(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 103530563 B
(45)授权公告日 2016.08.10

(21)申请号 201310349574.5

(22)申请日 2013.08.12

(30)优先权数据
2012134242 2012.08.10 RU

(73)专利权人 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 安德烈·Y·索洛多思尼克夫
基里尔·N·克鲁格洛夫

(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 21/57(2013.01)

(56)对比文件
US 2005/0216909 A1,2005.09.29,说明书第[0004],[0013]-[0015],[0022],[0030]-[0033]段、图1-4.
US 2003/0226139 A1,2003.12.04,说明书第[0006],[0013]-[0030]段、图1-2.
CN 101410800 A,2009.04.15,全文.
CN 101888623 A,2010.11.17,全文.
CN 1894661 A,2007.01.10,全文.
US 2009/0144718 A1,2009.06.04,全文.
US 2007/0094654 A1,2007.04.26,全文.
US 2007/0277167 A1,2007.11.29,全文.
US 2005/0097543 A1,2005.05.05,全文.

审查员 张峰

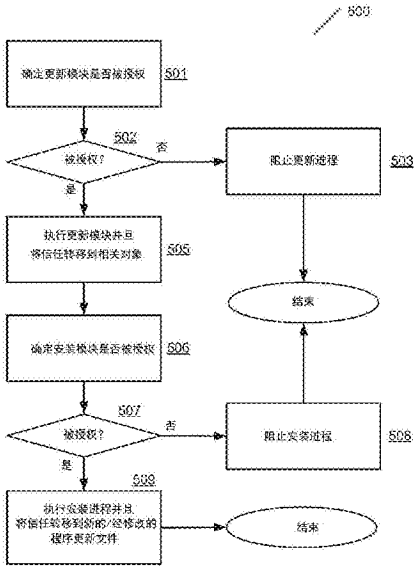
权利要求书1页 说明书11页 附图6页

(54)发明名称

用于更新经授权软件的系统和方法

(57)摘要

公开了用于在计算机上更新软件程序的系统、方法和计算机程序产品。系统检测更新进程要在计算机上执行的尝试并且检索关于软件程序的经授权类别的信息来确定所检测的更新进程是否被授权。当更新进程被授权时，系统(i)将更新进程指定为受信任的进程，(ii)允许更新进程在计算机上下载更新对象，(iii)以及将更新对象指定为受信任的对象。系统随后检测安装进程要安装更新对象的尝试，并且根据策略确定所检测的安装进程是否与经授权类别相关联并且与受信任的更新进程相关。当安装进程被授权并且与受信任的更新进程相关时，系统允许受信任的安装进程安装受信任的更新对象。



CN 103530563 B

1.一种用于在计算机上更新软件程序的方法,所述方法包括:

检测更新进程要在所述计算机上执行的尝试,其中在执行期间,所述更新进程下载用于软件程序的更新对象;

检索至少包含用于所述计算机的软件程序的经授权类别的策略,其中所述经授权类别包括经允许在所述计算机上执行和更新的软件程序,所述软件程序被检查为没有恶意软件;

从所述策略确定所检测的更新进程是否与软件程序的所述经授权类别相关联;

当所述更新进程与软件程序的所述经授权类别相关联时,(i)将所述更新进程指定为受信任的进程,(ii)允许所述受信任的更新进程在所述计算机上下载用于软件程序的更新对象,(iii)以及将所述软件程序的所述更新对象指定为受信任的对象;

由所述受信任的更新进程在所述计算机上下载所述受信任的更新对象,其中所述受信任的更新对象包含软件程序更新;

检测安装进程要在所述计算机上执行的尝试,其中在执行期间,所述安装进程安装所述受信任的更新对象;

从所述策略确定所检测的安装进程是否是所述受信任的更新进程的子进程并且与软件程序的所述经授权类别相关联;以及

当所述安装进程是所述受信任的更新进程的子进程并且与软件程序的所述经授权类别相关联时,将所述安装进程指定为受信任的并且允许所述受信任的安装进程在所述计算机上安装所述受信任的更新对象。

2.根据权利要求1所述的方法,进一步包括:

当所述更新进程不与软件程序的所述经授权类别相关联时,阻止所述更新进程要在所述计算机上执行的所述尝试;

当所述安装进程不是所述受信任的更新进程的子进程时,从所述策略确定所述安装进程是否与经授权类别相关联;以及

当所述安装进程不与所述经授权类别相关联时,阻止所述安装进程要在所述计算机上安装所述更新对象的所述尝试。

3.根据权利要求1所述的方法,进一步包括:

在所述计算机上部署远程管理服务器的管理代理端;

由所述管理代理端收集关于安装在所述计算机上的软件程序的信息;以及

将所收集的信息发送到所述远程管理服务器。

4.根据权利要求3所述的方法,进一步包括;

基于所述所收集的信息对所述软件程序进行分类;以及

生成包含软件程序的经授权类别的策略,其中允许来自所述经授权类别的软件程序在所述计算机上执行和更新。

5.根据权利要求4所述的方法,其中从所述策略确定所述更新进程是否与软件程序的所述经授权类别相关联进一步包括:

确定是否允许所述计算机的给定用户启动所述更新进程。

用于更新经授权软件的系统和方法

[0001] 相关申请的交叉引用

[0002] 根据美国法典第35篇119条(a)-(d)项，本申请要求于2012年8月10日提交的申请号为2012134242的俄罗斯申请的优先权，其通过援引的方式在此并入。

技术领域

[0003] 本公开总地涉及计算机安全领域，并且更具体地，涉及用于更新软件的系统、方法和计算机程序产品。

背景技术

[0004] 无法想象现代的办公室，无论其大小，会没有计算机网络。网络允许在可位于一座建筑物中或位于相远离地理区域的多个办公室中的企业计算机、笔记本电脑、智能电话、平板电脑、打印机、扫描仪和其他联网电子设备之间进行迅速而安全的信息交换。因此，企业的信息技术(IT)基础结构可能变得相当复杂并且要求经常的管理和支持以维持其操作和解决各种问题。例如，企业的计算机网络易受恶意软件诸如病毒、蠕虫、特洛伊木马和其他类型恶意软件的攻击的影响。恶意软件通常利用在计算机上运行的程序和应用程序的漏洞来渗透企业的安全基础结构。在大型更新过程期间，恶意软件经常可隐藏在软件更新中并感染一系列计算机。感染一台企业计算机的恶意软件可随后迅速地通过网络传播并感染其他计算机。由于需要识别受感染的计算机和修复由恶意软件所造成的损害，以及花费在补救受影响的计算机的工时损失，这些事件不利地影响公司的运作。而且，这些事件可引起机密信息、商业秘密、专有数据的泄漏或损失以及甚至盗窃金钱。那就是为什么对诸如内联网的企业网络的控制是非常重要的原因，特别是当部署在其上的各种应用程序执行和更新时。

[0005] 对企业网络上的程序和应用程序的执行和更新的控制通常由软件安全系统所实施。这些系统使用不同的方法用于控制软件的执行和更新。例如，一些安全系统允许未被明确禁止的所有软件的执行。在这种情况下，管理员可规定被禁止的应用程序的列表，以便允许所有其他应用程序在计算机上执行或更新。其他安全系统禁止未经明确允许的所有软件的执行。在这种情况下，管理员可规定经允许的应用程序的列表，以便禁止所有其他应用程序执行或更新。就所关心的管理的安全性和有效性而言，使用拒绝执行和更新所有被禁止的应用程序的方法的安全系统被认为是更有效的。这些类型的系统有时被称为“默认拒绝”系统。

[0006] 在操作中，进行应用程序控制的“默认拒绝”安全系统经常维持经授权对象列表(例如，用于网络中的一台或多台计算机的经允许的应用程序或程序的列表)。系统将允许执行在经授权对象列表上的应用程序，并且阻止所有其他应用程序和程序。这些列表可基于诸如唯一对象标识符、数字签名、制造商数据或其他类型信息的关于应用程序和程序的特定信息来创建。此外，针对不同的用户可维持不同的经授权对象列表。例如，针对图形设计者的经授权应用程序列表可包括图形设计软件；而针对会计师的经授权应用程序列表将

包括会计软件。这样，“默认拒绝”安全系统针对不同的用户维持单独的经授权软件列表，由此防止未经授权用户访问未经授权软件。

[0007] 通常，“默认拒绝”安全系统在保护企业计算机免受恶意软件之害方面是有效的。然而，在软件更新期间，这些类型的安全系统具有错误地禁止执行经授权软件或者甚至完全阻止一些软件更新的风险。这可在当在软件更新期间由经更新的软件创建附加文件或将附加文件加载到计算机且所述经更新的软件在经授权对象列表中未被识别时，或当修改现有文件时发生。尽管这些新创建的、所加载和/或经修改的文件与经授权应用程序相关联并且因此应被允许在计算机上执行，但是“默认拒绝”安全系统通常将阻止这些文件因为它们未出现在经授权对象列表中，这导致对恶意软件的误报检测。因此，对于安全系统而言，需要用于控制和管理软件在企业计算机上的部署和更新的、最小化恶意软件的传播并且同时防止对恶意软件的误报检测。

发明内容

[0008] 公开了用于在计算机上更新软件程序的系统、方法和计算机程序产品。在一个示范性实施例中，安全系统检测更新进程要在计算机上执行的尝试。系统检索至少包含用于计算机的软件程序的经授权类别的策略。允许来自软件程序的经授权类别的软件程序在计算机上执行。系统由策略来确定所检测的更新进程是否与软件程序的经授权类别相关联。当更新进程与软件程序的经授权类别相关联时，系统(i)指定更新进程为受信任的进程，(ii)允许受信任的更新进程在计算机上下载用于软件程序的更新对象，(iii)以及指定软件程序的更新对象为受信任的对象。

[0009] 受信任的更新进程可随后在计算机上下载受信任的更新对象。受信任的更新对象包含软件程序更新。系统随后检测安装进程要在计算机上执行的尝试。在执行期间，安装进程安装受信任的更新对象。系统由策略来确定所检测的安装进程是否与受信任的更新进程有关并且是否与软件程序的经授权类别相关联。当安装进程与受信任的更新进程有关并且与软件程序的经授权类别相关联时，系统指定安装进程为受信任的并且允许受信任的安装进程在计算机上安装受信任的更新对象。

[0010] 上述示范性实施例的简化概括用来提供对本发明的基本理解。该概括不是本发明所有预期方面的广泛综述，并且既不意图识别所有实施例的关键或者重要元素也不意图划定任何或者所有实施例的范围。其唯一目的是以简化形式呈现一个或多个实施例作为以下本发明的更详细说明的前序。为了完成前述内容，一个或多个实施例包括权利要求中所描述和所特别指出的特征。

附图说明

[0011] 附图合并到本说明书中并且构成本说明书的一部分，示出了本发明的一个或多个示范性实施例，并且与详细描述一起，用来解释实施例的原理和实现方案。

[0012] 在附图中：

[0013] 图1示出了企业的计算机网络的示范性实施例。

[0014] 图2示出了程序更新进程的示范性实现方案。

[0015] 图3示出了用于在客户端计算机上更新经授权程序的客户端-服务器基础结构的

示范性实施例。

[0016] 图4示出了用于更新经授权程序的系统的示范性实施例。

[0017] 图5示出了用于在企业的计算机上更新经授权程序的方法的示范性实现方案。

[0018] 图6示出了所公开的用于更新经授权程序的系统和方法可在其上实现的通用计算机的示范性实施例。

具体实施方式

[0019] 本文围绕用于在企业或家庭网络中的计算机上更新经授权软件程序的系统、方法和计算机程序产品描述了本发明的示范性实施例、实现方案以及各方面。本领域普通技术人员将了解的是，以下描述仅是说明性的并且不意图以任何方式进行限制。受益于本公开的本领域技术人员将易于获得其他实施例或实施方案的启示。现在将详细参考如附图中所示出的示范性实施例的实现方案。在贯穿于附图和以下描述中将尽可能使用同样的参考标记来指代相同或类似项。

[0020] 图1示出了由管理服务器所控制的示范性计算机网络(企业或家庭办公室)。网络101，其可以是包括一个或多个有线或无线局域网的企业内联网，可包含许多不同的联网电子通信设备，诸如台式计算机、笔记本电脑、平板电脑、智能电话、打印机等等，其在本文中被共同称为计算机103。连接到网络101的管理服务器102在计算机103上实施各种管理任务。管理服务器102可对于网络101是本地的。可替代地，管理服务器102可自网络101是远程的。在一个实现方案中，管理服务器102可在个体计算机103上部署本地管理代理端(agent)以实施网络101中的计算机103的各种远程管理任务。

[0021] 在各种实现方案中，管理服务器102可直接或者通过其部署在计算机103上的管理代理端来实施不同的管理任务。这些管理任务可包括但不限于在计算机103上更新反病毒数据库和应用程序、安装和更新软件、在计算机103上搜索安全漏洞、管理安全策略和组任务、实施计算机103的软件和硬件清点(inventory)、在管理服务器102上保存所收集的信息、以及其他类型的远程管理任务。

[0022] 在一个实现方案中，远程管理服务器102可经由因特网与计算机103通信，并且，由于该网络的不可靠性，可能与一个或多个计算机103失去连接。在这种情况下，管理服务器102可指定网络101中的任何其上部署管理代理端且与其维持连接的计算机103以作为服务器102的临时代理(proxy)发挥作用，来与至少已经与其临时失去连接的计算机103进行通信，直到在服务器102和该计算机之间的连接得到重新建立。因此，所指定的代理计算机103的管理代理端将从服务器102临时接收管理任务请求并且将其转发给网络101中的所指定的计算机103。

[0023] 图2示出了程序更新进程的示例。每次程序更新变为可用时，必须更新安装在网络101中的计算机103上的软件程序。这类更新由于数个原因而是必需的。例如，软件可能在其原始代码中具有错误(bug)，其在软件的初始测试期间被遗漏，因此其可能保留在卖给用户的软件的最终版本中。这些错误可造成软件不正确地工作、造成软件消耗过度的系统资源(例如，存储器、处理器时间)、或者创建可被恶意软件所利用的安全漏洞。因此，软件更新提供改正各种已知缺点、去除安全漏洞、和/或扩展或改进软件的功能性的必需的补救。

[0024] 在一个示范性实现方案中，可由更新模块201发起和实施在计算机103上所安装的

程序的更新。在一个实现方案中，模块201可以是正被更新的软件程序的组件(例如，代码、脚本和子例程)。在另一个实现方案中，模块201可以是实施一个或多个相关程序的更新的专用程序(例如，诸如Kaspersky One®的软件套件)。而在另一个实现方案中，模块201可以是部署在计算机103上的管理代理端的软件组件。无论哪一种情况，更新模块201均可配置为从远程更新服务器来获得用于安装在计算机103上的一个或多个程序的更新，所述远程更新服务器例如管理服务器102，可由例如制造正被更新的程序的公司或第三方更新服务供应商所托管。

[0025] 在一个实现方案中，更新模块201可周期性地或以预定日期/时间来在将消息发送到远程更新服务器以检查新的更新的计算机103上启动更新进程205。如果服务器有新的程序更新，那么更新模块201将更新下载到计算机103。在另一个实现方案中，当更新服务器有新的程序更新时，服务器可发起程序更新数据的传输或到更新模块201的传输。

[0026] 可提供程序更新作为具有以下文件类型：*.bat、*.cmd、*.js、*.vbs、*.pif中的一个的安装包202或者作为二进制文件。用于基于Windows的程序的典型安装包202可以是*.msi或*.msp文件类型。安装包202典型地包含用于在计算机103上正确更新软件程序所必需的代码和数据。这类数据可包括经更新的程序文件、关于目录结构的信息、OS注册表信息以及其他类型的数据。

[0027] 在一个示范性实施例中，安装模块203通过在计算机103上启动安装进程206来实施程序更新即安装包202的安装。在一个实现方案中，安装模块203的功能性可包括在更新模块201中。在Windows OS中，安装模块203可以是程序msiexec.exe。模块203通过跟随在安装包202中所包含的指令以及通过使用在该包202中的数据来更新程序。在一个实现方案中，在程序更新期间可使用数个安装模块203，与此同时对安装进程206的控制连续地从一个安装模块203转移到另一个。还有其他实现方案，此时参与模块203之间的控制的转移可以使用例如进程间通信(IPC)机制而以更不明显的方式完成。应该注意，在一个示范性实现方案中，安装模块203中的任何一个均可加载用于正确更新所必需的新安装包202。而在另一个实现方案中，安装模块203可将安装包202变成临时文件，并且随后使用组件对象模型(COM)-服务器机制从该临时文件初始化包202的安装。

[0028] 应该进一步注意，在程序更新期间，即包202的安装期间，可从安装包202和/或旧的程序文件204创建新的程序文件204或者可修改他们的元数据(例如，名称、大小、位置)。在这些情况下，在经授权程序即根据管理策略被允许更新的程序的更新期间，在创建新的或经修改的程序文件204期间可能会发生错误，因为关于这些文件或其元数据以将这些文件正确地分类为经授权或未经授权的信息不足。而且，元数据可在程序更新期间改变并且可不再与经授权元数据的类别相对应。在这些情况下，由于未经授权，程序的更新可能受到阻止。

[0029] 图3示出了用于在企业计算机上更新经授权程序的客户端-服务器基础结构的示范性实施例。系统300包括管理服务器102和多个计算机103。管理服务器102包括控制模块301、多个管理模块302以及数据存储303。管理模块302可包括但不限于清查(inventorization)模块304、分类模块305、策略模块306和网络拓扑建立模块(未示出)、用于在计算机上搜索和移除漏洞的模块(未示出)以及其他模块。数据存储303包含用于不同程序的更新、网络101的拓扑上的信息、安全补丁、已知漏洞列表、用于网络101中的每一个

计算机103的软件和硬件信息、程序分类规则、程序控制的策略以及其他数据。数据存储303可利用由反病毒公司所提供的信息来不断地更新。

[0030] 在一个示范性实施例中，管理服务器102在计算机103上部署管理代理端307。服务器102使用管理代理端307以远程地管理计算机103上的程序。管理服务器102使用控制模块301以与计算机103上的代理端307通信以及实施各种远程任务，诸如清查和分类计算机103上的应用程序和程序、应用安全策略和程序更新和。在一个实现方案中，管理服务器102可使用用于微软交换服务器的Kaspersky Administration Kit(卡巴斯基管理工具包)、Kaspersky Endpoint Security(卡巴斯基终端安全软件)、Kaspersky Security(卡巴斯基安全软件)和/或用于微软工作站的Kaspersky Anti-Virus(卡巴斯基反病毒软件)来实施各种管理任务。

[0031] 为了实施计算机103的管理，管理服务器102使用控制模块301来连接到在计算机103上所部署的管理代理端307。然后，管理服务器102激活清查模块304来制定管理代理端307要在计算机103上所实施的清查任务。在清查期间，管理代理端307收集关于在计算机103上所安装的软件应用程序的信息。所收集的信息可包括但不限于应用程序的版本、由应用程序所使用的数字签名、唯一程序标识符(例如，哈希或校验和)、制造商信息、应用程序安装路径等等。为每一个计算机103所收集的信息均由唯一标识符所指定，所述唯一标识符诸如计算机103的序列号、MAC地址、IP地址等等。为每一个计算机103所收集的清点数据储存在数据存储303中，并且可使用与计算机103相关联的唯一标识符在存储303中对其进行搜索。

[0032] 可由控制模块301周期性地实施清查进程，其使得能够在计算机103上找到由用户所安装的新应用程序和程序。这样，网络101中的所有已知应用程序的列表总是当前的。而且，在一个实现方案中，当用户在计算机103上安装新程序时，管理代理端307可收集关于该程序的信息并且将其发送到管理服务器102以在存储303中更新软件清点数据。

[0033] 在一个实现方案中，管理模块302使用分类模块305来将安装在计算机103上的应用程序分类为不同的类别。在一个实现方案中，数据存储303储存软件分类规则。这些规则可基于应用程序的不同的元数据，比如文件的名称或唯一标识符、或制造商的名称等等。例如，用于制造商名称“Opera Software”的规则可将该制造商分类为“浏览器”。而且，可以有将用于OS的操作所必需的所有应用程序和文件分组为类别“操作系统的文件”的规则。而且，例如，可以有将需要更新的所有应用程序分组为类别“用于更新的应用程序”的规则。而且，可以有将受信任的、非恶意的应用程序分类为被授权以在计算机103上安装和执行的应用程序的规则。分类模块305可将这类经授权的应用程序放置到经授权类别“经授权应用程序”中。

[0034] 在一个实现方案中，分类模块305通过对存储在存储303中的数据运用分类规则来对安装在计算机103上的所有应用程序进行分类。一旦程序分类为一个或多个类别，则分类模块305在数据存储303中将用于每个程序的分类信息添加到用于该程序的软件清点数据。在一个示范性实现方案中，可从由提供反病毒服务的公司所维持的远程数据库接收分类规则，该提供反病毒服务的公司诸如Kaspersky Lab(卡巴斯基实验室)。在这种情况下，分类模块305可连接到远程数据库以对尚未由来自存储303的规则所分类的应用程序加以分类，或者以接收新分类规则。分类模块305还可通过类别来过滤安装在计算机103上的程序以识

别哪些计算机103具有属于特定类别的程序。在另一个示范性实现方案中，系统管理员可创建由不同计算机用户所授权使用的程序的自定义类别，在该类别中包括例如特定的软件包、图形设计软件、任务计划器以及由用户所使用的浏览器。

[0035] 在一个示范性实施例中，管理模块302和分类模块305有权访问远程的干净对象数据库。该数据库可由反病毒公司提供(图3未示出)。干净对象数据库可包含关于诸如经授权(或受信任的)对象的信息，诸如各种文件、参考资料等。这类数据库包含关于所有“干净”的即其不承载恶意功能并且可在计算机103上安全地安装和执行的应用程序的非常大量的信息。分类模块305可与前述数据库连接以确定计算机103上存在哪些未经授权的(或不受信任的)程序。为此目的，分类模块305可将其在计算机103的软件清查期间所收集的信息与来自干净对象数据库的信息加以比较。在一个实现方案中，数据存储303包含用于分类未知(或不受信任的)对象的规则。这样的规则将允许对其数据在清查期间被收集，并且其数据在远程的干净对象数据库处不存在的应用程序加以分类。分类模块305通过对在清查的进程期间所收集的数据运用规则来确定这类应用程序的分类。

[0036] 在一个示范性实现方案中，系统还支持反向操作。知悉已经被分类为不受信任的应用程序实际上并非恶意的管理员可改变该应用程序的类别。同时，分类模块305可将经更新的分类信息发送给反病毒公司。例如，分类模块305可将经重新分类的应用程序的元数据发送给反病毒公司，诸如文件名称、唯一标识符、制造商名称和其他将应用程序分类为干净的信息。反病毒公司可检查应用程序的新分类的准确度并且通过将主题应用程序包括在干净对象数据库中来更新干净对象数据库。而且，可由反病毒公司来自动完成类似动作。例如，可将该特定应用程序不是恶意的信息发送给反病毒公司的服务器，其将在一定数目的用户已经发送类似信息之后更新干净对象数据库。

[0037] 在另一个实现方案中，知悉特定应用程序被分类模块305放置在错误类别中的管理员可在数据存储303中改变该应用程序的类别。同时，分类模块305可将关于应用程序的重新分类的信息发送给反病毒公司。在另一个实现方案中，当管理员无法确凿地确定是否应将某一应用程序分类为经授权(或受信任)时，其请求反病毒公司的服务器以实施应用程序的远程分类。这类动作帮助反病毒公司创建新的分类规则并且改正旧规则。然后，新的或所改正的分类规则用来更新数据存储303。

[0038] 在一个示范性实施例中，管理模块302还可包括策略模块306，所述策略模块306生成和运用对在计算机103上安装和执行应用程序和程序进行控制的策略。在一个实现方案中，策略是规定哪些用户可在计算机103上使用哪些应用程序的规则的集合。该规则的集合起码包括帮助基于程序的元数据来确定计算机103上的每个应用程序属于哪个类别的规则，以及帮助确定哪个应用程序可由哪些用户执行的规则。还有对强加给特定用户的各种限制加以规定的规则，其禁止某些应用程序的某些功能。对程序功能性进行限制的规则的示例是这样一个规则，当某些用户执行应用程序时，其拒绝应用程序访问某些网络端口。而且，其他规则可对某些用户完全禁止某些程序的使用。

[0039] 在一个示范性实现方案中，可以为计算机103的每个用户生成不同的策略。换句话说，对一个用户所允许的动作可以对另一个用户不允许，无论在同一台计算机103上或在不同的计算机103上。不允许使用由策略所禁止的应用程序类别。这样，为了禁止来自类别“Games”的所有应用程序的使用，不在来自一系列的管理模块302的策略模块306的帮助下

创建策略是足够的。因此，如果应用程序未列在某个用户的策略中的应用程序的经授权类别中，则禁止该用户在计算机103上使用该应用程序。

[0040] 如上文所解释的，用于每个计算机103的策略包含用来确定在计算机103上由用户所安装/执行的应用程序是否属于应用程序的经授权类别的规则的集合，利用该规则允许用户进行工作。在一个实现方案中，该规则的集合用来在计算机103上生成/更新经授权对象的数据库308。例如，策略可包含用于计算机103的特定用户的所有经授权应用程序的列表。该列表不但可包含经授权应用程序的名称，而且可包含经授权应用程序的其他元数据诸如唯一标识符，诸如哈希表或校验和、数字签名等等。策略模块306配置为将这类策略转移到网络101中的计算机103以允许计算机用户仅利用经授权的应用程序和程序进行工作。在一个实现方案中，策略模块306可生成转移策略到某些计算机103的任务并且将其发送到控制模块301。控制模块301在所规定的计算机103上与管理代理端307建立连接并且将适当的控制策略发送到每个计算机103。计算机103上的管理代理端307在计算机103上执行这些策略。同时，对经授权应用程序加以规定的规则的集合存储在经授权对象的数据库308中。

[0041] 在一个示范性实现方案中，管理代理端307配置为不断地监控计算机103的操作以识别用户在计算机上启动应用程序的尝试。如果做出了这样的尝试，那么代理端307可根据适当的控制策略来确定是否允许由该特定用户在该特定计算机103上启动该应用程序。代理端307通过将应用程序的元数据与存储在经授权对象的数据库308中的规则相比较来完成该过程。如果比较显示应用程序与应用程序的经授权类别相对应，那么将允许在计算机103上启动该应用程序。特别地，管理代理端307可使用规则的第一集合来确定该应用程序所属类别。然后，代理端307可使用规则的第二集合来确定在计算机103上是否可由该用户启动来自该类别的应用程序。因此，当策略包括针对计算机103的特定用户的在应用程序的某些类别内的经授权应用程序的列表，以及对用户在计算机103上尝试启动的应用程序的元数据与经授权应用程序的列表的比较指示该应用程序落入应用程序的经授权类别时，管理代理端307允许在计算机103上启动应用程序。

[0042] 在一个示范性实施例中，策略可能并不被发送到每个计算机103；反而，用于所有计算机103的策略可存储在中央数据库中，诸如管理服务器102的数据存储303。在该实现方案中，计算机103上的管理代理端307将例如基于应用程序的元数据来确定正尝试在计算机103上启动什么应用程序，并且将向管理服务器102发送询问以解决该应用程序是否被授权在计算机103上执行。服务器将检查存储在数据存储303中的控制策略和并应答指示是否允许或者不允许应用程序在计算机103上由用户执行。

[0043] 总之，为了在企业网络101中的计算机上实施应用程序的有效管理，有必要在计算机103上实施软件的清查、对所有应用程序和程序分类、并且生成和运用用于计算机103的每个用户的控制策略，以便在计算机103上仅可启动经授权的应用程序。

[0044] 图4显示了用于在计算机103上更新经授权程序的系统的示范性实施例。系统400包括更新模块201(其可以是发起程序更新的应用程序)、安装模块203(其可以是用于安装程序更新的应用程序)、管理代理端307和经授权对象的数据库308。在更新进程期间，系统使用安装包202(未示出)和新的以及经更新的程序文件204(未示出)。为了该系统400工作，应提前定义经授权应用程序的类别。而且，应为计算机103的所有用户生成策略以识别允许哪些用户在计算机103上利用哪些程序来工作。因此，根据分类，安装在计算机103上的需要

更新或者实施更新服务的所有软件都可放进经授权类别“用于更新的应用程序”中。可为该类别生成用于计算机103的所有用户的策略，其将允许用户仅使用经授权的程序。此外，策略中可提供的是，由经授权的程序或应用程序所创建的新的和经更新的程序文件204可被自动指定为“受信任的”以便该系统有效地发挥作用。因此，在一个实现方案中，系统将创建这样的规则，其将由经授权程序在更新进程期间所创建的新的或经更新的程序文件指定为受信任的。例如，在一个实现方案中，如果在更新进程的操作或者程序的操作期间创建这些文件，那么可将新参数“可信赖性(trustworthiness)”作为元数据添加到新的或者经更新的程序文件204。用于新的或经更新的程序文件204的这些以及其他的元数据参数，诸如这些文件的标识符，可存储在经授权对象的数据库308中用于继续更新进程。

[0045] 在另一个实现方案中，用于安装更新的计算机103上的所有软件可放置在单独的经授权类别“用于安装的应用程序”下。可为该类别生成用于所有计算机103的策略。该策略将允许那些应用程序实施安装功能。此外，为了该系统发挥作用，可以安装模块203也将成为受信任的方式生成用于该类别的策略，以给予模块203对其他受信任的、在来自经授权类别“用于更新的应用程序”的经授权应用程序的工作期间所生成的对象的访问权限。因此，在用于类别“用于安装的应用程序”的该策略内，可以存在假如提供到受信任的对象的访问权限则将指示安装模块203受信任的规则。于是，这些规则可允许创建用于安装模块203的新的程序文件204。在由模块201所发起的更新进程的生命期期间，那些文件也将被指定为受信任的。

[0046] 应该注意，上述类别可存储在计算机103上的经授权对象的数据库308中。这些类别可由分类模块305自动生成或由管理员生成。它们还可基于外部的专家知识，诸如由反病毒公司所给定的信息。

[0047] 图5示出了可由系统400(图4)所实施的、用于更新经授权程序的方法的示范性实现方案。在步骤501处，方法确定在计算机103上发起程序更新的更新模块201是否被允许在计算机103上执行。如上文所解释的，为了确定用户在何时尝试在计算机上启动软件程序，管理代理端307配置为监控它安装于其上的计算机103的资源。一旦管理代理端307检测到在计算机103上启动更新模块201的尝试，则代理端307还确定该更新模块201是否被授权以在计算机103上执行。为此，管理代理端307配置为识别更新模块201的元数据。管理代理端307可随后将该元数据中的至少一类(例如，更新模块201的名称)与存储在经授权对象的数据库308中的信息诸如经授权程序的类别以及相关联的策略相比较。基于比较的结果，管理代理端307确定更新模块201是否属于程序的经授权类别。在一个实现方案中，管理代理端307还可使用确定来自程序的经授权类别的该更新模块201是否可自动启动或者由计算机103的用户所启动。

[0048] 在步骤502处，如果管理代理端307基于关于经授权对象的信息而确定更新模块201未被授权在计算机103上执行，那么在步骤503处，管理代理端307阻止更新模块201的执行。如果有明确禁止在计算机103上执行特定更新模块201的策略，或者如果没有允许执行该更新模块201的策略，则可阻止该更新模块201。然而，在步骤502处，如果管理代理端307基于关于经授权对象的信息而确定更新模块201被授权在计算机103上执行，则在步骤504处，代理端307允许执行更新模块201。当模块201的元数据列在经授权类别“用于更新的应用程序”中或者列于在计算机103上的经授权对象的数据库308中所存储的相关联的策略中

时，可允许更新模块201在计算机103上执行。

[0049] 在更新模块201的执行期间，模块启动更新进程205，该更新进程205在计算机103上下载与更新模块201相关联的安装包202。由于更新模块201属于程序的经授权类别，因此用于计算机103的策略可将由更新模块201所启动的所有进程（例如，更新进程205）和作为更新模块的执行的结果而所创建/所下载的所有对象（例如，安装包202）指定为受信任的进程和对象。在一个示范性实现方案中，管理代理端307可在经授权对象的数据库308中对更新进程205的元数据和安装包202的元数据进行更新以包括“受信任的”的属性。然后，更新模块201将所下载的安装包202转移到安装模块203。

[0050] 在步骤505处，管理代理端307确定是否允许在计算机103上执行安装模块203。如上文所解释的，安装模块201启动在计算机103上安装安装包202的安装进程206。例如，Windows OS中的程序msiexec.exe可以是用于*.msi文件类型的安装包202的安装模块203。在一个实现方案中，管理代理端307配置为确定安装模块203的元数据以及将该元数据的至少一类（例如，安装模块203的名称）与来自经授权对象的数据库308的信息相比较，来自经授权对象的数据库308的信息诸如程序的经授权类别和相关联的策略。基于比较的结果，管理代理端307确定安装模块203是否属于程序的经授权类别，诸如类别“用于更新的应用程序”。在步骤507处，如果安装模块203被确定为被授权，则在步骤508处，管理代理端307将允许在计算机103上执行安装模块203。然而，在步骤506处，如果安装模块203被确定为未被授权，则在步骤507处，管理代理端307将阻止在计算机103上执行安装模块203。

[0051] 此外，管理代理端307可基于更新模块201和安装模块203之间的关系来确定是否允许或不允许在计算机103上执行安装模块203。例如，经授权对象的数据库308中的策略可包括这样的规则，该规则指示只有对象（或进程）列在程序的经授权类别中（或具有“受信任的”属性）才允许该对象（或进程）执行，并且该对象（或进程）由经授权的对象所创建，那么该子对象（或进程）就应被认为是受信任的。在这种情况下，如果管理代理端307确定安装模块203（或安装进程206）是更新模块201（或更新进程205）的子并且安装模块203（或安装进程206）与程序的经授权类别列在一起时，那么在步骤508处，管理代理端307将允许安装模块203的执行。然而，如果安装模块203并非由经授权的更新模块201所创建或者未与程序的经授权类别列在一起，那么在步骤507处，管理代理端307可阻止在计算机103上安装模块203的执行。

[0052] 在步骤508处，在执行期间，安装模块203启动安装进程506，其在计算机103上安装安装包202。由于安装包202属于受信任的程序类别（例如，具有“受信任的”属性的元数据）和/或安装模块203被认为是经授权的程序，因此用于计算机103的策略可将自受信任的、由经授权的安装模块203所执行的安装包202所创建的或由该安装包202所修改的所有更新程序文件204指定为受信任的进程和对象。在一个实现方案中，管理代理端307可在经授权对象的数据库308中更新新创建的或新修改的程序更新文件204的元数据以包括“受信任的”属性。该信任的转移允许更新系统400的平稳操作。

[0053] 在一个示范性实现方案中，当完成经授权应用程序计算机103的更新时，管理代理端307可连接到管理服务器102以将新的或经修改的程序文件转发到服务器，使得关于该应用程序的信息存储在数据存储303中用于将来使用。如上文所解释的，在一个示范性实现方案中，管理服务器102的控制模块301有权访问由反病毒公司所提供的远程的干净对象数据

库。因此，一旦控制模块102接收用于数据存储303的更新诸如新的或经修改的程序文件的元数据，就可将该信息转发到远程的干净对象数据库，如果后者缺少该信息的话。在另一个实现方案中，计算机103上的管理代理端307可直接连接到远程的干净对象数据库并且将关于新的或经修改的程序文件的信息发送到数据库。

[0054] 图6描绘了计算机系统5的一个示范性实施例，其可用于实现本文所述的用于更新经授权软件的系统和方法。如所示的，计算机系统5可包括由系统总线10所连接的一个或多个硬件处理器15、存储器20、一个或多个硬盘驱动器30、光学驱动器35、串行端口40、图形卡45、声卡50和网卡55。系统总线10可以是数个类型的总线结构中的任何一种，包括使用各种已知的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可包括一个或多个Intel® Core2Quad2.33GHz处理器或其他类型的微处理器。

[0055] 系统存储器20可包括只读存储器(ROM)21和随机存取存储器(RAM)23。可在如DRAM(动态RAM)、EPROM、EEPROM、闪存或其他类型的存储器架构中实现存储器20。ROM21存储基本输入/输出系统22(BIOS)，该BIOS包含有助于在计算机系统5的组件之间转移信息的基本例程，诸如在启动期间。RAM23存储诸如Windows® XP Professional或其他类型操作系统的操作系统24(OS)，该操作系统24在计算机系统5中负责程序的管理和协调以及硬件资源的分配和共享。存储器20也存储应用程序和程序25。存储器20还存储由程序25所使用的各种运行时数据26。

[0056] 计算机系统5可进一步包括诸如SATA磁性硬盘驱动器(HDD)的硬盘驱动器30，以及用于读取自或写入到诸如CD-ROM、DVD-ROM或其他光学媒介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媒介提供了实现本文所公开的算法和方法的计算机可读指令、数据结构、应用程序以及程序模块/子例程的非易失性存储。尽管示例性计算机系统5采用磁盘和光盘，但是本领域技术人员应该理解，可存储由计算机系统5可访问的数据的其他类型的计算机可读媒介，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，也可在计算机系统5的替代实施例中使用。

[0057] 计算机系统5进一步包括诸如通用串行总线(USB)的多个串行端口40，其用于连接数据输入设备75，诸如键盘、鼠标、触摸板以及其他设备。串行端口40还可用来连接数据输出设备80以及其他外围设备85，数据输出设备80诸如打印机、扫描仪和其他设备，其他外围设备85诸如外部数据存储设备等等。系统5还可包括声卡50，用于经由内部或外部扬声器65重现声音。此外，系统5可包括诸如以太网、WiFi、GSM、蓝牙或其他有线的、无线的或蜂窝网络接口的网卡55，用于将计算机系统5连接到诸如因特网的网络70。

[0058] 在各种实施例中，本文所述的系统和方法可在硬件、软件、固件或其任何组合中实现。如果在软件中实现，则方法可在非暂时性计算机可读介质上存储为一个或多个指令或代码。计算机可读介质包括数据存储。以示例的方式但并非限制，这类计算机可读介质可包括RAM、ROM、EEPROM、CD-ROM、闪速存储器或其他类型的电、磁或光存储介质、或任何其他可用来以指令或数据结构的形式承载或存储所期望的程序代码并可由计算机所访问的介质。

[0059] 为了清晰起见，本文没有公开系统和方法的所有常规特征。应该理解，在任何实际的实现方案的开发中，为了达到开发者的特定目标，必需制定大量特定于实现方案的决策，并且这些特定目标将针对不同的实现方案和不同的开发者而变化。应该理解，这类开发工作可能是复杂并且耗时的，但是其对于从本公开中受益的本领域普通技术人员而言仍然会

是常规的工程任务。

[0060] 此外，应理解的是，本文所使用的措辞或术语是为了描述而不是限制，使得本领域技术人员根据本文所呈现的教导和指导并结合相关领域技术人员的知识来解释本说明书的术语或措辞。此外，除非如此明确地予以阐述，否则对于说明书或权利要求中的任何术语都并非旨在将其归结为不常见的或特殊的意义。

[0061] 本文所公开的各种实施例包括本文以例示方式所指的已知组件现在的和将来的已知等同物。此外，虽然已示出并描述了实施例和应用，但对于从本公开中受益的本领域技术人员显而易见的是，比以上所提到内容更多的修改是可能的，而不脱离本文所公开的发明构思。

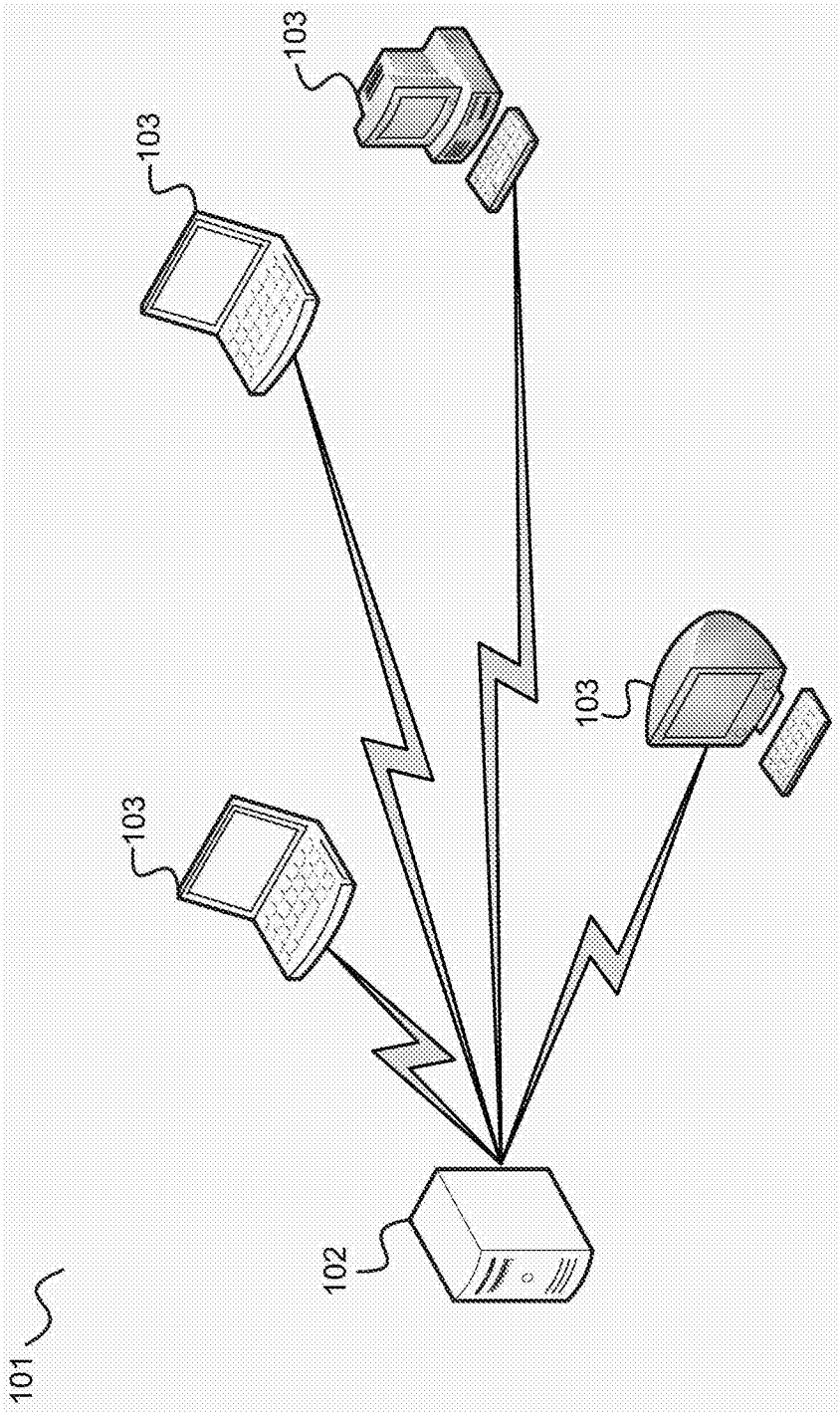


图1

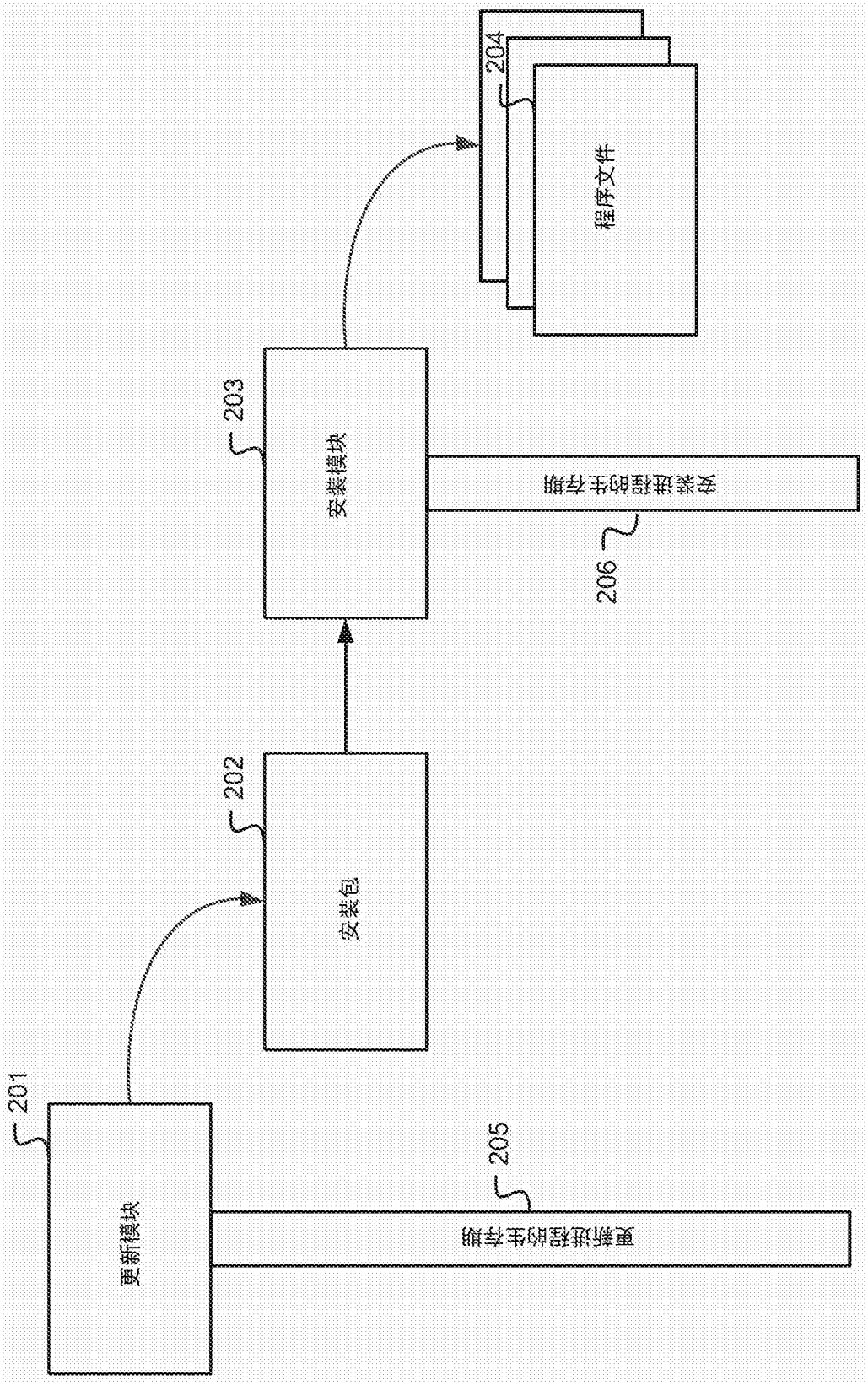


图2

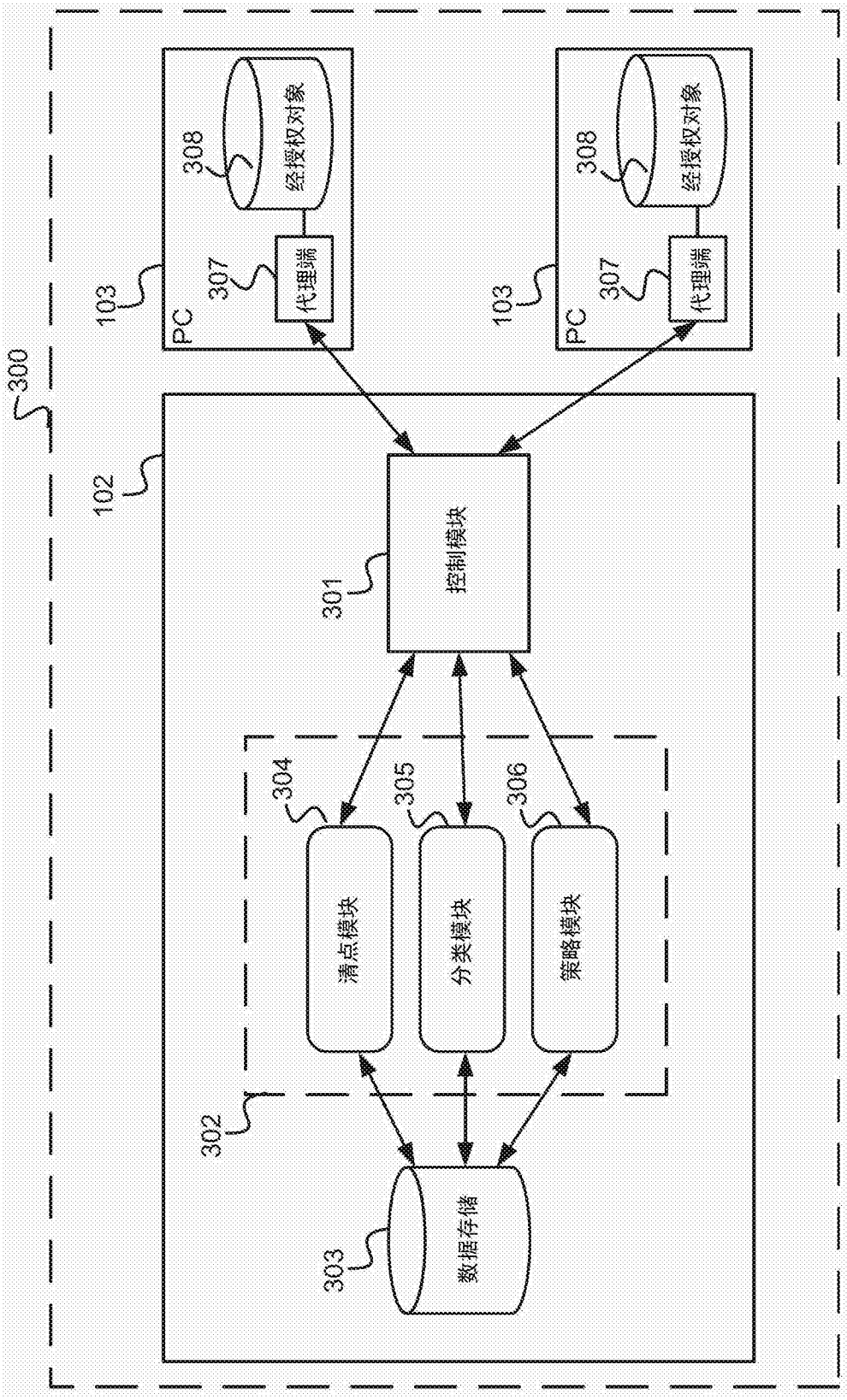

300
102
103
PC
307
代理端
308
经授权对象
103
PC
307
代理端
308
经授权对象
301
控制模块
302
304
清点模块
305
分类模块
306
策略模块
303
数据存储

图3

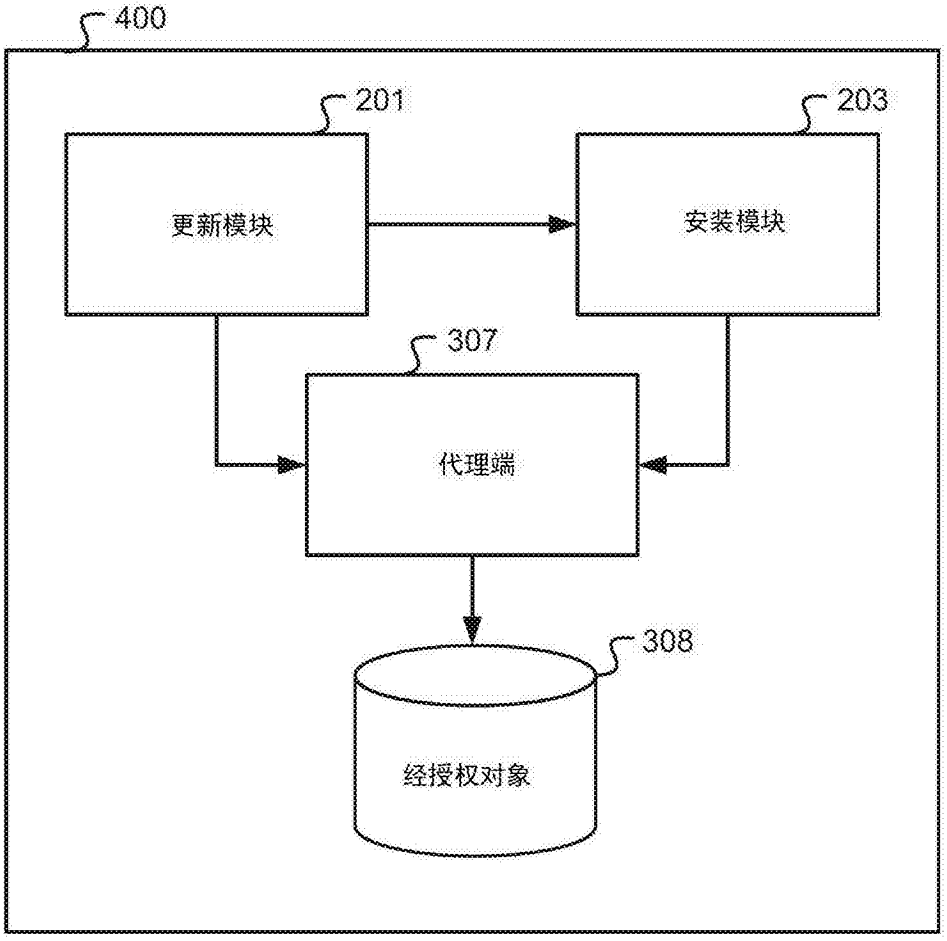


图4

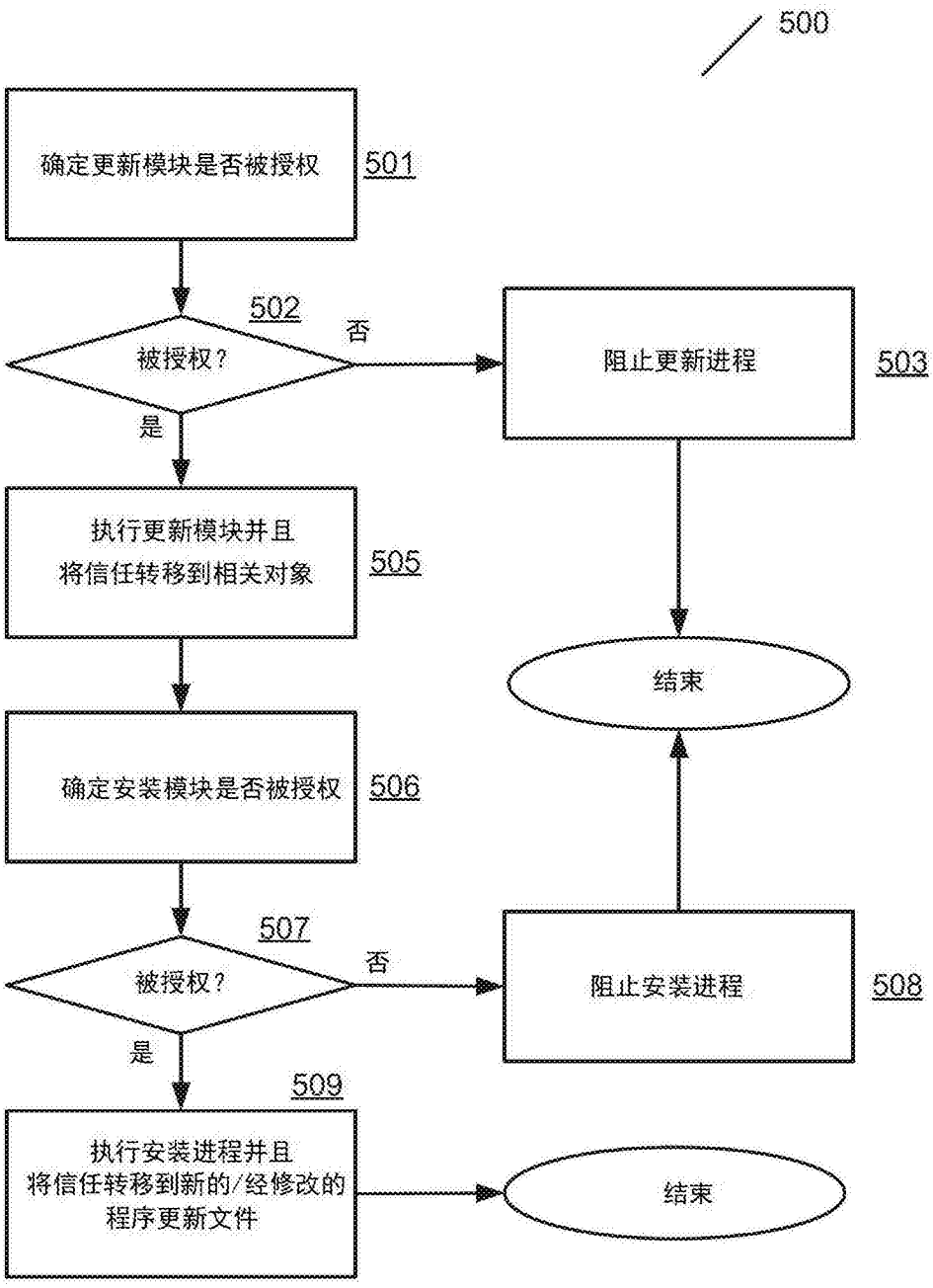

500
确定更新模块是否被授权
501
502
被授权？
否
是
阻止更新进程
503
执行更新模块并且
将信任转移到相关对象
505
结束
确定安装模块是否被授权
506
507
被授权？
否
是
阻止安装进程
508
509
执行安装进程并且
将信任转移到新的/经修改的
程序更新文件
结束

图5

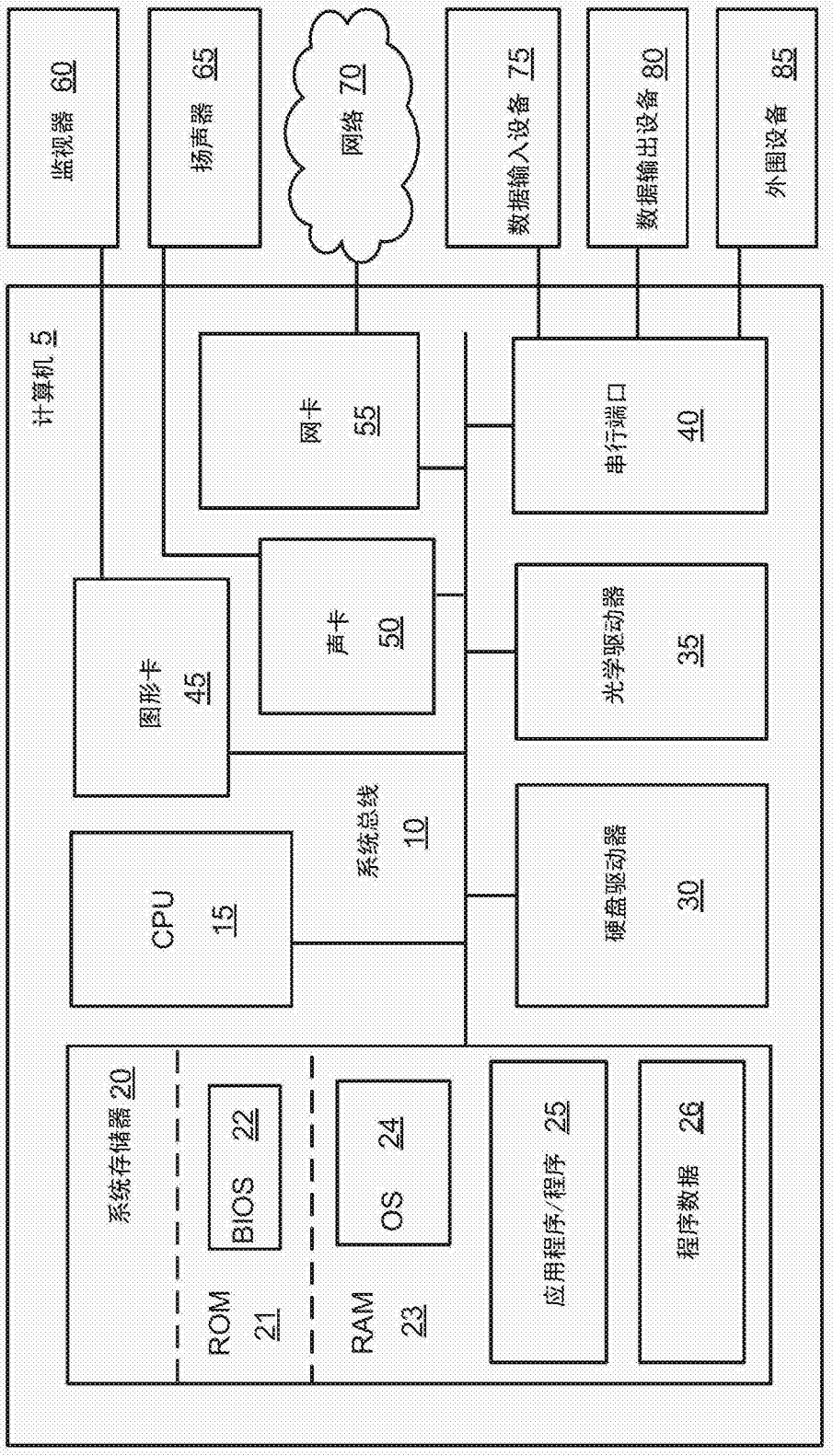

监视器 60
扬声器 65
网络 70
数据输入设备 75
数据输出设备 80
外围设备 85
计算机 5
网卡 55
串行端口 40
图形卡 45
声卡 50
光学驱动器 35
系统总线 10
CPU 15
硬盘驱动器 30
系统存储器 20
ROM 21
BIOS 22
RAM 23
OS 24
应用程序/程序 25
程序数据 26

图6