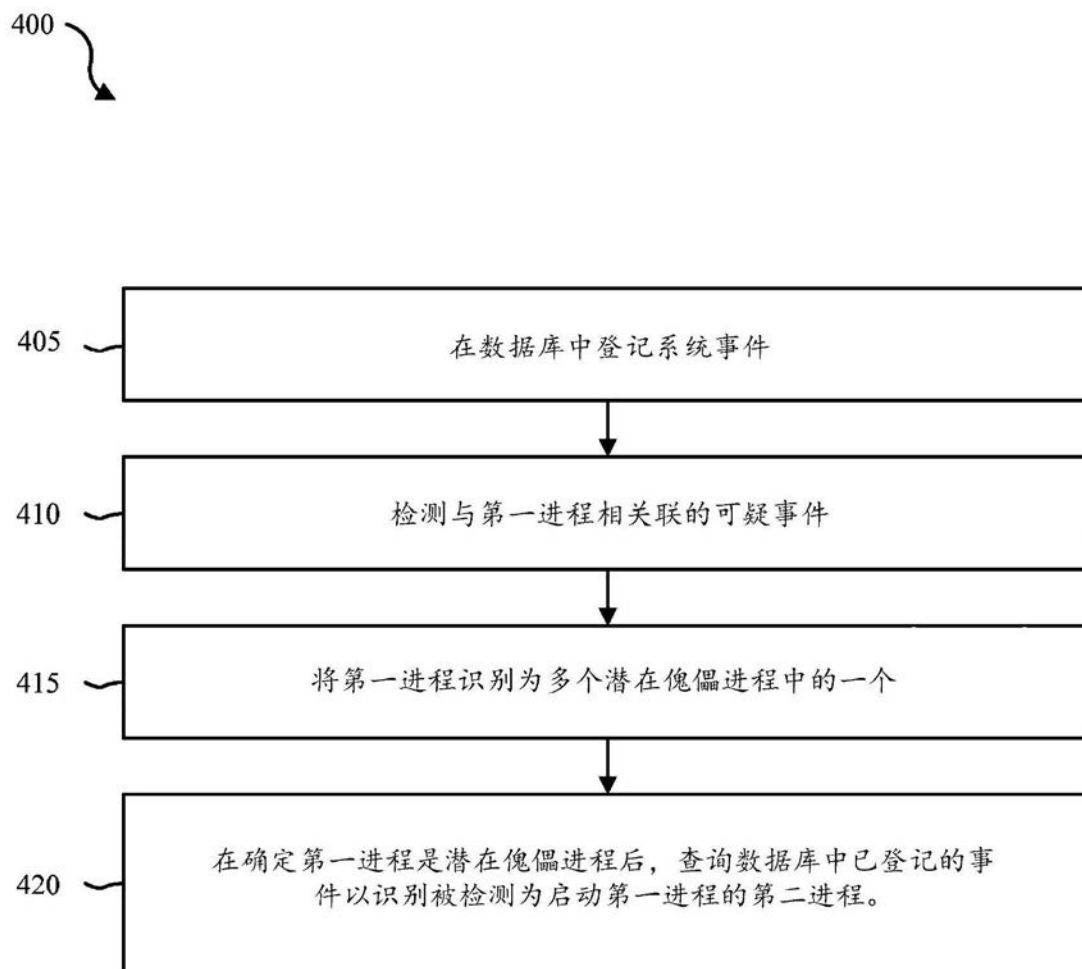


图4

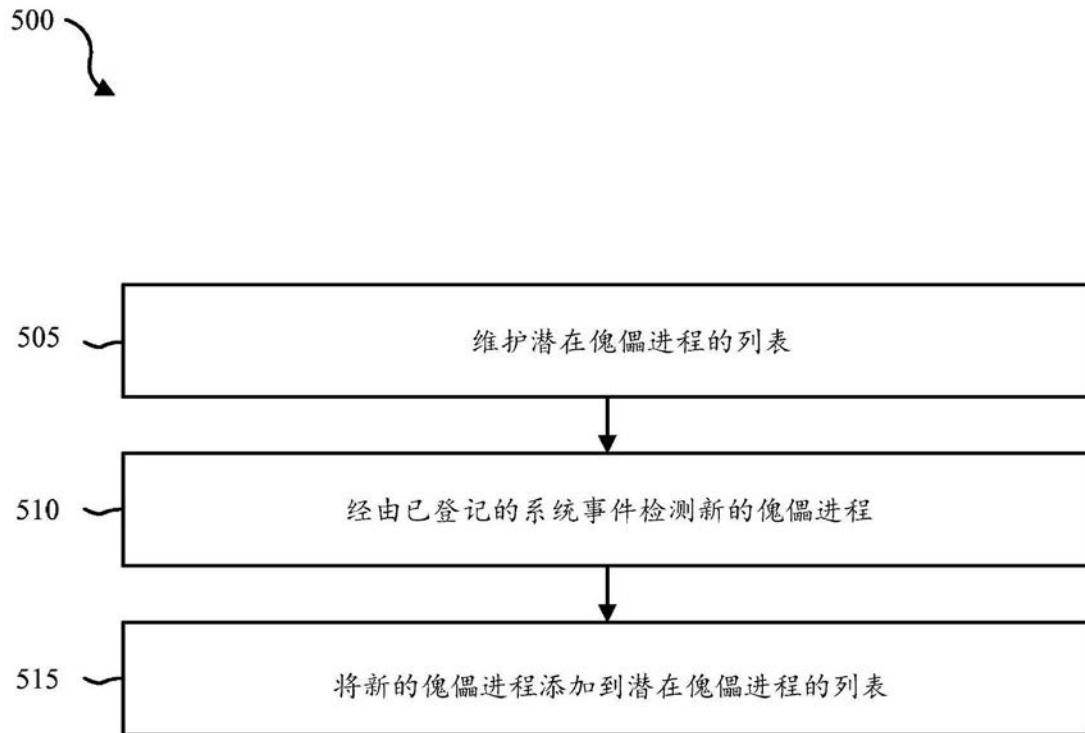


图5

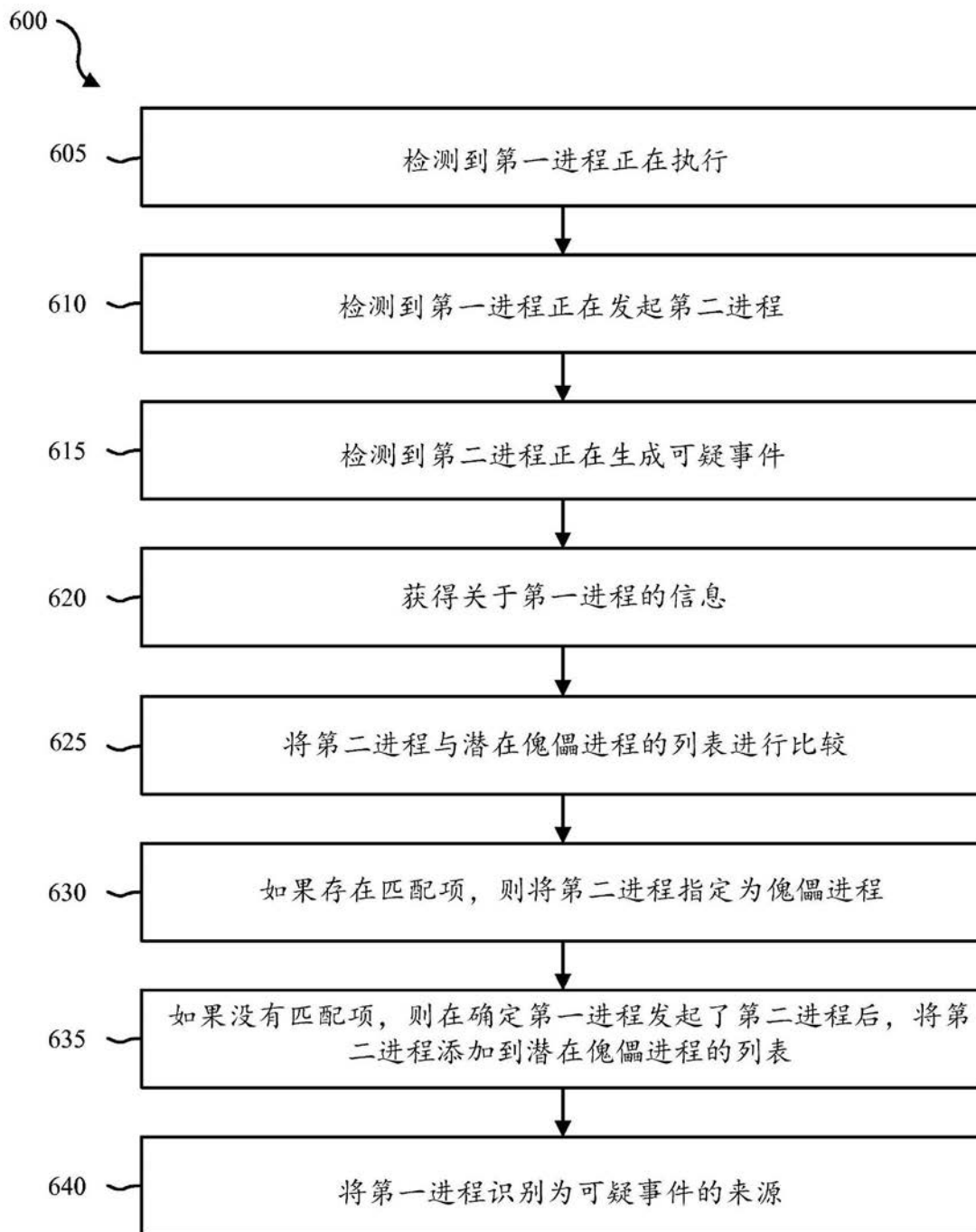


图6

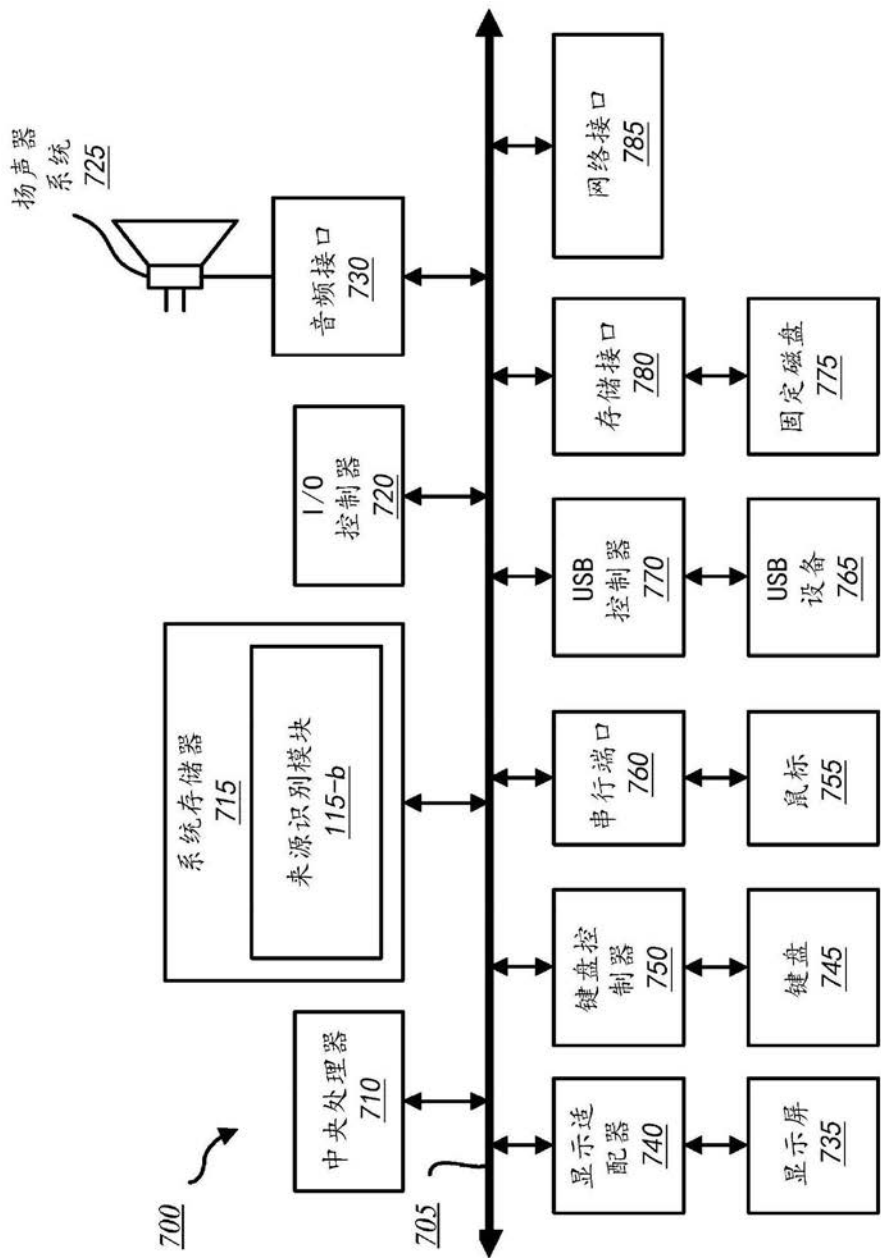


图7