(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 110119619 B
(45)授权公告日 2023.08.04

(21)申请号 201811332077.3
(22)申请日 2018.11.09
(65)同一申请的已公布的文献号
申请公布号 CN 110119619 A
(43)申请公布日 2019.08.13
(30)优先权数据
2018104436 2018.02.06 RU
16/150,896 2018.10.03 US
(73)专利权人 卡巴斯基实验室股份制公司
地址 俄罗斯莫斯科
(72)发明人 谢尔盖·V·戈尔德契克
谢尔盖·V·索尔达托夫
康斯坦丁·V·萨普罗诺夫
(74)专利代理机构 北京同达信恒知识产权代理有限公司 11291
专利代理师 黄志华 何月华
(51)Int.Cl.
G06F 21/56(2013.01)
G06F 9/455(2006.01)
(56)对比文件
US 2014181897 A1,2014.06.26
CN 105874463 A,2016.08.17
CN 106557697 A,2017.04.05
CN 107103238 A,2017.08.29
CN 1885224 A,2006.12.27
审查员 石蒙蒙

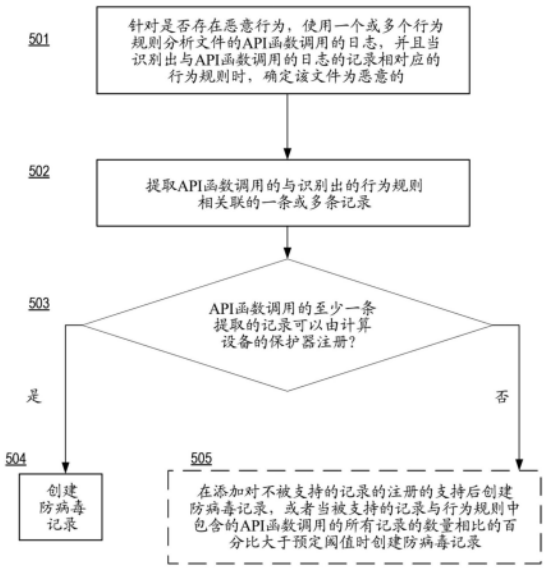
权利要求书3页 说明书13页 附图5页

(54)发明名称

创建防病毒记录的系统和方法

(57)摘要

本发明公开了创建防病毒记录的系统和方法。示例性方法包括:针对是否存在恶意行为,通过防止针对性攻击的保护器使用一个或多个行为规则分析文件的API函数调用的日志;当识别出与API函数调用的日志的记录相对应的行为规则时,确定所述文件为恶意的;提取与识别出的所述行为规则相关联的API函数调用的一条或多条记录;确定所述API函数调用的至少一条提取的记录是否可以由计算设备的保护器进行注册;以及当所述API函数调用的所述至少一条提取的记录可以由所述计算设备的保护器进行注册时,创建用于所述计算设备的保护器的防病毒记录,其中,创建的所述防病毒记录至少包括所述API函数调用的所提取的记录。



CN 110119619 B

1.一种创建防病毒记录的方法,所述方法包括:

针对是否存在恶意行为,通过防止针对性攻击的保护器使用一个或多个行为规则分析文件的API函数调用的日志;

当识别出与API函数调用的日志的记录相对应的行为规则时,通过所述防止针对性攻击的保护器确定所述文件为恶意的;

通过所述防止针对性攻击的保护器提取API函数调用的与识别出的所述行为规则相关联的一条或多条记录;

通过所述防止针对性攻击的保护器确定所述API函数调用的至少一条提取的记录是否能够由计算设备的保护器进行注册;

当所述API函数调用的所述至少一条提取的记录能够由所述计算设备的保护器进行注册时,通过所述防止针对性攻击的保护器创建用于所述计算设备的保护器的防病毒记录,其中,创建的所述防病毒记录至少包括所述API函数调用的所提取的记录;以及

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时,向所述计算设备的保护器添加对所述API函数调用的不被支持的记录的注册的支持。

2.根据权利要求1所述的方法,还包括:

在添加所述支持之后,创建用于所述计算设备的保护器的所述防病毒记录,其中,创建的所述防病毒记录包括所述API函数调用的所提取的记录。

3.根据权利要求1所述的方法,还包括:

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时,在所述API函数调用的被支持的记录与所述行为规则中包含的API函数调用的所有记录的数量相比的百分比大于预定的阈值时,创建用于所述计算设备的保护器的所述防病毒记录,创建的所述防病毒记录仅包括所述API函数调用的所述被支持的记录。

4.根据权利要求1所述的方法,其中,在所述防止针对性攻击的保护器上对所述文件的所述分析包括以下中的至少一项:使用信誉服务的检查、使用雅拉规则的检查、或者专家分析。

5.根据权利要求1所述的方法,其中,所述防止针对性攻击的保护器包括沙箱,并通过执行由所述文件启动的进程来填充API函数调用的所述日志,所述文件在所述沙箱中。

6.根据权利要求5所述的方法,其中,所述沙箱如以下项中的至少一者实现:在虚拟机上实现、基于文件系统和寄存器的部分虚拟化实现、以及基于所述文件系统和寄存器的访问规则实现。

7.一种用于创建防病毒记录的系统,包括:

至少一个处理器,所述至少一个处理器被配置为:

针对是否存在恶意行为,通过防止针对性攻击的保护器使用一个或多个行为规则分析文件的API函数调用的日志;

当识别出与API函数调用的日志的记录相对应的行为规则时,确定所述文件为恶意的;

提取API函数调用的与识别出的所述行为规则相关联的一条或多条记录;

确定所述API函数调用的至少一条提取的记录是否能够由计算设备的保护器进行注册;

当所述API函数调用的所述至少一条提取的记录能够由所述计算设备的保护器进行注

册时，创建用于所述计算设备的保护器的防病毒记录，其中，创建的所述防病毒记录至少包括所述API函数调用的所提取的记录；以及

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，向所述计算设备的保护器添加对所述API函数调用的不被支持的记录的注册的支持。

8.根据权利要求7所述的系统，还包括：

在添加所述支持之后，创建用于所述计算设备的保护器的所述防病毒记录，其中，创建的所述防病毒记录包括所述API函数调用的所提取的记录。

9.根据权利要求7所述的系统，还包括：

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，在所述API函数调用的被支持的记录与所述行为规则中包含的API函数调用的所有记录的数量相比的百分比大于预定的阈值时，创建用于所述计算设备的保护器的所述防病毒记录，创建的所述防病毒记录仅包括所述API函数调用的所述被支持的记录。

10.根据权利要求7所述的系统，其中，在所述防止针对性攻击的保护器上对所述文件的所述分析包括以下中的至少一项：使用信誉服务的检查、使用雅拉规则的检查、或者专家分析。

11.根据权利要求7所述的系统，其中，所述防止针对性攻击的保护器包括沙箱，并通过执行由所述文件启动的进程来填充API函数调用的所述日志，所述文件在所述沙箱中。

12.根据权利要求11所述的系统，其中，所述沙箱如以下项中的至少一者实现：在虚拟机上实现、基于文件系统和寄存器的部分虚拟化实现、以及基于所述文件系统和寄存器的访问规则实现。

13.一种非暂时性计算机可读介质，其上存储有用于创建防病毒记录的计算机可执行指令，所述非暂时性计算机可读介质包括用于以下操作的指令：

针对是否存在恶意行为，通过防止针对性攻击的保护器使用一个或多个行为规则分析文件的API函数调用的日志；

当识别出与API函数调用的日志的记录相对应的行为规则时，确定所述文件为恶意的；

提取API函数调用的与识别出的所述行为规则相关联的一条或多条记录；

确定所述API函数调用的至少一条提取的记录是否能够由计算设备的保护器进行注册；

当所述API函数调用的所述至少一条提取的记录能够由所述计算设备的保护器进行注册时，创建用于所述计算设备的保护器的防病毒记录，其中，创建的所述防病毒记录至少包括所述API函数调用的所提取的记录；以及

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，向所述计算设备的保护器添加对所述API函数调用的不被支持的记录的注册的支持。

14.根据权利要求13所述的非暂时性计算机可读介质，还包括用于以下操作的指令：

在添加所述支持之后，创建用于所述计算设备的保护器的所述防病毒记录，其中，创建的所述防病毒记录包括所述API函数调用的所提取的记录。

15.根据权利要求13所述的非暂时性计算机可读介质，还包括用于以下操作的指令：

当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，在所述API函数调用的被支持的记录与所述行为规则中包含的API函数调用的所有记录

的数量相比的百分比大于预定的阈值时，创建用于所述计算设备的保护器的所述防病毒记录，创建的所述防病毒记录仅包括所述API函数调用的所述被支持的记录。

16.根据权利要求13所述的非暂时性计算机可读介质，其中，在所述防止针对性攻击的保护器上对所述文件的所述分析包括以下中的至少一项：使用信誉服务的检查、使用雅拉规则的检查、或者专家分析。

17.根据权利要求13所述的非暂时性计算机可读介质，其中，所述防止针对性攻击的保护器包括沙箱，并通过执行由所述文件启动的进程来填充API函数调用的所述日志，所述文件在所述沙箱中。

创建防病毒记录的系统和方法

技术领域

[0001] 本发明涉及计算机安全领域，并且更具体地涉及用于创建防病毒记录的系统和方法。

背景技术

[0002] 传统的签名分析对于检测恶意文件效果不佳，所述恶意文件特别是多态病毒、混淆文件以及外壳代码(shellcode)。

[0003] 因此，现代防病毒应用程序还使用其它技术来检测恶意软件。例如，扫描可以使用所谓的“沙箱(sandbox)”。“沙箱”指的是专门与系统的其余部分隔离的环境。对于沙箱中执行的进程，对资源的访问和使用是受限制的。例如，可以在虚拟机上、基于文件系统和寄存器的部分虚拟化、基于对文件系统和寄存器的访问规则、或者基于混合算法来实现“沙箱”。在“沙箱”中执行被扫描的文件。在文件执行过程中，将关于API函数调用和系统事件(并且还包括正在分析、发送和接收的数据以及网络连接等)的记录(即信息)存储在调用日志(API函数调用的日志)中。防病毒应用还对获得的调用日志的记录所满足的行为规则(例如，已知的恶意行为模式)进行搜索。调用日志用于保存在文件执行期间由文件执行的API(application programming interface，应用程序编程接口)函数调用的记录。API函数调用或过程(procedure)调用(命令CALL，用于调用过程)被定义为用于执行过程(API函数)调用的无条件的控制转移。命令CALL执行堆栈中返回地址的存储以及用以执行API函数的转移。关于被调用的API函数的信息包括：发送到API函数的数据、由API函数返回的数据、调用API函数的进程、提供API函数的库/应用程序/内核、API函数的代码、调用API函数的地址、API函数所在的地址、返回地址等。调用日志还保存关于来自调用的API函数的返回命令(例如，从过程返回的RET命令)的信息。在执行来自API函数的返回命令时，从堆栈中提取返回地址并且发生到该返回地址的控制转移。通常，“沙箱”中文件的执行发生在有限的时间范围内(多达几十秒)。

[0004] 用于检测文件中的恶意功能的另一技术是使用仿真的技术。仿真涉及在执行代码期间在仿真器中模仿主机系统。

[0005] 上面描述的技术在现代防病毒技术中一起使用。通常，首先对文件执行签名分析。然后，如果在签名分析期间未发现恶意行为，则在仿真器或“沙箱”中执行该文件(鉴于防病毒软件制造商更强大的计算能力，在“沙箱”中的执行通常是由防病毒软件制造商完成的)。如果在仿真器或沙箱中的执行期间未发现恶意功能，则转移该文件以在用户的计算机上执行。然而，仍然有可能在仿真器或沙箱中的执行期间未发现恶意功能。为了提供更有效的保护，未知文件在用户计算机上的执行是在行为分析器(主动保护器)的监督下进行的。行为分析器，与“沙箱”和仿真器类似，在文件在用户的计算机上执行的过程中收集并分析API函数的调用日志。对于上述检测技术，由于它们工作原理的不同，调用日志和行为规则既可以具有共同的记录也可以具有单独的记录。行为分析器利用已安装的驱动程序/拦截器拦截恶意代码执行期间执行的API函数的调用以及拦截来自调用的API函数的返回命令，并将拦

截的调用和返回命令保存在调用日志中。然后，行为分析器在调用日志中针对行为规则(已知的恶意行为的模式)进行搜索并做出判定(例如计算机病毒、网络蠕虫、特洛伊木马程序(Trojan horse program)或附条件的有害软件)。行为分析器分析调用日志的原理类似于“沙箱”和仿真器的工作。然而，行为分析器对文件执行时间没有限制。行为分析器在其它方面是不同的。例如，仿真器和“沙箱”的检测技术和绕过(bypass)技术不适用于行为分析器，因为文件是在用户的计算机上执行，而不是在隔离的环境中或仿真器中执行。当行为分析器正在执行文件时，用户的计算机可以正在执行其有效载荷。因此，恶意文件可能在被行为分析器检测到并破坏之前损害系统。

[0006] 为了进行更细致的分析，“沙箱”通常位于服务器上，该服务器具有更强的计算能力并且具有更长的时间执行被研究的文件。在“沙箱”中分析到达防病毒公司的未知的可疑文件。如果该分析识别出与行为规则相对应的行为，则该文件被判定为恶意的。然后，分析者可以创建用于通过用户的计算设备上的保护器(保护模块，例如，防病毒)来检测该文件的防病毒记录或行为规则——例如，分析者可以更新：用于签名分析的防病毒记录、用于行为分析器的行为规则、或者用于仿真器的行为规则。然而，在虚拟机上检测恶意文件和由分析者创建用于防病毒应用(即，计算设备的保护器)的防病毒记录之间可能有相当长的时间间隔。因此，需要解决的技术问题是：基于识别出的与虚拟机(“沙箱”)的调用日志的记录相对应的行为规则来为计算设备的保护器及时地创建防病毒记录。

发明内容

[0007] 公开了用于创建防病毒记录的系统和方法。在一个示例性方面，提供了一种用于创建防病毒记录的系统，所述系统包括防止针对性攻击的保护器的硬件处理器，所述硬件处理器被配置为：针对是否存在恶意行为，使用一个或多个行为规则分析文件的API函数调用的日志；当识别出与API函数调用的日志的记录相对应的行为规则时，确定所述文件为恶意的；提取API函数调用的与识别出的所述行为规则相关联的一条或多条记录；确定所述API函数调用的至少一条提取的记录是否可以由计算设备的保护器进行注册；以及当所述API函数调用的所述至少一条提取的记录可以由所述计算设备的保护器进行注册时，创建用于所述计算设备的保护器的防病毒记录，其中，创建的所述防病毒记录至少包括所述API函数调用的所述提取的记录。

[0008] 在一个示例性方面，提供了一种在包括硬件处理器的计算机中实现的方法，所述方法包括：针对是否存在恶意行为，使用一个或多个行为规则分析文件的API函数调用的日志；当识别出与API函数调用的日志的记录相对应的行为规则时，确定所述文件为恶意的；提取API函数调用的与识别出的所述行为规则相关联的一条或多条记录；确定所述API函数调用的至少一条提取的记录是否可以由计算设备的保护器进行注册；以及当所述API函数调用的所述至少一条提取的记录可以由所述计算设备的保护器进行注册时，创建用于所述计算设备的保护器的防病毒记录，其中，创建的所述防病毒记录至少包括所述API函数调用的所述提取的记录。

[0009] 在一个示例性方面，在所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，所述方法还包括向所述计算设备的保护器添加对所述API函数调用的不被支持的记录的注册的支持。

[0010] 在一个示例性方面，所述方法还包括在添加所述支持之后，创建用于所述计算设备的保护器的所述防病毒记录。创建的所述防病毒记录包括所述API函数调用的所述提取的记录。

[0011] 在一个示例性方面，当所述计算设备的保护器不支持所述API函数调用的所述至少一条提取的记录的注册时，所述方法还包括在所述API函数调用的被支持的记录与所述行为规则中包含的API函数调用的所有记录的数量相比的百分比大于预定的阈值时，创建用于所述计算设备的保护器的所述防病毒记录。创建的所述防病毒记录仅包括所述API函数调用的所述被支持的记录。

[0012] 在一个示例性方面，在所述防止针对性攻击的保护器上对所述文件的所述分析包括以下中的至少一项：使用信誉服务的检查、使用雅拉(YARA)规则的检查、或者专家分析。

[0013] 在一个示例性方面，所述防止针对性攻击的保护器包括沙箱，并通过在所述沙箱中执行由所述文件启动的进程来填充API函数调用的所述日志。

[0014] 在一个示例性方面，所述沙箱如以下项中的至少一者实现：在虚拟机上实现、基于文件系统和寄存器的部分虚拟化实现、以及基于对所述文件系统和寄存器的访问规则实现。

[0015] 以上对本发明的示例性方面的简要概述用于提供对本发明的教导的基本理解。本概述不是对所有预期方面的广泛综述，并且既不旨在标识所有方面的关键的或重要的要素，也不旨在描绘本发明的教导的任何方面或所有方面的范围。为了实现前述内容，本发明的一个或更多个方面包括在权利要求中所描述和示例性指出的特征。

[0016] 如上所述，根据本发明的教导创建的防病毒记录，有利地提高了对计算设备的保护。

[0017] 此外，增加了可识别的恶意文件的数量。例如，通过基于识别出的与虚拟机的调用日志的记录相对应的行为规则创建用于计算设备的保护模块的防病毒记录，增加了可识别的恶意文件的数量。

[0018] 在又一优势中，和与其它方法相关的时间相比，缩短了用于创建防病毒记录的时间。例如，通过基于识别出的与虚拟机的调用日志的记录相对应的行为规则创建用于保护器的防病毒记录，有利地缩短了创建防病毒记录所需的持续时间。

附图说明

[0019] 并入本说明书并构成本说明书的一部分的附图示出了本发明的一个或多个示例性方面，并且与具体实施方式一起用于阐述这些示例性方面的原理和实现方式。

[0020] 图1示出了根据本发明的教导的用于创建防病毒记录的信息系统。

[0021] 图2为示出示例性“沙箱”的框图。

[0022] 图3示出了计算设备的保护器的示例性框图。

[0023] 图4示出了用于防止针对性攻击的保护器的示例性框图。

[0024] 图5示出了根据本发明的教导的创建防病毒记录的方法的流程图。

[0025] 图6示出了可以在其上实现本发明的各个方面的通用计算机系统的示例。

具体实施方式

[0026] 本文在用于创建防病毒记录的系统、方法和计算机程序产品的范畴内对示例性方面进行了描述。本领域的普通技术人员将认识到,以下描述仅是说明性的,并不旨在以任何方式进行限制。其它方面将很容易将其自身暗示给了解本发明的优点的本领域的技术人员。现在将详细参考如附图中所示的示例性方面的实现。在整个附图和以下描述中将尽可能使用相同的附图标记来指代相同或相似的项目。

[0027] 术语表

[0028] 为了增加清楚性,首先在此提供在描述本发明的一个或多个示例性方面时所使用的术语。

[0029] 感染指示(IOC,较少地,称为感染指标)是可在计算机或网络上观察到的对信息系统的侵入的人工标志或残留标志。通常感染指示是防病毒记录、IP地址、文件的校验和、URL地址以及僵尸网络命令中心的域名等。存在许多用于感染指示的标准,特别是:

[0030] • OpenIOC

[0031] (http://blogs.rsa.com/understanding-indicators-of-compromise-ioc-part-i/,http://openioc.org/),

[0032] • STIX(https://stix.mitre.org/),

[0033] • CybOX(https://cybox.mitre.org)以及其它标准。

[0034] 计算机攻击(也称为网络攻击)是软件和硬件对信息系统和信息电信网络的针对性动作,执行计算机攻击的目的是为了破坏这些系统和网络中的信息安全性。

[0035] 针对性攻击(TA)是针对特定组织或特定个体的计算机攻击的特例。

[0036] 模糊哈希或柔性指纹(flexible fingerprint)(局部敏感哈希)是文件指纹,该文件指纹对于形成该文件指纹的文件中的变化是稳定的。也就是说,在检测到恶意文件时,恶意文件的指纹的值也将用于检测许多类似的(可能未知)的恶意文件。这种指纹的主要特征是它对镜像文件变化的不变性,例如,请参见专利号US8955120。

[0037] 模糊判定是保护器(防病毒应用程序)在文件执行期间检测可疑动作时的响应,所述模糊判定诸如指示文件具有恶意文件的特征的判定。例如,在借助于柔性指纹检测文件时,触发模糊判定。该模糊判定指示被检测的文件是恶意的并且还为所述判定提供了一定程度的概率。

[0038] 回到对本发明的教导的描述,图1示出了根据本发明的教导的用于创建防病毒记录的信息系统。信息系统100包括至少一个计算设备101(也简称为计算机),所述至少一个计算设备101通信地连接到计算机网络109。计算设备101包括标准计算设备(例如,个人计算机、笔记本、智能电话、网络设备(例如,路由器、交换机、集线器))等。应当注意,计算设备101可以是物理设备或在物理设备上操作的软件(诸如虚拟机)。可以使用本领域中通常已知的任一网络拓扑(诸如以下类型中的任一类型的网络拓扑:完全连接类型的网络拓扑、总线类型的网络拓扑、星形类型的网络拓扑、环形类型的网络拓扑、蜂窝或混合类型的网络拓扑)109来组织信息系统100。

[0039] 可以在计算设备101上安装计算设备的保护器102(保护模块)。应当注意,如果信息系统100包括两个或更多个计算设备101,则在一些计算设备101上可以不安装保护器102。

[0040] 信息系统100还包括防止针对性攻击的保护器103，防止针对性攻击的保护器103例如可以位于单独的服务器上。该单独的服务器可以通过计算机网络109连接到至少一个计算设备101。代理服务器(未示出)可以用于通过网络109将计算设备101连接到因特网和检测器110。防止针对性攻击的保护器103可以包括：威胁数据库105、调用日志104以及虚拟机106，将在下面对威胁数据库105、调用日志104以及虚拟机106进行进一步详细描述。

[0041] 所述系统还包括其上运行有保护器102的至少一个计算设备101(作为示例，图1示出了两个保护器102，在每个计算设备101上运行有一个保护器102)。

[0042] 保护器102可以包括计算设备101的调用日志107和威胁数据库108。保护器102的各个模块(诸如仿真器、行为分析器等，更多细节参见图3)可以在被研究文件的执行期间，将关于API函数调用的记录(诸如函数名称、发送的参数、函数调用的时间)注册在调用日志107中。

[0043] 在示例性方面，用于API函数调用的调用日志107的每条记录都包含以下信息：

[0044] • 被调用函数的标识符(诸如名称)；

[0045] • 由该文件启动的进程的唯一标识符(进程标识符，PID)；

[0046] • 执行进程地址空间的指令的线程的唯一标识符(线程标识符，TID)；

[0047] • 上述函数的参数集；以及

[0048] • 函数调用的时间。

[0049] 用于保护器102的防病毒记录可以被定义为用于相应保护器102的签名和/或规则(诸如实时访问扫描器(on-access scanner)的防病毒记录、用于行为分析器的行为规则、或者仿真器的行为规则)。在签名的情况下，防病毒记录可以构成例如文件代码段的哈希和、来自文件代码的不同段的哈希和的集合以及用于选择这些哈希和的规则(例如，如果遇到签名中包含的三个哈希和中的两个哈希和，则认为遇到了整个签名)等。用于行为分析器的行为规则以及仿真器的行为规则的示例将在下面给出。

[0050] 检测器110(检测模块)可以位于远程服务器上，该远程服务器经由计算机网络109连接到保护器102-保护器103，并且用于执行文件的更详细分析。

[0051] 图2是示出了示例性“沙箱”的框图。该“沙箱”构成了用于安全执行进程的计算机环境。在本发明的示例性方面，例如，可以以虚拟机的形式、基于文件系统和寄存器的部分虚拟化、基于对文件系统和寄存器的访问规则、或者基于混合算法来实现“沙箱”。图2示出了以虚拟机106的形式实现的“沙箱”的示例，在虚拟机106上安装了操作系统(OS)131。虚拟机106被设计为在防止针对性攻击的保护器103的控制下执行文件133。应当注意，文件133可以为任何给定数据格式的文件并且可以包含任何给定的数据。

[0052] 在执行文件133期间，包含在虚拟机106中的日志组件132拦截API函数调用并相继输入关于API函数调用的记录。将该记录输入调用日志134中，调用日志134同样位于虚拟机106中。虚拟机的调用日志134包括在虚拟机上执行文件期间的关于至少API函数调用的注册的记录。

[0053] API函数调用或过程调用(命令CALL，用于调用过程)被定义为执行过程(API函数)调用的无条件的控制转移。命令CALL执行堆栈中返回地址的存储以及用以执行API函数的转移。关于被调用的API函数的信息包括：发送到API函数的数据、由API函数返回的数据、调用API函数的进程、提供API函数的库/应用程序/内核、API函数的代码、调用API函数的地

址、API函数所在的地址、返回地址等。调用日志还保存关于来自调用的API函数的返回命令(从过程返回的RET命令)的信息——在执行来自API函数的返回命令时,从堆栈中提取出返回地址并且发生到该返回地址的控制转移。“沙箱”中文件的执行通常发生在有限的时间范围内(多达几十秒)。

[0054] 在示例性方面,用于API函数调用的调用日志134的每条记录(以及,相应地,调用日志104的每条记录)都包含以下信息:

[0055] • 被调用函数的标识符(诸如名称);

[0056] • 由该文件133启动的进程的唯一标识符(进程标识符,PID);

[0057] • 执行进程地址空间的指令的线程的唯一标识符(线程标识符,TID);以及

[0058] • 上述函数的参数集。

[0059] 在又一示例性方面,在虚拟机106上执行文件133期间,还注册以下信息:

[0060] • 用于从API函数到返回地址的控制转移的命令;

[0061] • 直接的Windows NT Native API函数调用;

[0062] • 用于从Windows NT Native API函数调用返回的命令;

[0063] • 更改计算机系统状态的方法,特别是关闭或重启计算机系统的事件;

[0064] • 操作系统中的事件;

[0065] • 感染指示;

[0066] • 系统调用;

[0067] • 保护器的判定;

[0068] • 文件的校验和或文件的一部分的校验和;

[0069] • 文件的来源;

[0070] • 文件执行的仿真结果;

[0071] • 文件在计算设备上出现的时间;

[0072] • 文件通过网络获取的数据;

[0073] • 文件通过网络发送的数据;

[0074] • 计算机上的DNS劫持;

[0075] • 操作系统的自动更新的断开;

[0076] • 防火墙的断开;

[0077] • 保护器的断开;

[0078] • “用户账户控制”组件的断开;

[0079] • 操作系统的“系统恢复”组件的断开;

[0080] • 文件管理器中的“显示隐藏文件、文件夹和磁盘”选项的断开;

[0081] • 防火墙规则的更改;

[0082] • 主机文件的更改;以及

[0083] • 文件的自动删除。

[0084] 行为规则和防病毒记录还包括上述信息。

[0085] 如果发生以下退出条件中的一个退出条件,则文件133在虚拟机130上的执行完成:

[0086] • 进程完成;

[0087] • 已经过指定的时间段；

[0088] • 之前的步骤已经执行了给定数量的重复次数(诸如3)；以及

[0089] • 由进程执行的指令数量超过指令数量的给定阈值(诸如100000条指令)。

[0090] 如果满足退出条件，即，在虚拟机106上已经完成了文件133的执行，则保护器103保存包含在虚拟机中的调用日志134的记录，其中，记录被保存在调用日志104中，该调用日志104包含在运行保护器103的操作系统中。

[0091] 在又一示例性方面，日志组件132拦截记录并将记录输入到调用日志134中，所述记录包含关于负责与网络、寄存器、文件系统、RAM、进程和线程一起工作的操作的系统调用的信息。

[0092] 应当注意，当日志组件132拦截API函数调用时，日志组件132具有访问调用栈和在处理器上正在执行的指令的权限。在另一示例性方面，日志组件132将关于异常和在处理器上正在执行的指令的记录保存在调用日志134中。

[0093] 在又一示例性方面，日志组件132被设计为识别以下异常：

[0094] • 在执行调用指令时违反访问权限；

[0095] • 在写入内存时违反访问权限；

[0096] • 在命令计数器的地址以及在命令计数器的地址一致的地址处，以给定精度从内存读取时违反访问权限；

[0097] • 违反数据执行预防政策；

[0098] • 堆栈上的缓冲区溢出；

[0099] • 动态内存损坏(堆损坏)；以及

[0100] • 尝试执行禁止指令、错误指令或特权指令。

[0101] 例如，日志组件132可以根据对寄存器、系统结构以及截取API函数调用KiUserExceptionDispatcher时的调用栈的内容的分析，识别“写入内存时违反访问权限”的异常。

[0102] 保护器103被设计为识别威胁数据库105中包含的行为规则中的与调用日志104的记录相对应的行为规则。该行为规则包括在触发行为规则的情况下关于至少一个API函数的调用和判定(例如，计算机病毒、因特网蠕虫、特洛伊木马程序或附条件的有害软件)的信息。在示例性方面，行为规则包括API函数调用集合以及关于该API函数调用集合的逻辑表达式。例如，如果已经使用如由行为规则所指定的特定参数来调用特定API函数，则保护器103可以在调用日志104中找到该行为规则。在另一示例性方面，行为规则可以另外包括施加于该相同行为规则的其它记录的条件。这样的条件例如可以是检查防病毒记录的记录数量、检查防病毒记录的记录连续顺序等。

[0103] 因此，当在虚拟机和仿真器上执行时，行为分析器和“沙箱”中都使用行为规则。行为规则的差异可以涉及关于API函数调用的信息以及施加于API函数调用的条件。例如，行为分析器可以从列表中注册关于API函数调用的记录，该列表可以完全或部分地与可以由虚拟机或仿真器注册的API函数调用的列表不同。同时，在另一示例性方面，行为分析器、虚拟机和仿真器的所述API函数调用的列表可以一致。

[0104] 图3示出了计算设备的保护器的示例性框图。计算设备的保护器102可以依次包括被设计为用于确保计算设备101的信息安全性的多个模块(特定模块)：实时访问扫描器、按

需扫描器、电子邮件防病毒、网络(web)防病毒、行为分析器(主动防护模块)、HIPS模块(Host Intrusion Prevention System,主机入侵防御系统)、数据丢失防护(Data Loss Prevention,DLP)模块、漏洞扫描器、仿真器、防火墙等。在本发明的示例性方面,这些特定模块可以是保护器102的一体部分。在又一示例性方面,这些模块可以以单独的程序组件的形式实现。

[0105] 实时访问扫描器包含用于检测在用户的计算机系统上被打开、被启动或被保存的所有文件的恶意活动的程序。按需扫描器与实时访问扫描器的不同之处在于按需扫描器扫描用户指示的文件和目录,即根据用户的要求扫描文件和目录。

[0106] 电子邮件防病毒用于检查收到和发出的电子邮件是否包含恶意文件。网络防病毒用于防止可能包含在用户访问的网站上的恶意代码的执行,并且还阻止网站的打开。HIPS模块用于检测有害的和恶意的程序活动,并在执行时阻止此类活动。DLP模块用于检测和防止来自计算机或网络的机密数据的丢失。漏洞扫描器用于检测计算设备101上的漏洞(例如,保护器102的某些组件已经被关闭、过时的病毒数据库、网络端口已经关闭等)。防火墙根据指定的规则提供对网络业务的检查和过滤。仿真器的任务为在仿真器中执行代码期间模拟主机系统。行为分析器在文件执行过程中使用行为规则来检测正在被执行的文件的行为并按照信任级别对这些文件进行分类。行为分析器在调用日志107中搜索关于API函数调用的与威胁数据库108的行为规则对应的至少一条注册的记录。在特定示例性方面,调用日志107和威胁数据库108位于计算设备101上。

[0107] 行为规则包括在触发该行为规则的情况下关于至少一个API函数的调用和判定(例如,计算机病毒、因特网蠕虫、特洛伊木马程序或附条件的有害软件)的记录。在另一示例性方面,行为规则包括:API函数调用集合以及关于该API函数调用集合的逻辑表达式。例如,如果已经使用如由所述行为规则指定的特定参数来调用特定API函数,则保护器102可以在调用日志107中找到该行为规则。

[0108] 行为规则还与在触发该规则时做出的判定(即,与该行为规则对应的最可能的恶意或有害软件的类别)相对应。例如,判定可以如下:计算机病毒、因特网蠕虫、特洛伊木马程序或附条件的有害软件。

[0109] 在另一示例性方面,(行为分析器、仿真器以及“沙箱”的)所述行为规则至少包括以下内容:

[0110] • 来自可疑API函数列表的API函数的调用(例如,所述列表可以包含以下API函数:WinExec、CreateProcess、GetFileSize、CreateFile);

[0111] • API函数GetFileSize的调用已经执行了10次;

[0112] • 在调用API函数WriteFile(写入文件)之后调用API函数WinExec(启动用于执行的文件);

[0113] • 计算机上的DNS劫持;

[0114] • 操作系统的自动更新的断开;

[0115] • 防火墙的断开;

[0116] • 保护器的断开;以及

[0117] • UAC(用户账户控制(User Account Control)——Windows OS的组件)的断开。

[0118] 在检测到恶意软件(可疑行为、垃圾邮件和计算机威胁的其它标志)时,图3所示的

模块创建相应的通知(然后该通知可被转换为保护器102的判定),从而通知保护器关于检测到的威胁以及需要采取的用于除掉该威胁的动作(诸如,删除或修改文件、禁止执行等)。在另一示例性方面,已经检测到恶意软件的保护器本身可以执行用于除掉威胁的动作。在又一示例性方面,判定可以是模糊判定或测试性判定(因为该判定可能会产生错误警报)——在这种情况下,保护器将不执行用于除掉威胁的动作,而是将通知发送给远程服务器(图中未示出)。在另一示例性方面,恶意软件包括以下类别(与类别对应的判定):恶意软件和附条件的有害软件。恶意软件可以具有子类别:病毒、蠕虫、特洛伊木马程序、加壳程序、恶意实用程序。附条件的有害软件为广告软件(广告程序)、涉及色情内容的软件(色情程序)、其使用可能对计算机造成伤害的合法软件(风险程序)以及其它软件。

[0119] 图4示出了用于防止针对性攻击的保护器("防止针对性攻击的保护器(protector against targeted attacks)")的示例性框图。位于服务器上的防止针对性攻击的保护器103例如可以包含以下模块:"沙箱"、入侵检测系统(Intrusion Detection System,IDS)、信誉服务、用于校验YARA规则的模块、防病毒模块、风险评分模块、分析器、DLP模块和其它检测器。

[0120] "沙箱"模块具有类似于上述计算设备的保护器102的仿真器的功能,不同之处在于"沙箱"可以使用额外的计算能力并且工作更长时间。防止针对性攻击的保护器103则没有时间限制。相反,时间限制是计算设备的保护器102中固有的。

[0121] "沙箱"是用于安全执行进程的计算机环境,且用于在由文件启动的进程的执行期间确定可疑活动。

[0122] 例如,可以以虚拟机的形式(请参见图2所示的示例性方面)、基于文件系统和寄存器的部分虚拟化、基于对文件系统和寄存器的访问规则、或者基于混合方法来实现"沙箱"。

[0123] 入侵检测系统是识别以下项的模块:对计算设备101或网络109的未经授权的访问或对计算设备101或网络109的未经授权的控制。

[0124] 信誉服务可以包含关于计算设备101上的文件的普及度的信息(具有该文件的计算设备101的数量、文件启动的数量等)。

[0125] 用于校验YARA规则的模块用于校验YARA签名——开放式签名格式(请参见http://yararules.com/)。

[0126] DLP模块用于检测来自计算机或网络的机密数据并防止来自计算机或网络的机密数据的丢失。

[0127] 位于服务器上的防止针对性攻击的保护器103将被研究的文件133发送到日志组件132以用于在虚拟机106中执行。在执行文件133期间,日志组件132在调用日志134中至少注册关于API函数调用的记录(例如,函数的名称、发送的参数、函数调用的时间)。因此,虚拟机的调用日志134将至少包括在虚拟机106上执行文件133期间注册的关于API函数调用的记录。如上所述,在虚拟机106上完成文件133的执行之后,保护器103将包含在虚拟机中的调用日志134的记录保存在调用日志104中。

[0128] 本发明在以下情况下开始工作:已经从威胁数据库105中识别出与调用日志104的记录相对应的行为规则,即已经借助于虚拟机检测到恶意文件。

[0129] 就像用于行为分析器的行为规则一样,用于虚拟机的行为规则在触发该行为规则的情况下也包括关于至少一个API函数调用及其判定(例如,计算机病毒、因特网蠕虫、特洛

伊木马程序或附条件的有害软件)的信息。在一个示例性方面,行为规则特别包括API函数调用集合以及关于该API函数调用集合的逻辑表达式。例如,如果已经使用如由所述行为规则指定的特定参数来调用特定API函数,则保护器103将在调用日志104中找到该规则。

[0130] 行为规则还与在触发该规则时做出的判定(即,与该规则对应的最可能的恶意或有害软件的类别)相对应。例如判定可以如下:计算机病毒、因特网蠕虫、特洛伊木马程序或附条件的有害软件。

[0131] 在一个示例性方面,所述行为规则包括以下内容:

[0132] • 来自可疑API函数列表的API函数的调用(例如,所述列表可以包含以下API函数:WinExec、CreateProcess、GetFileSize、CreateFile);

[0133] • API函数GetFileSize的调用已经执行了10次;

[0134] • 在调用API函数WriteFile(写入文件)之后调用API函数WinExec(启动用于执行的文件);

[0135] • 计算机上的DNS劫持;

[0136] • 操作系统的自动更新的断开;

[0137] • 防火墙的断开;

[0138] • 保护器的断开;以及

[0139] • UAC(用户账户控制——Windows OS的组件)的断开。

[0140] 图5示出了根据本发明的教导的创建防病毒记录的方法的流程图。根据本发明的一个方面,在步骤501中,所述方法通过以下方式确定文件是恶意的:针对是否存在恶意行为,使用一个或多个行为规则分析文件的API函数调用的日志,并且当识别出与API函数调用的日志的记录相对应的行为规则时,确定该文件为恶意的。例如,当识别出与API函数调用的记录相对应的行为规则时,确定图2中示出的文件133为恶意的。例如,检查文件的API函数调用的日志以确定是否存在至少一个恶意行为(即,是否存在一个或多个行为规则中的至少一个行为规则)。在步骤502中,所述方法提取API函数调用的与识别出的行为规则相关联的一条或多条记录。然后所述方法进行到步骤503。在步骤503中,所述方法确定API函数调用的至少一条提取的记录是否由计算设备的保护器注册。使用防止针对性攻击的保护器进行确定。当计算设备的保护器注册了API函数调用的所述至少一条提取的记录时,所述方法进行到步骤504。否则,所述方法进行到可选步骤505。在步骤504中,当可以注册API函数调用的所述至少一条提取的记录时,所述方法创建用于计算设备的保护器102的防病毒记录。创建的防病毒记录至少包括所提取的API函数调用的记录。在一个方面,创建的防病毒记录包括所提取的关于API函数调用的记录。在步骤505中,所述方法在添加对不被支持的记录的注册的支持后创建防病毒记录,或者当被支持的记录与行为规则中包含的API函数调用的所有记录的数量相比的百分比大于预定的阈值时创建防病毒记录。

[0141] 因此,通过为保护器102创建防病毒记录,解决了所提出的技术问题。即,由于为保护器102创建的防病毒记录使得可以检测以前因为缺少相应的防病毒记录而可能无法检测到的恶意软件,因此提高了恶意文件的检测质量。和与以前已知方法相关的时间相比,用于为保护器102创建防病毒记录的时间也缩短了。

[0142] 在一个方面,为位于远程服务器上的检测器110创建行为规则。在另一方面,为计算设备101上实例化的保护器102创建防病毒记录。在又一方面,防病毒记录指的是用于相

应的保护器102的一个或多个签名和/或一条或多条规则。在又一方面，防病毒记录可以包括以下项的记录：实时访问扫描器、用于行为分析器的行为规则、或者用于仿真器的行为规则。

[0143] 在一个方面，对防止针对性攻击的保护器上的文件的分析包括以下项中的至少一者：使用信誉服务的检查、使用YARA规则的检查、或者专家分析。专家分析可以为由计算机安全领域的专家对文件进行人工分析。

[0144] 在另一方面，如果保护器102(安装在计算设备101上)不支持关于API函数调用的至少一条提取的记录的注册，则向保护器102添加对关于API函数调用的不被支持的记录的注册的支持。在添加支持之后，创建的用于保护器102的防病毒记录包括提取的关于API函数调用的记录。例如，在虚拟机106中，日志组件132可以支持关于API函数调用GetFileSize的记录的注册，然而，计算设备101上的保护器102的行为分析器可以不支持该函数调用的注册。然而，行为规则包括上述API函数调用的记录(例如，行为规则如下：“API函数调用GetFileSize被执行了10次”)。因此，向保护器102添加对关于API函数调用GetFileSize的记录的注册的支持。然后，为保护器102创建防病毒记录(在给定的示例中，是指用于行为分析器的行为规则)，该防病毒记录包括所提取的关于API函数调用的记录。

[0145] 在又一示例性方面，如果保护器102(安装在计算设备101上)不支持关于API函数调用的所述至少一条提取的记录的注册，则可以创建仅包括关于API函数调用的被支持的记录。在一个方面，当关于API函数调用的被支持的记录与行为规则中包含的关于API函数调用的所有记录的数量相比的百分比大于给定的数值(诸如75％)时，创建仅包括关于API函数调用的被支持的记录的防病毒记录。作为示例，让我们假设已经触发了以下行为规则：“来自可疑API函数列表中的以下API函数的调用：WinExec、CreateProcess、GetFileSize、CreateFile；在调用API函数WriteFile后调用API函数WinExec”。还假设保护器102支持API函数调用WinExec、CreateProcess、CreateFile、WriteFile的注册，但不支持关于API函数调用GetFileSize的记录的注册。对于该情况，被支持的API函数与触发的行为规则中包含的所有API函数的数量的百分比为80％，其大于75％(给定数值的示例)，所以，可以创建防病毒记录。

[0146] 图6为示出根据示例性方面的在其上可以实现本发明的各个方面的通用计算机系统20的框图。应当注意的是，计算机系统20可以对应于系统100和/或系统100的各个组件。

[0147] 如图所示，该计算机系统20(其可以是个人计算机或服务器)包括中央处理单元21、系统存储器22和连接各个系统部件的系统总线23，所述系统部件包括与中央处理单元21相关联的存储器。如本领域的普通技术人员将理解的，系统总线23可以包括总线存储器或总线存储器控制器、外围总线、以及能够与任何其它总线架构交互的本地总线。系统存储器可以包括永久存储器(ROM)24和随机存取存储器(random-access memory，RAM)25。基本输入/输出系统(basic input/output system，BIOS)26可以存储用于在计算机系统20的各元件之间传输信息的基本程序，诸如在使用ROM 24加载操作系统时的那些基本程序。

[0148] 计算机系统20还可以包括用于读取和写入数据的硬盘27、用于在可移动磁盘29上读取和写入的磁盘驱动器28、以及用于读取和写入可移动光盘31(诸如CD-ROM、DVD-ROM和其它光学介质)的光盘驱动器30。硬盘27、磁盘驱动器28和光盘驱动器30分别通过硬盘接口32、磁盘接口33和光盘驱动器接口34连接到系统总线23。驱动器和相应的计算机信息介质

是用于存储计算机系统20的计算机指令、数据结构、程序模块和其它数据的电源独立的模块。

[0149] 示例性方面包括使用硬盘27、可移动磁盘29和可移动光盘31通过控制器55连接到系统总线23的系统。本领域普通技术人员将会理解，也可以使用能够以计算机可读形式存储数据的任何类型的介质56(固态驱动器、闪存卡、数字盘、随机存取存储器(RAM)等)。

[0150] 计算机系统20具有可存储操作系统35的文件系统36、以及附加的程序应用37、其它程序模块38和程序数据39。计算机系统20的用户可以使用键盘40、鼠标42、或本领域普通技术人员已知的任何其它输入设备(诸如但不限于麦克风、操纵杆、游戏控制器、扫描仪等)来输入命令和信息。这种输入设备通常通过串行端口46插入计算机系统20，串行端口46又连接到系统总线，但是本领域的普通技术人员将理解，输入设备也可以以其它方式连接，例如但不限于通过并行端口、游戏端口或通用串行总线(universal serial bus，USB)连接。监控器47或其它类型的显示设备也可以通过诸如视频适配器48的接口连接到系统总线23。除了监控器47之外，个人计算机还可以配备有其它的外围输出设备(未示出)，例如扬声器、打印机等。

[0151] 计算机系统20可以使用与一个或多个远程计算机49的网络连接而在网络环境中操作。一个或多个远程计算机49可以为本地计算机工作站或服务器，其包括在描述计算机系统20的性质时描述的上述元件中的大多数元件或全部元件。计算机网络中还可以存在其它设备，诸如但不限于路由器、网站、对等设备或其它的网络节点。

[0152] 网络连接可以形成局域计算机网络(local-area computer network，LAN)50和广域计算机网络(wide-area computer network，WAN)。这种网络用在公司计算机网络和公司内部网络中，并且这些网络通常具有访问互联网的权限。在LAN或WAN网络中，个人计算机20通过网络适配器或网络接口51连接到局域网50。当使用网络时，计算机20系统可以使用调制解调器54或本领域普通技术人员熟知的、实现与广域计算机网络(诸如因特网)的通信的其它模块。可以是内部设备或外部设备的调制解调器54，可以通过串行端口46连接到系统总线23。本领域普通技术人员将理解，所述网络连接是使用通信模块建立一台计算机与另一台计算机的连接的许多熟知方式的非限制性示例。

[0153] 在各个方面中，本文中所描述的系统和方法可以以硬件、软件、固件或它们的任何组合来实现。如果以软件实现，则上述方法可以作为一个或多个指令或代码而被存储在非暂时性计算机可读介质上。计算机可读介质包括数据存储器。作为示例而非限，这种计算机可读介质可以包括RAM、ROM、EEPROM、CD-ROM、闪存或其它类型的电存储介质、磁存储介质或光存储介质、或可用来携带或存储所期望的指令或数据结构形式的程序代码并可以被通用计算机的处理器访问的任何其它介质。

[0154] 在各个方面中，本发明中所描述的系统和方法可以按照模块来描述。本文中所使用的术语“模块”指的是例如使用硬件(例如通过专用集成电路(application specific integrated circuit，ASIC)或现场可编程门阵列(field-programmable gate array，FPGA))实现的实际的设备、部件、或部件的布置，或者指的是硬件和软件的组合，例如通过微处理器系统和实现模块功能的指令集(该指令集在被执行时将微处理器系统转换成专用设备)来实现这样的组合。一个模块还可以被实施为两个模块的组合，其中仅通过硬件促进某些功能，并且通过硬件和软件的组合促进其它功能。在某些实现方式中，可以在通用计算

机的处理器上实现模块的至少一部分(以及在一些情况下,模块的全部)。因此,每个模块可以以各种适合的配置来实现,而不应受限于本文中所例示的任何特定的实现方式。

[0155] 为了清楚起见,本文中没有公开各个方面的所有常规特征。应当领会的是,在本发明的任何实际的实现方式的开发中,必须做出许多特定实现方式的决定,以便实现开发者的特定目标,并且这些特定目标将对于不同的实现方式和不同的开发者变化。应当理解的是,这种开发努力会是复杂的且费时的,但对于了解本发明的优点的本领域的普通技术人员来说仍然是工程的常规任务。

[0156] 此外,应当理解的是,本文中所使用的措辞或术语出于描述而非限制的目的,从而本说明书的术语或措辞应当由本领域技术人员根据本文中所提出的教导和指导结合相关领域技术人员的知识来解释。此外,不旨在将本说明书或权利要求中的任何术语归于不常见的或特定的含义,除非明确如此阐述。

[0157] 本文中所公开的各个方面包括本文中以说明性方式所引用的已知模块的现在和未来已知的等同物。此外,尽管已经示出并描述了各个方面和应用,但是对于了解本发明的优点的本领域技术人员将显而易见的是,在不脱离本文中所公开的发明构思的前提下,相比于上文所提及的内容而言的更多修改是可行的。

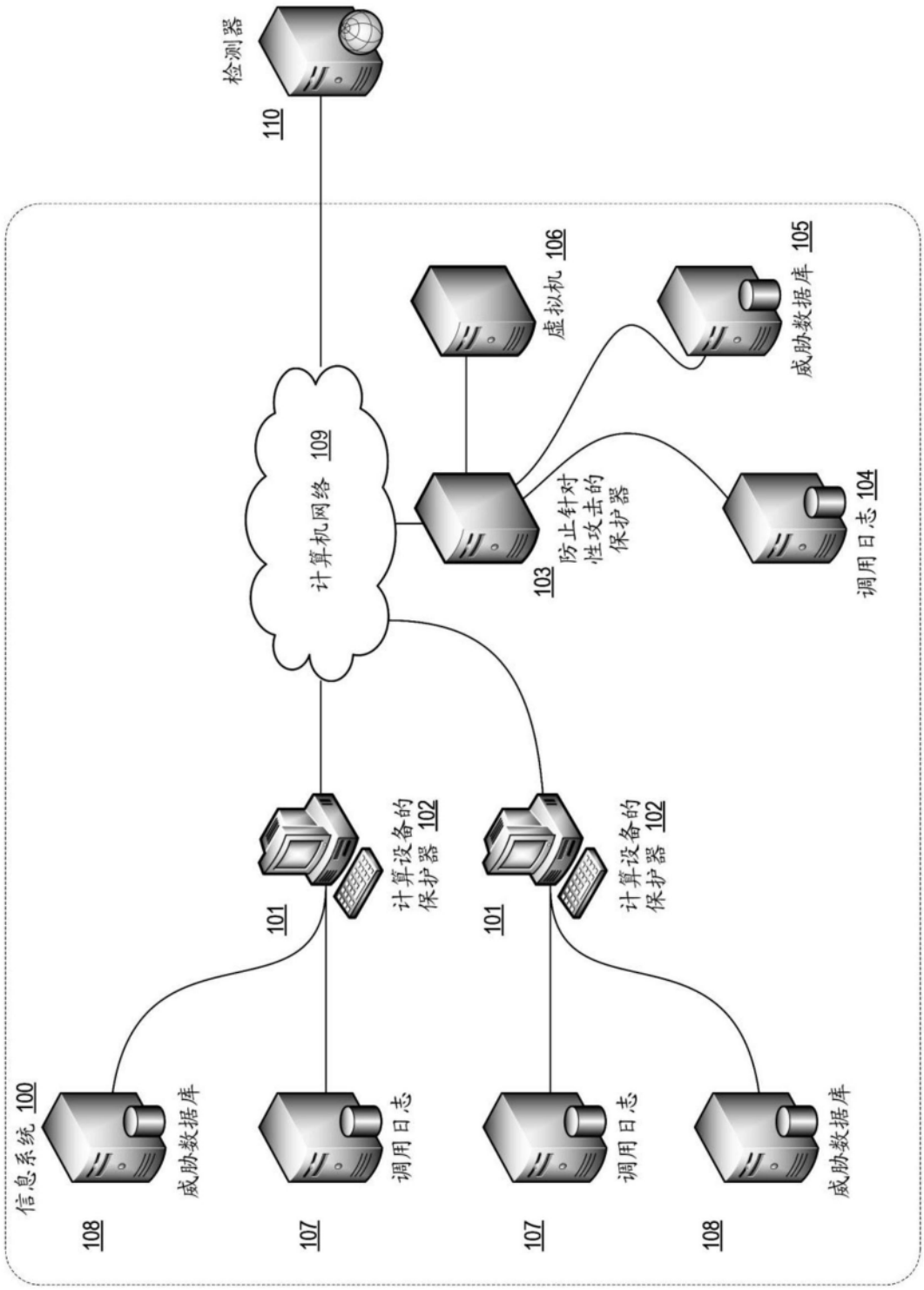

检测器
110
计算机网络 109
虚拟机 106
威胁数据库 105
防止针对性攻击的保护器
103
调用日志 104
计算设备的保护器 102
101
计算设备的保护器 102
101
信息系统 100
威胁数据库
108
调用日志
107
调用日志
107
威胁数据库
108

图1

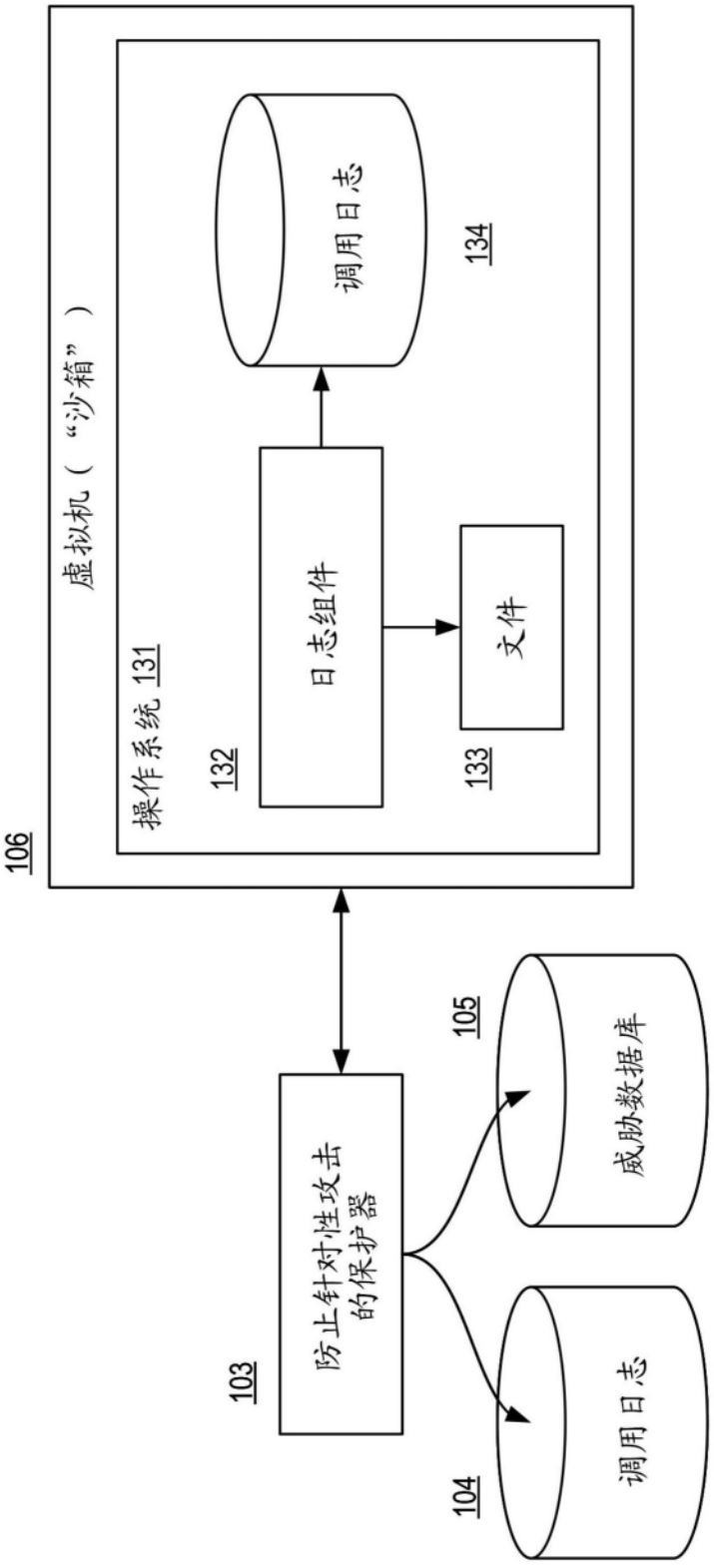

虚拟机（“沙箱”）
106
操作系统 131
132
日志组件
133
文件
调用日志
134
103
防止针对性攻击的保护器
105
威胁数据库
104
调用日志

图2

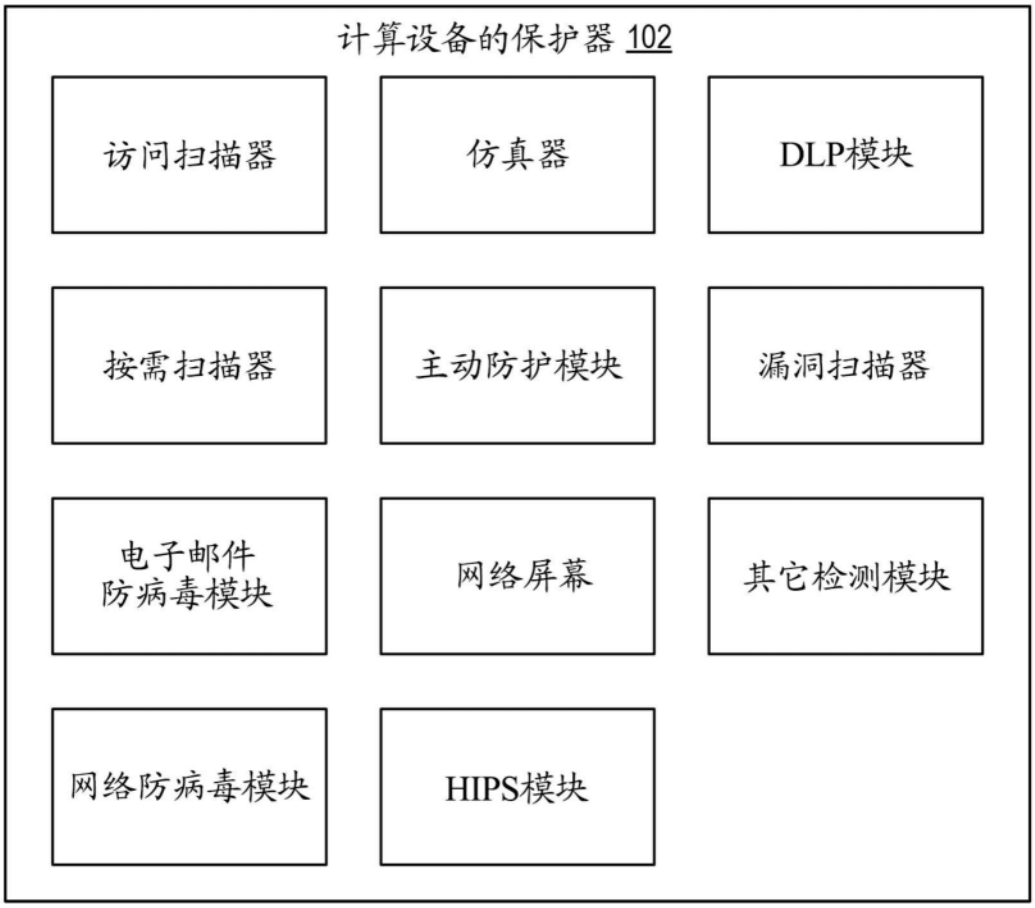

计算设备的保护器 102
访问扫描器
仿真器
DLP模块
按需扫描器
主动防护模块
漏洞扫描器
电子邮件
防病毒模块
网络屏幕
其它检测模块
网络防病毒模块
HIPS模块

图3

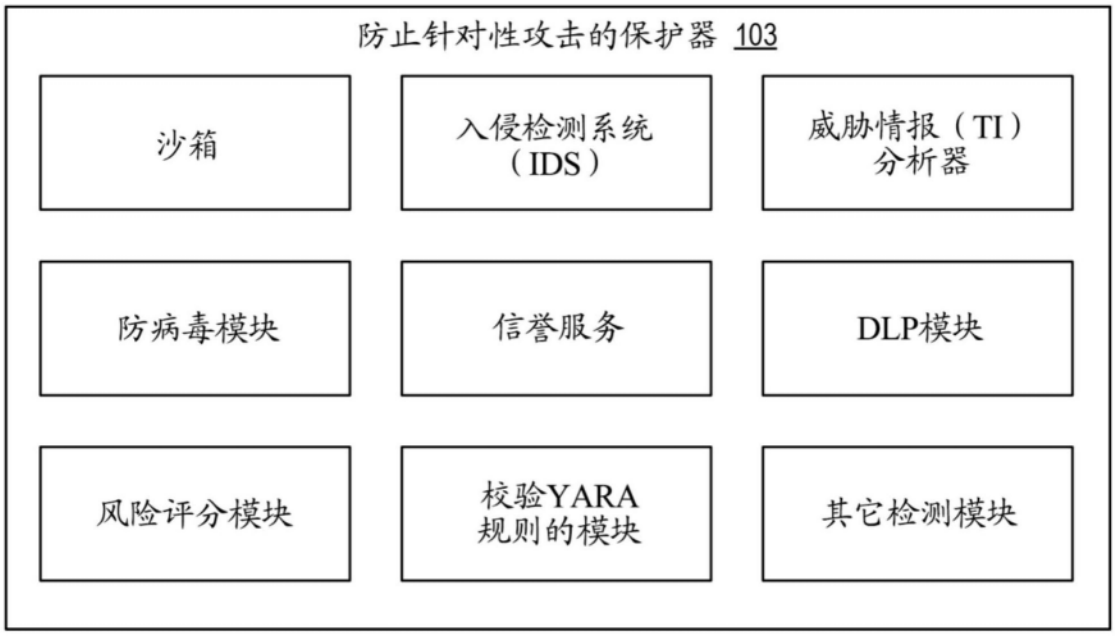

防止针对性攻击的保护器 103
沙箱
入侵检测系统
（IDS）
威胁情报（TI）
分析器
防病毒模块
信誉服务
DLP模块
风险评分模块
校验YARA
规则的模块
其它检测模块

图4

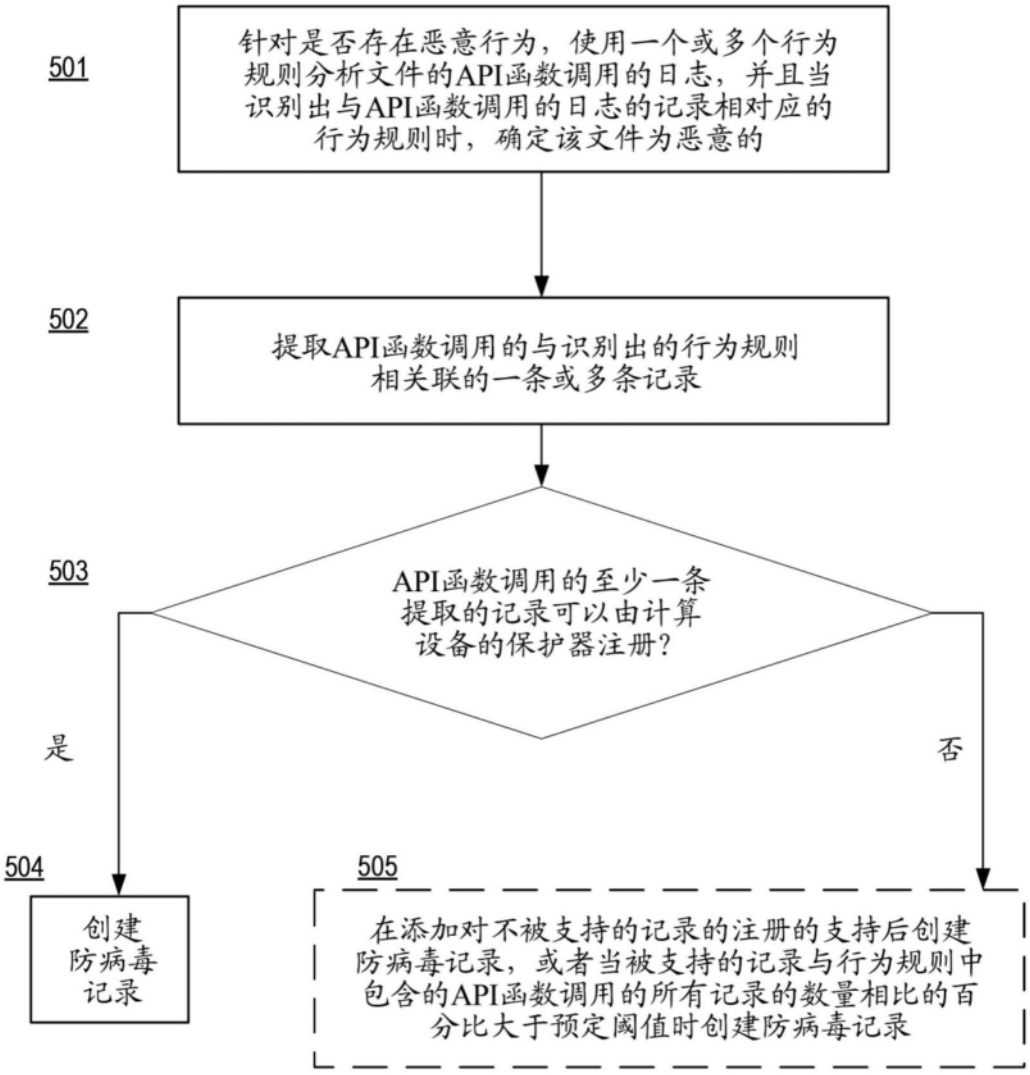

501
针对是否存在恶意行为，使用一个或多个行为规则分析文件的API函数调用的日志，并且当识别出与API函数调用的日志的记录相对应的行为规则时，确定该文件为恶意的
502
提取API函数调用的与识别出的行为规则相关联的一条或多条记录
503
API函数调用的至少一条提取的记录可以由计算设备的保护器注册?
是
否
504
创建防病毒记录
505
在添加对不被支持的记录的注册的支持后创建防病毒记录，或者当被支持的记录与行为规则中包含的API函数调用的所有记录的数量相比的百分比大于预定阈值时创建防病毒记录

图5

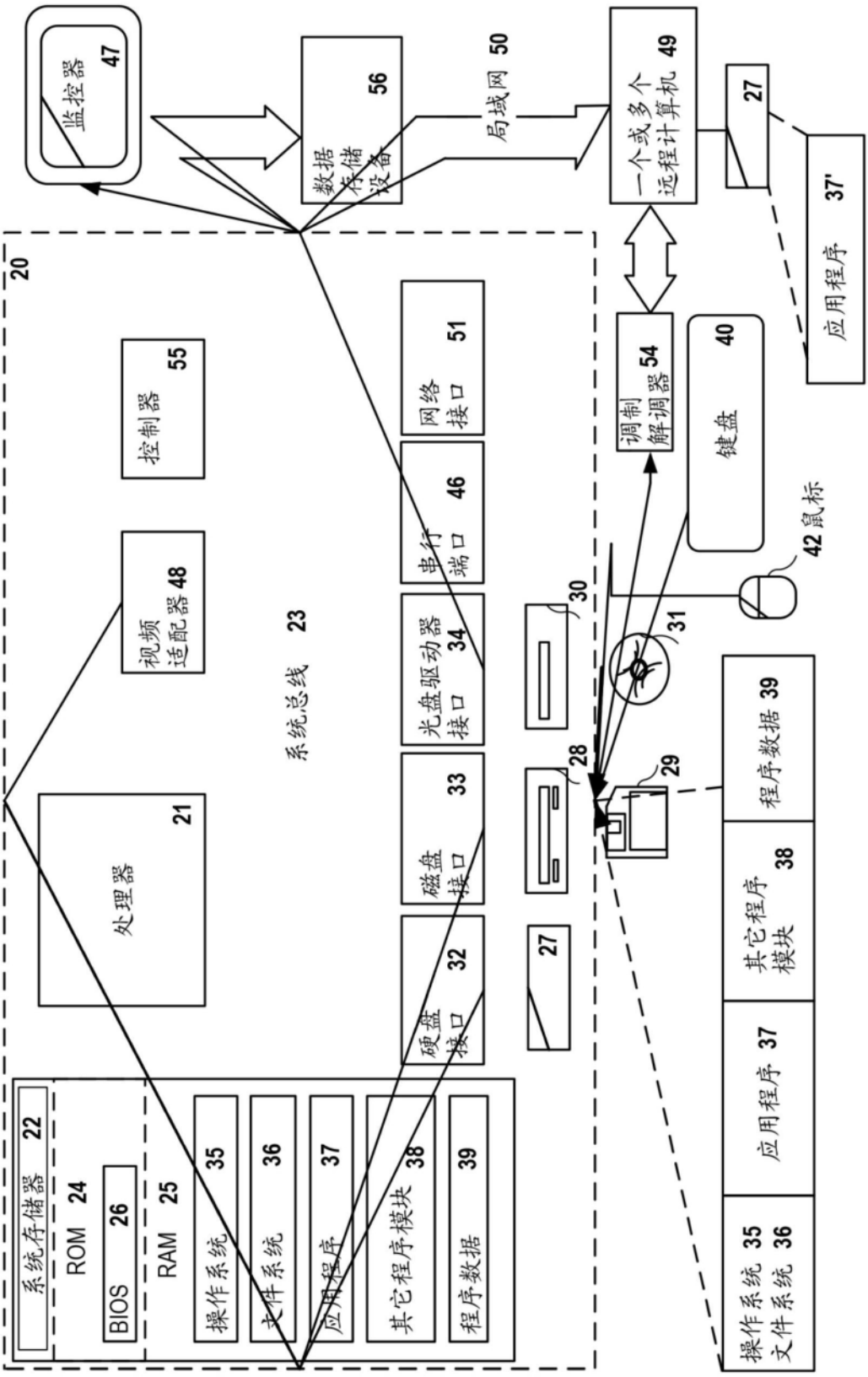

监控器 47
数据存储设备 56
局域网 50
一个或多个远程计算机 49
27
应用程序 37'
20
网络接口 51
控制器 55
调制解调器 54
键盘 40
串行端口 46
视频适配器 48
光盘驱动器接口 34
鼠标 42
30
31
系统总线 23
磁盘接口 33
28
29
处理器 21
程序数据 39
硬盘接口 32
27
其它程序模块 38
应用程序 37
操作系统 35
文件系统 36
系统存储器 22
ROM 24
BIOS 26
RAM 25
操作系统 35
应用程序 36
文件系统 37
其它程序模块 38
程序数据 39

图6