(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 110737887 A
(43)申请公布日 2020.01.31

(21)申请号 201911003904.9

(22)申请日 2019.10.22

(71)申请人 厦门美图之家科技有限公司
地址 361000 福建省厦门市火炬高新区软件园华讯楼C区B1F-089

(72)发明人 陈鸿图

(74)专利代理机构 北京超凡宏宇专利代理事务所(特殊普通合伙) 11463
代理人 张萌

(51)Int.Cl.
G06F 21/51(2013.01)
G06F 21/53(2013.01)
G06F 21/56(2013.01)

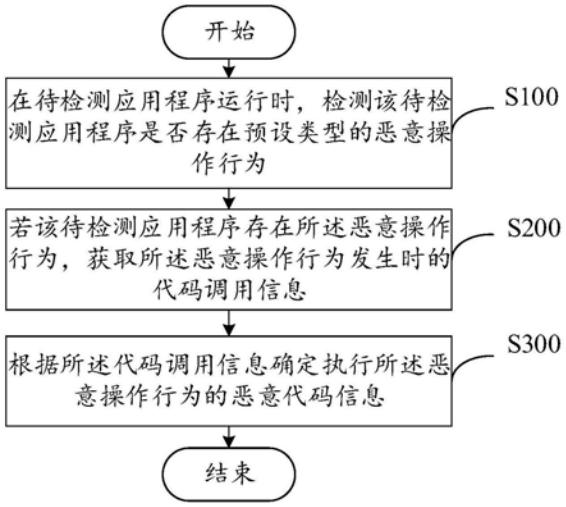
权利要求书2页 说明书7页 附图2页

(54)发明名称

恶意代码检测方法、装置、电子设备及存储介质

(57)摘要

本申请提供一种恶意代码检测方法、装置、电子设备及存储介质。该方法通过在待检测应用程序运行时，检测到该待检测应用程序发生预设类型的恶意操作行为时，获取该待检测应用程序的代码调用信息。根据该代码调用信息确定发生该恶意操作行为的恶意代码信息。如此，检测该恶意代码由应用程序运行时自动进行，提高了排查恶意代码的精度与效率。



CN 110737887 A

1.一种恶意代码检测方法,其特征在于,应用于电子设备,所述方法包括:

在待检测应用程序运行时,检测该待检测应用程序是否存在预设类型的恶意操作行为;

若该待检测应用程序存在所述恶意操作行为,获取所述恶意操作行为发生时的代码调用信息;

根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

2.根据权利要求1所述的恶意代码检测方法,其特征在于,所述方法还包括:

若该待检测应用程序存在所述恶意操作行为,截断所述恶意操作行。

3.根据权利要求1所述的恶意代码检测方法,其特征在于,所述电子设备还与服务器通信,所述方法还包括:

将执行所述恶意操作行为的恶意代码信息发送给所述服务器,以方便开发人员查看或所述服务器的后台程序告警处理。

4.根据权利要求1所述的恶意代码检测方法,其特征在于,所述待检测应用程序运行时,检测该待检测应用程序是否存在恶意操作行为的步骤包括:

所述待检测应用程序退出本程序显示界面时,检测所述待检测应用程序是否触发新的非法显示界面。

5.根据权利要求4所述的恶意代码检测方法,其特征在于,所述待检测应用程序为Android应用程序,所述检测所述待检测应用程序是否触发新的非法显示界面的步骤包括:

获取所述Android应用程序用于触发目标显示界面的配置信息;

将所述目标显示界面的配置信息与预设合法配置信息进行匹配;

若所述目标显示界面的配置信息与所述预设合法配置信息匹配失败,则所述目标显示界面为所述非法显示界面。

6.根据权利要求5所述的恶意代码检测方法,其特征在于,所述待检测应用程序退出本程序显示界面时的步骤包括:

获取所述Android应用程序当前打开的显示界面数量,所述显示界面为所述Android应用程序的Activity组件所显示的显示界面;

当所述显示界面数量为0,所述Android应用程序退出本程序显示界面。

7.根据权利要求5所述的恶意代码检测方法,其特征在于,所述获取所述恶意操作行为发生时的代码调用信息的步骤包括:

通过Java虚拟机获取所述恶意操作行为发生时的方法调用栈;

根据所述方法调用栈获得所述代码调用信息。

8.一种恶意代码检测装置,其特征在于,应用于电子设备,所述恶意代码检测装置包括操作行为检测模块、调用信息获取模块和恶意代码确定模块;

所述操作行为检测模块用于在待检测应用程序运行时,检测该待检测应用程序是否存在恶意操作行为;

所述调用信息获取模块用于若该待检测应用程序存在恶意操作行为,获取所述恶意操作行为发生时的代码调用信息;

所述恶意代码确定模块用于根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

9.一种电子设备，其特征在于，包括处理器和存储器，所述存储器存储有能够被所述处理器执行的机器可执行指令，所述处理器可执行所述机器可执行指令，以实现权利要求1-7任一项所述的恶意代码检测方法。

10.一种存储介质，其特征在于，其上存储有计算机程序，所述计算机程序被执行时，实现权利要求1-7中任意一项所述的恶意代码检测方法。

恶意代码检测方法、装置、电子设备及存储介质

技术领域

[0001] 本申请涉及计算机领域，具体而言，涉及一种恶意代码检测方法、装置、电子设备及存储介质。

背景技术

[0002] 在程序开发过程，往往会使用到第三方现成的SDK(Software Development Kit，软件开发工具包)以加快开发进度。然而第三方SDK的开发者可能会在代码中插入恶意代码，用以牟利。例如，利用插入的代码进行广告推广，以收取广告费。

[0003] 针对该问题，目前主要通过静态排查以及SDK详情比对的方式进行恶意代码的排查。其中，静态排查通过将SDK进行反编译，获得该SDK可读的源代码，开发人员对该源码进行排查。该SDK详情比对通过将没有恶意代码的老版本SDK与待排查的新版本SDK进行文件比对，找出该新版本SDK相较于老版本SDK发生修改的位置，对其进行重点排查。如此，以缩小排查范围。

[0004] 上述两种方式都是通过开发人员进行人力排查，在SDK体积较大时，费时费力且对开发人员的技术水平要求较高。排查的效率与精度严重依赖于开发人员。

发明内容

[0005] 本申请的目的在于提供一种恶意代码检测方法、装置、电子设备及存储介质，旨在准确查找应用程序中的恶意代码。

[0006] 本申请实施例的目的之一在于提供一种恶意代码检测方法，应用于电子设备，所述方法包括：

[0007] 在待检测应用程序运行时，检测该待检测应用程序是否存在恶意操作行为；

[0008] 若该待检测应用程序存在恶意操作行为，获取所述恶意操作行为发生时的代码调用信息；

[0009] 根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

[0010] 可选地，所述方法还包括：

[0011] 若该待检测应用程序存在所述恶意操作行为，截断所述恶意操作行。

[0012] 可选地，所述电子设备还与服务器通信，所述方法还包括：

[0013] 将执行所述恶意操作行为的恶意代码信息发送给所述服务器，以方便开发人员查看或所述服务器的后台程序告警处理。

[0014] 可选地，所述待检测应用程序运行时，检测该待检测应用程序是否存在恶意操作行为的步骤包括：

[0015] 所述待检测应用程序退出本程序显示界面时，检测所述待检测应用程序是否触发新的非法显示界面。

[0016] 可选地，所述待检测应用程序为Android应用程序，所述检测所述待检测应用程序是否触发新的非法显示界面的步骤包括：

[0017] 获取所述Android应用程序用于触发目标显示界面的配置信息；

[0018] 将所述目标显示界面的配置信息与预设合法配置信息进行匹配；

[0019] 若所述目标显示界面的配置信息与所述预设合法配置信息匹配失败，则所述目标显示界面为所述非法显示界面。

[0020] 可选地，所述待检测应用程序退出本程序显示界面时的步骤包括：

[0021] 获取所述Android应用程序当前打开的显示界面数量，所述显示界面为所述Android应用程序的Activity组件所显示的显示界面；

[0022] 当所述显示界面数量为0，所述Android应用程序退出本程序显示界面。

[0023] 可选地，所述获取所述恶意操作行为发生时的代码调用信息的步骤包括：

[0024] 通过Java虚拟机获取所述恶意操作行为发生时的方法调用栈；

[0025] 根据所述方法调用栈获得所述代码调用信息。

[0026] 本申请实施例的目的之二在于提供一种恶意代码检测装置，应用于电子设备，所述恶意代码检测装置包括操作行为检测模块、调用信息获取模块和恶意代码确定模块；

[0027] 所述操作行为检测模块用于在待检测应用程序运行时，检测该待检测应用程序是否存在恶意操作行为；

[0028] 所述调用信息获取模块用于若该待检测应用程序存在恶意操作行为，获取所述恶意操作行为发生时的代码调用信息；

[0029] 所述恶意代码确定模块用于根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

[0030] 本申请实施例的目的之三在于提供一种电子设备，包括处理器和存储器，所述存储器存储有能够被所述处理器执行的机器可执行指令，所述处理器可执行所述机器可执行指令，以实现所述恶意代码检测方法。

[0031] 本申请实施例的目的之四在于提供一种存储介质，其上存储有计算机程序，所述计算机程序被执行时，实现所述恶意代码检测方法。

[0032] 相对于现有技术而言，本申请具有以下有益效果：

[0033] 本申请实施例提供的一种恶意代码检测方法、装置、电子设备及存储介质。该方法通过在待检测应用程序运行时，检测到该待检测应用程序发生预设类型的恶意操作行为时，获取该待检测应用程序的代码调用信息。根据该代码调用信息确定发生该恶意操作行为的恶意代码信息。如此，检测该恶意代码由程序运行时自动进行，提高了排查恶意代码的精度与效率。

附图说明

[0034] 为了更清楚地说明本申请实施例的技术方案，下面将对实施例中所需要使用的附图作简单地介绍，应当理解，以下附图仅示出了本申请的某些实施例，因此不应被看作是对范围的限定，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他相关的附图。

[0035] 图1为本申请实施例提供的电子设备的硬件结构图；

[0036] 图2为本申请实施例提供的恶意代码检测方法的步骤流程图；

[0037] 图3为本申请实施例提供的手机桌面的示意图；

[0038] 图4为本申请实施例提供的恶意代码检测装置的结构示意图。

[0039] 图标:100-电子设备;110-恶意代码检测装置;120-存储器;130-处理器;1101-操作行为检测模块;1102-调用信息获取模块;1103-恶意代码确定模块。

具体实施方式

[0040] 为使本申请实施例的目的、技术方案和优点更加清楚,下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本申请一部分实施例,而不是全部的实施例。通常在此处附图中描述和示出的本申请实施例的组件可以以各种不同的配置来布置和设计。

[0041] 因此,以下对在附图中提供的本申请的实施例的详细描述并非旨在限制要求保护的本申请的范围,而是仅仅表示本申请的选定实施例。基于本申请中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本申请保护的范围。

[0042] 应注意到:相似的标号和字母在下面的附图中表示类似项,因此,一旦某一项在一个附图中被定义,则在随后的附图中不需要对其进行进一步定义和解释。

[0043] 如背景技术部分所介绍,开发应用程序时若使用了第三方的SDK,该SDK可能会被其开发人员插入恶意代码,用以谋取利益。因此,应用程序开发人员在使用第三方SDK时,需要对其进行排查,避免混入恶意代码。目前,主要通过开发人员在应用程序开发过程中,人工进行排查,效率低下且排查精度较低。

[0044] 基于此,本申请实施例提供一种恶意代码检测方法,应用于电子设备100。所述电子设备100可以是,但不限于,智能手机、个人电脑(Personal Computer,PC)、平板电脑、个人数字助理(Personal Digital Assistant,PDA)、移动上网设备(Mobile Internet Device,MID)等。

[0045] 请参照图1,该电子设备100包括恶意代码检测装置110、存储器120以及处理器130。所述存储器120、处理器130以及各元件相互之间直接或间接地电性连接,以实现数据的传输或交互。例如,这些元件相互之间可通过一条或多条通讯总线或信号线实现电性连接。所述恶意代码检测装置110包括至少一个可以软件或固件(firmware)的形式存储于所述存储器120中或固化在所述电子设备100的操作系统(Operating System,OS)中的软件功能模块。所述处理器130用于执行所述存储器120中存储的可执行模块,例如所述恶意代码检测装置110所包括的软件功能模块及计算机程序等。

[0046] 其中,所述存储器120可以是,但不限于,随机存取存储器(Random Access Memory,RAM),只读存储器(Read Only Memory,ROM),可编程只读存储器(Programmable Read-Only Memory,PROM),可擦除只读存储器(Erasable Programmable Read-Only Memory,EPROM),电可擦除只读存储器(Electric Erasable Programmable Read-Only Memory,EEPROM)等。其中,存储器120用于存储程序,所述处理器130在接收到执行指令后,执行所述程序。

[0047] 所述处理器130可能是一种集成电路芯片,具有信号的处理能力。上述的处理器可以是通用处理器,包括中央处理器(Central Processing Unit,简称CPU)、网络处理器(Network Processor,简称NP)等;还可以是数字信号处理器(DSP)、专用集成电路(ASIC)、

现场可编程门阵列(FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。可以实现或者执行本申请实施例中的公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

[0048] 请参照图2,图2为本申请实施例所提供的恶意代码检测方法,以下将对所述方法包括各个步骤进行详细阐述。

[0049] 步骤S100,在待检测应用程序运行时,检测该待检测应用程序是否存在恶意操作行为。

[0050] 应理解,该待检测应用程序为发布给用户使用的正式版本的应用程序,通过用户在使用该待检测应用程序的过程中,检测是否发生恶意操作行为。还应理解到,部分SDK其本身并不存在恶意代码,用户在使用集成了该SDK的应用程序时,SDK会自动从网络上在线下载恶意代码,并进行运行。若开发人员在开发过程进行人工排查,并不能发现并解决该类型的恶意代码。

[0051] 步骤S200,若该待检测应用程序存在恶意操作行为,获取所述恶意操作行为发生时的代码调用信息。

[0052] 步骤S300,根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

[0053] 在该待检测应用程序运行期间,该电子设备100一旦检测到恶意操作行为,则获取发生恶意操作行为时的代码调用信息。应理解,若该恶意代码来自于SDK,则发生恶意行为时,该电子设备100会调用SDK中的相关恶意代码。因此,该电子设备100获取该待检测应用程序在发生恶意行为时的程序运行的上下文信息,进而确定发生该恶意操作行为的恶意代码信息。

[0054] 如此,通过对待检测应用程序运行时的恶意操作行为进行检测,使得开发人员不用在开发时进行人工排查恶意代码,提高了排查效率。且检测精度极高,避免因为开发人员个人技术水平的原因,将一些正常代码误判为恶意代码。

[0055] 可选地,为了提高用户体验,该电子设备100检测到恶意操作行为时,截断该恶意操作行为进行,避免对用户体验造成影响。例如,该电子设备100一旦检测到将要打开某个非法广告界面,则对其进行截断,阻止该非法广告界面的弹出。

[0056] 可选地,该电子设备100还与服务器通信连接,在检测到发生该恶意操作行为的恶意代码信息时,将该恶意代码信息发送给该服务器,继而通知开发人员进行处理。例如,开发人员通过该服务器获知该恶意代码信息,并根据恶意代码信息对SDK进行相应的修改。其中,该恶意代码信息包括该恶意代码的名称以及在SDK中的位置。

[0057] 同时,针对自动从网络上在线下载恶意代码的SDK,在发生恶意操作行为时,同样会将恶意代码信息发送给服务器,使得开发人员也能够进行相应的排查。

[0058] 可选地,该应用待检测应用程序为Android应用程序,请参照图3,图3为该电子设备100所提供的Android操作系统的桌面环境。发明人研究发现,部分恶意代码的恶意操作行为极其隐蔽。用户在使用完Android应用程序返回桌面时,会触发新的非法显示界面,该非法显示界面可以是该Android应用程序所提供的与其他Android应用程序非常相似的界面,也可以是被该Android应用程序所触发的其他Android应用程序所提供的显示界面。使得用户误以为是该非法显示界面是由于自己误触发相应Android应用程序后所显示的界面。

[0059] 应理解，该非法显示界面的显示过程，并不是用户主动触发，完全是在用户不知情的情况弹出该非法显示界面。

[0060] 例如，用户在使用完Android应用程序返回桌面时会弹出一些购物网站的界面，使得用户误以为是误触发了购物类Android应用程序。

[0061] 基于此，该电子设备100需要先判断该Android应用程序是否退出本程序显示界面。该电子设备100通过获取该Android应用程序当前打开的显示界面的数量，当该显示界面的数量为0时，表明该Android应用程序退出本程序显示界面。其中，该电子设备100可以响应用户点击Android系统所提供的返回键、Home键或者切换到其他Android应用程序等操作，使得当前Android应用程序退出本程序显示界面。上述方式均会将当前Android应用程的显示界面的数量置0。

[0062] 应理解，该显示界面为所述Android应用程序的Activity组件所显示的显示界面。该Activity组件为Android系统的四大组件之一，包括Activity组件、Service组件、BroadcastReceiver组件以及ContentProvider组件。其中，Activity组件用于负责与用户交互时显示操作界面，当Android应用程序的Activity组件所显示的界面为0，表示该Android应用程序退出本程序显示界面，不再占据该电子设备100的屏幕。

[0063] 基于上述原理，该电子设备100检测在该Android应用程序退出本程序显示界面时，是否触发新的目标显示界面。

[0064] 该电子设备100获取该Android应用程序用于触发目标显示界面的配置信息，将所述目标显示界面的配置信息与预设合法配置信息进行匹配，若所述目标显示界面的配置信息与所述预设合法配置信息匹配失败，则所述目标显示界面为非法显示界面。

[0065] 应理解，该Android系统通过Intent机制实现Android系统各组件以及Android程序之间的交互。因此，该Android应用程序若触发目标显示界面，则需要通过Intent机制，并传递该目标显示界面的配置信息，使得Android系统根据该配置信息选取相应的组件或者其他Android程序进行显示。

[0066] 其中，值得说明的是，该目标显示界面可以是该Android应用程序自身的Activity组件所显示的界面，也可以是其他应用Android程序所显示的界面。

[0067] 基于此原理，该电子设备100获取该目标显示界面的配置信息，并与预设合法配置信息进行匹配。其中，该配置信息包括其他Android应用程序的包名以及scheme协议，所述scheme协议用于使得其他Android应用程序根据该scheme协议中的数据，执行相应的显示动作。

[0068] 例如，所述其他Android应用程序的包名为购物类Android应用程序的包名，所述scheme协议的数据为某商品的链接。该购物类Android应用程序会根据所述scheme协议中的链接，打开相应的购物界面。

[0069] 若该电子设备100将该目标显示界面的配置信息与预设合法配置信息匹配失败，则该目标显示界面为非法显示界面，属于恶意代码所执行的恶意操作行为。该电子设备100通过Java虚拟机获取发生恶意操作行为时的方法调用栈，继而获得该恶意代码的恶意代码信息。

[0070] 具体的，在一种可能的示例中，可以通过重写Android系统中的Application类中的startActivity()以及startActivities()等方法实现该恶意代码检测方法。

[0071] 可选地，该电子设备100还与服务器通信。若该电子设备100发现执行恶意操作行为的代码信息，将执行所述恶意操作行为的恶意代码信息发送给所述服务器，以方便开发人员查看或所述服务器的后台程序告警处理。

[0072] 如此，使得开发人员及时了解应用程序的当前状况，及时针对恶意代码做相应的处理，以提高用户的使用体验。

[0073] 请参照图4，本申请实施例还提供一种恶意代码检测装置110，应用于电子设备100。从功能上划分，所述恶意代码检测装置110包括操作行为检测模块1101、调用信息获取模块1102和恶意代码确定模块1103。

[0074] 该操作行为检测模块1101用于在待检测应用程序运行时，检测该待检测应用程序是否存在恶意操作行为。

[0075] 在本实施例中，该操作行为检测模块1101用于执行图2中的步骤S100，关于该操作行为检测模块1101的详细描述可以参考步骤S100的详细描述。

[0076] 该调用信息获取模块1102用于若该待检测应用程序存在恶意操作行为，获取所述恶意操作行为发生时的代码调用信息。

[0077] 在本实施例中，该调用信息获取模块1102用于执行图2中的步骤S200，关于该调用信息获取模块1102的详细描述可以参考步骤S200的详细描述。

[0078] 该恶意代码确定模块1103用于根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息。

[0079] 在本实施例中，该恶意代码确定模块1103用于执行图2中的步骤S300，关于该恶意代码确定模块1103的详细描述可以参考步骤S300的详细描述。

[0080] 本申请实施例还提供一种电子设备100，包括处理器130和存储器120，所述存储器120存储有能够被所述处理器130执行的机器可执行指令，所述处理器130可执行所述机器可执行指令，以实现所述恶意代码检测方法。

[0081] 本申请实施例还提供一种存储介质，其上存储有计算机程序，所述计算机程序被执行时，实现所述恶意代码检测方法。

[0082] 综上所述，本申请实施例提供的一种恶意代码检测方法、装置、电子设备及存储介质。该方法通过在待检测应用程序运行时，检测到该待检测应用程序发生预设类型的恶意操作行为时，获取该待检测应用程序的代码调用信息。根据该代码调用信息确定发生该恶意操作行为的恶意代码信息。如此，检测该恶意代码由程序运行时自动进行，提高了排查恶意代码的精度与效率。

[0083] 在本申请所提供的实施例中，应该理解到，所揭露的装置和方法，也可以通过其它的方式实现。以上所描述的装置实施例仅仅是示意性的，例如，附图中的流程图和框图显示了根据本申请的多个实施例的装置、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上，流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分，所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意，在有些作为替换的实现方式中，方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如，两个连续的方框实际上可以基本并行地执行，它们有时也可以按相反的顺序执行，这依所涉及的功能而定。也要注意的是，框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合，可以用执行规定的功能或动作的专用的基于

硬件的系统来实现，或者可以用专用硬件与计算机指令的组合来实现。

[0084] 另外，在本申请各个实施例中的各功能模块可以集成在一起形成一个独立的部分，也可以是各个模块单独存在，也可以两个或两个以上模块集成形成一个独立的部分。

[0085] 所述功能如果以软件功能模块的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读取存储介质中。基于这样的理解，本申请的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算机设备(可以是个人计算机，服务器，或者网络设备等)执行本申请各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM,Read-Only Memory)、随机存取存储器(RAM,Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0086] 需要说明的是，在本文中，诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来，而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

[0087] 以上所述，仅为本申请的各种实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应所述以权利要求的保护范围为准。

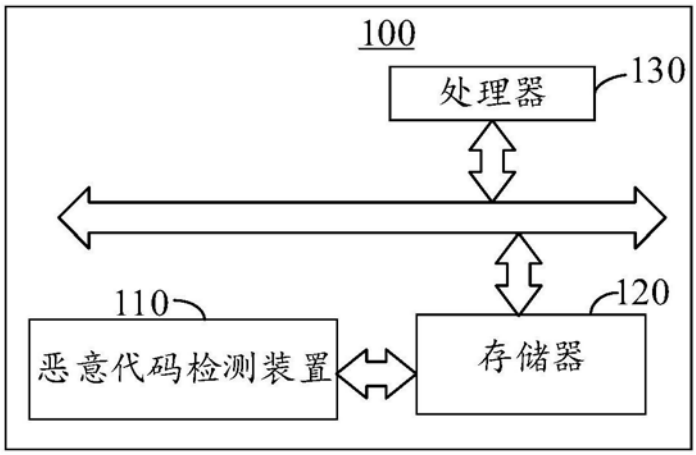

100
处理器
130
110
恶意代码检测装置
120
存储器

图1

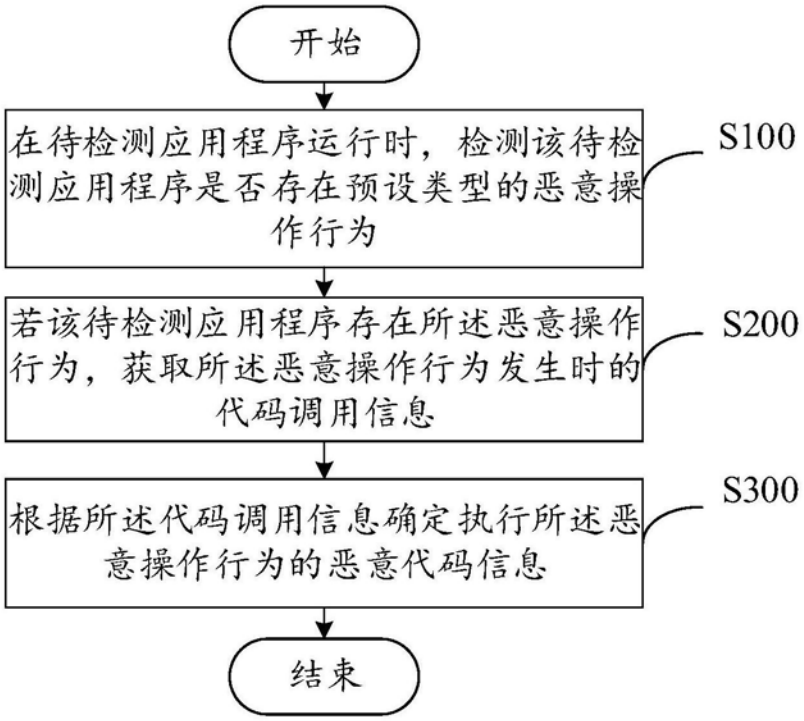

开始
在待检测应用程序运行时，检测该待检测应用程序是否存在预设类型的恶意操作行为
S100
若该待检测应用程序存在所述恶意操作行为，获取所述恶意操作行为发生时的代码调用信息
S200
根据所述代码调用信息确定执行所述恶意操作行为的恶意代码信息
S300
结束

图2

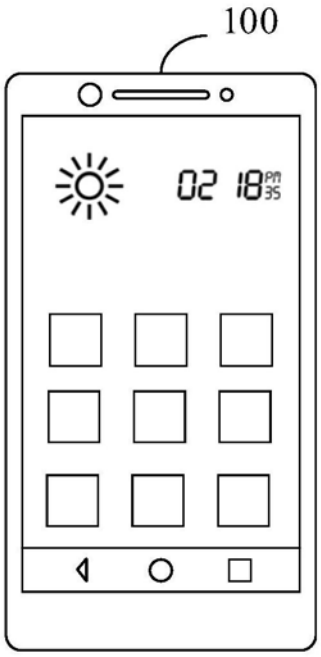


图3

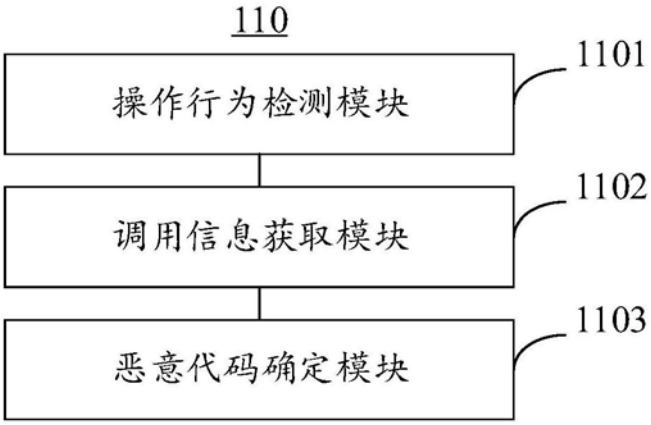


图4