(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 104239786 A
(43)申请公布日 2014.12.24

(21)申请号 201410539412.2

(22)申请日 2014.10.13

(71)申请人 北京奇虎科技有限公司
地址 100088 北京市西城区新街口外大街28号D座112室(德胜园区)
申请人 奇智软件(北京)有限公司

(72)发明人 李常坤 刘星 石浩然 杨威 孙年忠 王玺 张海

(74)专利代理机构 北京市立方律师事务所 11330
代理人 王增鑫

(51)Int.Cl.
G06F 21/55(2013.01)

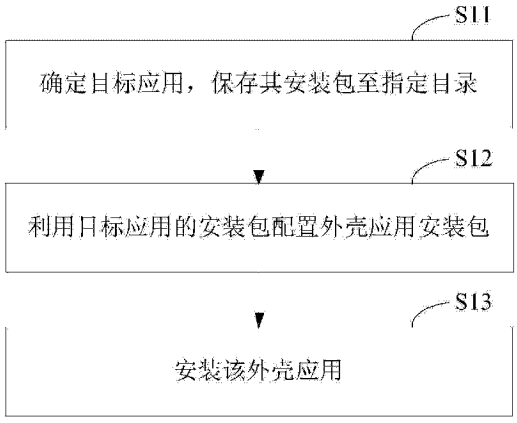
权利要求书2页 说明书23页 附图8页

(54)发明名称

免ROOT主动防御配置方法及装置

(57)摘要

本发明提供了一种免ROOT主动防御配置方法和相应的装置，该方法包括以下步骤：确定目标应用，保存其安装包至指定目录；利用目标应用的安装包配置外壳应用安装包，向其中注入用于调用监控单元的桩模块，修改其中的配置参数以用于加载所述目标应用，所述监控单元用于实现对源自所述目标应用的事件行为的挂钩监控；安装该外壳应用。本发明提出的主动防御方案，对现有系统的改动很小，不会影响系统的兼容性，而且实现简单、高效。



CN 104239786 A

1. 一种免 ROOT 主动防御配置方法，其特征在于，该方法包括以下步骤：

确定目标应用，保存其安装包至指定目录；

利用目标应用的安装包配置外壳应用安装包，向其中注入用于调用监控单元的桩模块，修改其中的配置参数以用于加载所述目标应用，所述监控单元用于实现对源自所述目标应用的事件行为的挂钩监控；

安装该外壳应用。

2. 根据权利要求 1 所述的免 ROOT 主动防御配置方法，其特征在于，所述监控单元从远程插件接口获得对应于特定事件行为的挂钩插件，所述挂钩插件用于挂钩监控特定事件行为。

3. 根据权利要求 1 所述的免 ROOT 主动防御配置方法，其特征在于，利用目标应用配置外壳应用的过程包括如下具体步骤：

解析目标应用安装包，生成外壳应用的镜像；

修改或替换镜像中的代码文件，以注入所述桩模块；

修改镜像中的配置文件的配置参数，用于加载指定目录中的目标应用；

对外壳应用镜像进行打包签名，完成外壳应用的封装。

4. 根据权利要求 1 所述的免 ROOT 主动防御配置方法，其特征在于，所述监控单元所监控的事件行为包括以下任意一种或多种行为类型：获取运营商信息、APN 操作、通知栏广告操作、获取手机识别码操作、创建快捷方式、电话拨打操作、短信操作、联系人操作、URL 访问操作、子进程侵入操作、应用加载操作、命令操作、衍生物操作、激活设备管理器操作。

5. 根据权利要求 1 至 4 中任意一项所述的免 ROOT 主动防御配置方法，其特征在于，将目标应用配置为外壳应用的过程中，还为外壳应用配置交互接口，通过该交互接口向系统服务发送捕获的事件行为。

6. 一种免 ROOT 主动防御配置装置，其特征在于，包括：

确定装置，用于确定目标应用，保存其安装包至指定目录；

构造装置，其利用目标应用的安装包配置外壳应用安装包，向其中注入用于调用监控单元的桩模块，修改其中的配置参数以用于加载所述目标应用，所述监控单元用于实现对源自所述目标应用的事件行为的挂钩监控；

安装装置，用于安装该外壳应用。

7. 根据权利要求 6 所述的免 ROOT 主动防御配置方法，其特征在于，所述监控单元，用于从远程插件接口获得对应于特定事件行为的挂钩插件，所述挂钩插件用于挂钩监控特定事件行为。

8. 根据权利要求 6 所述的免 ROOT 主动防御配置方法，其特征在于，所述构造装置包括：

解析单元，用于解析目标应用安装包，生成外壳应用镜像；

代码单元，用于修改或替换镜像中的代码文件，以注入所述桩模块；

配置单元，用于修改镜像中的配置文件的配置参数，用于加载指定目录中的目标应用；

封装单元，用于对外壳应用镜像进行打包签名，完成外壳应用的封装。

9. 根据权利要求 6 所述的免 ROOT 主动防御配置方法，其特征在于，所述监控单元所监

控的事件行为包括以下任意一种或多种行为类型：获取运营商信息、APN 操作、通知栏广告操作、获取手机识别码操作、创建快捷方式、电话拨打操作、短信操作、联系人操作、URL 访问操作、子进程侵入操作、应用加载操作、命令操作、衍生物操作、激活设备管理器操作。

10. 根据权利要求 6 至 9 中任意一项所述的免 ROOT 主动防御配置方法，其特征在于，所述外壳应用配置有交互接口，通过该交互接口向系统服务发送捕获的事件行为。

免 ROOT 主动防御配置方法及装置

技术领域

[0001] 本发明涉及计算机安全领域，具体而言，本发明涉及一种免 ROOT 主动防御配置方法，相应还涉及一种免 ROOT 主动防御配置装置。

背景技术

[0002] Unix 系的操作系统，以 Android 为典型代表，广泛应用于各种移动通信终端中。Android 具有相对较为严格的用户权限管理机制，默认状态下，用户的权限较低。要突破权限限制，需要将系统的权限提高到最高级别，也即进行 ROOT 授权。获得最高权限后，用户便可对第三方应用的恶意行为进行拦截，对消耗系统资源的设置项进行修改，因此，多数情况下，市面上的安全软件需要在已经获得 ROOT 授权的 Android 移动终端上工作，才能达到其最优效果。但是，一般用户并不掌握较高的专业知识，未必能对其终端进行 ROOT 授权，即使进行了 ROOT 授权，在为安全软件开放更高权限的同时，也给了恶意程序以可乘之机。更为矛盾的是，在非 ROOT 条件下，一部分恶意程序能工作，而传统的安全防御软件却会失去绝对优势。因此，在非 ROOT 条件下解决 Android、Ubuntu 等类似系统的安全防御需求，是业内一直以来的努力方向。

[0003] 主动防御技术是满足上述需求的较佳解决方案。主动防御是基于程序事件行为自主分析判断的实时防护技术，不以病毒的特征码作为判断病毒的依据，而是从最原始的病毒定义出发，直接将程序的行为作为判断病毒的依据。主动防御是用软件自动实现了反病毒工程师分析判断病毒的过程，解决了传统安全软件无法防御未知恶意软件的弊端，从技术上实现了对木马和病毒的主动防御。

[0004] 请参阅 2014 年 9 月 3 日公开、公开号为 CN104023122A 的专利申请，其请求保护一种安全防御方法及装置。该方案的基本思路是通过下载预先定制的待注入应用程序来替换当前终端的相应的应用程序，并且在系统重启后优先启动该待注入应用程序，从而实现主动防御。该思路主要是为了解决如何构建安全防御机制的问题，而其中所涉的待注入应用程序是由当前终端的应用程序进行反编译、修改代码和重新封装后生成的，也就是采用了二次打包技术。本领域技术人员可以理解，这种依赖于对应用程序进行全面的二次打包实现的行为监控方式存在不足，表现在如下几个方面：

[0005] 首先，安装失败率高。事实上，越来越多的应用程序已经具备防止二次打包的免疫力，如果应用程序已经做好了防止二次打包的免疫设置，那么，强行向目标应用程序注入监控代码，会导致该目标应用不能安装，或者安装后出现异常崩溃，建构主动防御环境的成功率较低。

[0006] 其次，存在监控不全面的先天不足。挂钩函数构成应用程序的一部分，恶意程序可以利用 JAVA 反射机制中的反射调用、JNI 本地调用 (Native) 等技术来逃避这一防御机制。

[0007] 此外，监控精细程度不高。二次打包后的应用程序，其监控对象往往局限于应用程序本身，难以具体到精细行为，难以对诸如短信操作、联系人接入或删除操作、URL 访问操作、衍生物操作、安装操作、子进程侵入等具体行为做出精细的监控。

[0008] 综合以上的分析可知，业内关于主动防御技术的研究，仍有较大的提升空间。

发明内容

[0009] 本发明的首要目的在于，在免 ROOT 条件下建构更为高效的主动安全防御机制，从而提供一种免 ROOT 主动防御配置方法。

[0010] 本发明的另一目的在于配合首要目的而提供一种免 ROOT 主动防御配置装置。

[0011] 为实现本发明的上述目的，本发明提供如下技术方案：

[0012] 本发明提供的一种免 ROOT 主动防御配置方法，包括以下步骤：

[0013] 确定目标应用，保存其安装包至指定目录；

[0014] 利用目标应用的安装包配置外壳应用安装包，向其中注入用于调用监控单元的桩模块，修改其中的配置参数以用于加载所述目标应用，所述监控单元用于实现对源自所述目标应用的事件行为的挂钩监控；

[0015] 安装该外壳应用。

[0016] 本发明揭示的一种实施例中，所述目标应用的确定，通过用户界面的已安装目标应用列表的指示区域被动变化到选定状态而确定；确定目标应用并将该目标应用安装包复制到所述指定目录后，卸载该目标应用。

[0017] 本发明的另一实话例中，所述目标应用的确定，以接收安装广播的方式获取新装应用作为所述目标应用，从远程规则库接口获取关于该目标应用的处理规则，根据该处理规则向用户界面弹窗以获取对该目标应用的确定；将该目标应用安装包复制到所述指定目录之前或之后，停止该目标应用的安装。

[0018] 具体的，所述监控单元从远程插件接口获得对应于特定事件行为的挂钩插件，所述挂钩插件用于挂钩监控特定事件行为。

[0019] 具体的，利用目标应用配置外壳应用的过程包括如下具体步骤：

[0020] 解析目标应用安装包，生成外壳应用镜像；

[0021] 修改或替换镜像中的代码文件，以注入所述桩模块；

[0022] 修改镜像中的配置文件的配置参数，用于加载指定目录中的目标应用；

[0023] 对外壳应用镜像进行打包签名，完成外壳应用的封装。

[0024] 进一步，封装外壳应用的步骤中，采用手机识别码或随机码的方式对外壳应用进行签名。

[0025] 较佳的，所述外壳应用中，监控单元先于所述指定目录中的目标应用被加载。

[0026] 较佳的，所述外壳应用安装包的文件名与目标应用安装包保持一致，而外壳应用安装包所配置的图标至少之一与目标应用不同。

[0027] 具体的，所述监控单元所监控的事件行为包括以下任意一种或多种行为类型：获取运营商信息、APN 操作、通知栏广告操作、获取手机识别码操作、创建快捷方式、电话拨打操作、短信操作、联系人操作、URL 访问操作、子进程侵入操作、应用加载操作、命令操作、衍生物操作、激活设备管理器操作。

[0028] 进一步，将目标应用配置为外壳应用的过程中，还为外壳应用配置有交互模块，该交互模块被注册为系统服务，用于针对监控单元监控到的事件行为向用户界面弹窗以获取对应于事件行为的处理策略。

[0029] 本发明提供的一种免 ROOT 主动防御配置装置，包括：

[0030] 确定装置，用于确定目标应用，保存其安装包至指定目录；

[0031] 构造装置，其利用目标应用的安装包配置外壳应用安装包，向其中注入用于调用监控单元的桩模块，修改其中的配置参数以用于加载所述目标应用，所述监控单元用于实现对源自所述目标应用的事件行为的挂钩监控；

[0032] 安装装置，用于安装该外壳应用。

[0033] 本发明所揭示的一种实施例中，所述确定装置包括：

[0034] 选定单元，其通过用户界面的已安装目标应用列表的指示区域被动变化到选定状态而确定；

[0035] 处理单元，其用于在目标应用确定并将该目标应用安装包复制到所述指定目录后，卸载该目标应用。

[0036] 本发明揭示的另一实施例中，所述确定装置包括：

[0037] 选定单元，其以接收安装广播的方式获取新装应用作为所述目标应用，从远程规则库接口获取关于该目标应用的处理规则，根据该处理规则向用户界面弹窗以获取对该目标应用的确定；

[0038] 处理单元，其在将该目标应用安装包复制到所述指定目录之前或之后，停止该目标应用的安装。

[0039] 具体的，所述监控单元，用于从远程插件接口获得对应于特定事件行为的挂钩插件，所述挂钩插件用于挂钩监控特定事件行为。

[0040] 进一步，所述构造装置包括：

[0041] 解析单元，用于解析目标应用安装包，生成外壳应用镜像；

[0042] 代码单元，用于修改或替换镜像中的代码文件，以注入所述桩模块；

[0043] 配置单元，用于修改镜像中的配置文件的配置参数，用于加载指定目录中的目标应用；

[0044] 封装单元，用于对外壳应用镜像进行打包签名，完成外壳应用的封装。

[0045] 其中，所述封装单元，其采用手机识别码或随机码的方式对外壳应用进行签名。

[0046] 较佳的，所述外壳应用中，监控单元先于所述指定目录中的目标应用被加载。

[0047] 较佳的，所述外壳应用安装包的文件名与目标应用安装包保持一致，而外壳应用安装包所配置的图标至少之一与目标应用不同。

[0048] 具体的，所述监控单元所监控的事件行为包括以下任意一种或多种行为类型：获取运营商信息、APN 操作、通知栏广告操作、获取手机识别码操作、创建快捷方式、电话拨打操作、短信操作、联系人操作、URL 访问操作、子进程侵入操作、应用加载操作、命令操作、衍生物操作、激活设备管理器操作。

[0049] 进一步，所述外壳应用配置有交互模块，该交互模块被注册为系统服务，用于针对监控单元监控到的事件行为向用户界面弹窗以获取对应于事件行为的处理策略。

[0050] 相较于现有技术，本发明至少具有如下优点：

[0051] 1、真正实现了动态主动防御。本发明以目标应用为基本单位提出建构其主动防御环境的解决方案，可以通过在实时监测目标应用被安装后，或者通过识别用户对需要建立主动防御机制的目标程序的选定后，根据目标应用构造一个伪装成目标应用的外壳应用，

再由该外壳应用去加载监控单元和真正的目标应用，为目标应用程序及时动态建立防御机制，后续可借助这一外壳应用的运行实现主动防御。这一过程不需要对系统进行 ROOT 授权，不依赖于联网条件，更不依赖于以特征码为基础的病毒库，因此而真正实现了对目标应用程序的主动防御。

[0052] 2、所建立的主动防御机制安全有效。如前所述，本发明构造所述外壳应用时，是根据目标应用的安装包进行构造的，而目标应用的安装包本身被安全保存。由此，本发明一方面由于未改变待运行的目标应用的代码和配置，因而目标应用能够满足自校验要求，而外壳应用被视为所述的目标应用而合法存在；另一方面，即使带有恶意的目标应用企图利用 JAVA 反射机制避开检测，也难以逃脱监控单元的观察；再一方面还可以通过监控单元实现对真正的目标程序的事件行为的监控，以类似观察者的身份全面监视目标应用的一切事件行为，对各种特定事件行为及时做出响应，突破 JVM 局限，可以实现对 Java 函数、JNI 函数、系统函数调用的监控，显然较为全面。

[0053] 3、实现对目标应用的精细监控。由于监控单元可以监控目标应用的一切事件行为，对各种函数调用均可无障碍地实施监控，因此，具体到应用层面，本发明不仅可以实现对包括电话、短信、联系人等常规应用的具体操作行为的监控，也可以实现诸如衍生物（安装包）、提权命令、应用加载等高端事件行为的监控，其监控效果更为全面、具体、有效。

[0054] 结合上述的分析可知，本发明提出的上述方案，对现有系统的改动很小，不会影响系统的兼容性，而且实现简单、高效。

[0055] 本发明附加的方面和优点将在下面的描述中部分给出，这些将从下面的描述中变得明显，或通过本发明的实践了解到。

附图说明

[0056] 本发明上述的和 / 或附加的方面和优点从下面结合附图对实施例的描述中将变得明显和容易理解，其中：

[0057] 图 1 为本发明一种免 ROOT 主动防御配置方法的典型实施例的原理示意图；

[0058] 图 2 为本发明的免 ROOT 主动防御配置方法中将安装原包配置成外壳应用的过程的原理示意图；

[0059] 图 3 为本发明一种免 ROOT 主动防御配置装置的结构示意图；

[0060] 图 4 为本发明一种免 ROOT 主动防御方法的典型实施例的原理示意图；

[0061] 图 5 为本发明的免 ROOT 主动防御方法中利用外壳应用的运行对目标应用的事件行为进行监控的原理示意图；

[0062] 图 6 为本发明的免 ROOT 主动防御方法中对捕获的事件进行处理的原理示意图；

[0063] 图 7 为本发明一种免 ROOT 主动防御装置的结构示意图；

[0064] 图 8 为根据本发明实现的一个程序实例的用户界面之一，用于展示发现未防御应用之后的弹框交互功能；

[0065] 图 9 为根据本发明实现的一个程序实例的用户界面之一，用于展示扫描到的应用程序列表，并向用户提供用于确定目标应用的选择区域；

[0066] 图 10 为根据本发明实现的一个程序实例的用户界面之一，用于展示单个应用所有事件行为的默认处理策略，并提供给用户对处理策略进行修改的选项；

[0067] 图 11 为根据本发明实现的一个程序实例的用户界面之一，用于展示进行事件行为拦截后的人机交互效果，具体是拦截发送短信的事件行为；

[0068] 图 12 为根据本发明实现的一个程序实例的用户界面之一，用于展示进行事件行为拦截后的人机交互效果，具体是拦截插入短信的事件行为。

具体实施方式

[0069] 下面详细描述本发明的实施例，所述实施例的示例在附图中示出，其中自始至终相同或类似的标号表示相同或类似的元件或具有相同或类似功能的元件。下面通过参考附图描述的实施例是示例性的，仅用于解释本发明，而不能解释为对本发明的限制。

[0070] 本技术领域技术人员可以理解，除非特意声明，这里使用的单数形式“一”、“一个”、“所述”和“该”也可包括复数形式。应该进一步理解的是，本发明的说明书中使用的措辞“包括”是指存在所述特征、整数、步骤、操作、元件和 / 或组件，但是并不排除存在或添加一个或多个其他特征、整数、步骤、操作、元件、组件和 / 或它们的组。应该理解，当我们称元件被“连接”或“耦接”到另一元件时，它可以直接连接或耦接到其他元件，或者也可以存在中间元件。此外，这里使用的“连接”或“耦接”可以包括无线连接或无线耦接。这里使用的措辞“和 / 或”包括一个或更多个相关联的列出项的全部或任一单元和全部组合。

[0071] 本技术领域技术人员可以理解，除非另外定义，这里使用的所有术语（包括技术术语和科学术语），具有与本发明所属领域中的普通技术人员的一般理解相同的意义。还应该理解的是，诸如通用字典中定义的那些术语，应该被理解为具有与现有技术的上下文中的意义一致的意义，并且除非像这里一样被特定定义，否则不会用理想化或过于正式的含义来解释。

[0072] 本技术领域技术人员可以理解，这里所使用的“终端”、“终端设备”既包括无线信号接收器的设备，其仅具备无发射能力的无线信号接收器的设备，又包括接收和发射硬件的设备，其具有能够在双向通信链路上，执行双向通信的接收和发射硬件的设备。这种设备可以包括：蜂窝或其他通信设备，其具有单线路显示器或多线路显示器或没有多线路显示器的蜂窝或其他通信设备；PCS(Personal Communications Service，个人通信系统)，其可以组合语音、数据处理、传真和 / 或数据通信能力；PDA(Personal Digital Assistant，个人数字助理)，其可以包括射频接收器、寻呼机、互联网 / 内联网访问、网络浏览器、记事本、日历和 / 或 GPS(Global Positioning System，全球定位系统）接收器；常规膝上型和 / 或掌上型计算机或其他设备，其具有和 / 或包括射频接收器的常规膝上型和 / 或掌上型计算机或其他设备。这里所使用的“终端”、“终端设备”可以是便携式、可运输、安装在交通工具（航空、海运和 / 或陆地）中的，或者适合于和 / 或配置为在本地运行，和 / 或以分布形式，运行在地球和 / 或空间的任何其他位置运行。这里所使用的“终端”、“终端设备”还可以是通信终端、上网终端、音乐 / 视频播放终端，例如可以是 PDA、MID(Mobile Internet Device，移动互联网设备）和 / 或具有音乐 / 视频播放功能的移动电话，也可以是智能电视、机顶盒等设备。

[0073] 本技术领域技术人员可以理解，这里所使用的远端网络设备，其包括但不限于计算机、网络主机、单个网络服务器、多个网络服务器集或多个服务器构成的云。在此，云由基于云计算 (Cloud Computing) 的大量计算机或网络服务器构成，其中，云计算是分布式计算

的一种，由一群松散耦合的计算机集组成的一个超级虚拟计算机。本发明的实施例中，远端网络设备、终端设备与 WNS 服务器之间可通过任何通信方式实现通信，包括但不限于，基于 3GPP、LTE、WIMAX 的移动通信、基于 TCP/IP、UDP 协议的计算机网络通信以及基于蓝牙、红外传输标准的近距无线传输方式。

[0074] 本领域技术人员应当理解，本发明所称的“应用”、“应用程序”、“应用软件”以及类似表述的概念，是业内技术人员所公知的相同概念，是指由一系列计算机指令及相关数据资源有机构造的适于电子运行的计算机软件。除非特别指定，这种命名本身不受编程语言种类、级别，也不受其赖以运行的操作系统或平台所限制。理所当然地，此类概念也不受任何形式的终端所限制。同理，本发明所称的“目标应用”、“安装包”之间存在对应关系，安装包为目标应用的文件存在形式。

[0075] 本发明的一种免 ROOT 主动防御配置方法，主要用于为操作系统构建应用程序的安全防御环境，以便在不影响应用程序的正常操作的前提下，实现主动防御。为此本发明将提供一典型实施例用于说明该方法的基本实现。相应的，应用了上述免 ROOT 主动防御配置方法的应用程序，其利用所述配置方法的机理进行工作，也包含一种与前者相应的主动防御方法。为便于说明，以下将以 Unix 系的 Android 操作系统及其应用程序为例，详细说明上述两种方法以及其相应装置的具体实现。

[0076] 本发明的方法所应用的环境包括可与远程服务器或云端通信的移动终端，该移动终端安装有 Android 操作系统，该系统处于未经 ROOT 授权的状态。需要指出的是，即使该操作系统处于 ROOT 授权后的状态，本发明所述的各种方法也依然适用于该操作系统中。也就是说，本发明各种方法的实现，不受操作系统是否开放最高权限所限制。

[0077] 请参阅图 1 的原理示意图，该图揭示了所述免 ROOT 主动防御配置方法的典型实施例，包括如下几大步骤：

[0078] S11、确定目标应用，保存其安装包至指定目录。

[0079] 所述的目标应用，即需要建构主动防御环境的目标应用程序，就特定的处于非 ROOT 授权环境下的 Android 系统而言，出于权限限制的考虑，一般适用于第三方应用。

[0080] 本发明所称的指定目录，是指本发明出于文件组织、管理效率的考虑而为这些需要建构主动防御环境而提供的自定义默认目录，所有通过本发明建立了主动防御环境的目标应用的安装包，均可被移动或复制保存到该指定目录中，进一步还可以对其进行加密或隐藏，以确保其安全性。需要指出的是，这里的指定目录，还可以是系统已经存在的目录。既可以是单个目录，也可以是多个目录。概括而言，是为本发明所采用的用于存放由本发明建构主动防御环境的目标应用安装包的目录。

[0081] 目标应用的确定和处理，非常灵活，以下提供几种确定目标应用和后续处理的实施方式：

[0082] 方式一：

[0083] 对于已经完成安装的应用程序而言，本发明可以自动或受用户指令控制对这些已安装应用程序进行扫描，获得这些应用程序的安装信息，将这些应用程序作为候选目标应用列表显示在用户界面中（参阅图 9），在图形用户界面的相应指示区域中为列表中的每个候选目标应用提供相对应的选择开关，由用户对这些开关状态进行设定，从而获得用户对具体目标应用的确定。具体而言，用户可以将某个目标应用所对应的指示区域中的选择开

关，从未选定状态切换至选定状态，如图 9 中的“已监控”、“点击监控”二态开关示例，这种情况下，即可视为用户完成了对该目标应用的确定操作。

[0084] 众所周知的，Android 系统中，第三方应用的安装会涉及对如下目录做如下操作：data/app，第三方应用安装目录，安装时先把 apk 文件复制到此目录；data/dalvik-cache，将 apk 解压后的代码文件（.dex 文件）安装到该目录下；data/data，用于建立并存放应用程序所需的数据。基于上述原理可知，第三方应用的 apk 文件即为其安装包，在 data/app 中可以找到该安装包。因此，对于已安装的目标应用而言，可以从 data/app 中复制相应的 apk 文件到指定目录中，然后卸载该目标应用。

[0085] 方式二：

[0086] 参阅图 8，对于准备或者正在进行安装的应用程序而言，本发明可以通过将自身注册为默认安装器的形式，获取该应用程序的安装广播信息。继而，将这个新装应用程序作为目标应用，将其安装包或签名之类的特征信息通过远程规则库接口发送到云端服务器中，由云端服务器对其做出安全性判断。一种实施例中，云端服务器为应用程序的安全级别设定黑、灰、白三种级别，分别代表不同危险程度，并设定对应的处理规则。例如，黑应用禁止安装，灰应用由用户自行选择，白应用则可径行安装。当然，可以进一步简化为灰、白两种，或者简化为黑、白两种。本领域技术人员熟悉服务器的这种云端控制技术，将在后续进一步概要揭示。无论如何，本发明将从本机远程规则库接口中获得云端服务器有关这些应用的处理规则的反馈，利用反馈结果做出相应的后续处理。具体而言，当针对当前目标应用返回黑应用标识时，可以随即停止该目标应用的安装；当标识为白应用或灰应用时，则可放行安装。出于交互性的考虑，当完成远程判断后，本发明将向用户界面弹窗提醒用户有关判断结果，并显示相应的处理建议，询问用户是否确定对当前新装应用建构主动防御环境，用户从中确定对当前新装目标应用进行主动防御的标识后，即确定了该目标应用。

[0087] 同理，用户确定该目标应用之后，本发明会将该目标应用的安装包存放至所述的指定目录中。另外，出于本发明后续将为该已确定的目标应用建构主动防御环境的考虑，本发明会立即停止该目标应用的安装，停止安装的操作既可以发明在用户确定该目标应用之前也可以发生在之后。

[0088] 其它变通方式：

[0089] 如前提供的两种典型的有关确定目标应用的方式，可由本领域技术人员变通利用。例如，对于方式一中的已安装目标应用而言，可以适用方式二中将已安装应用通过远程规则库接口发送到云端进行安全等级判断，并在返回结果后，参照方式二的处理方式，对已安装应用进行处理。又如，如果当前应用属于黑应用，而用户仍然希望安装该应用，则仍可允许用户在建立主动防御环境的前提下保留该已安装应用程序，或者允许相应的新装应用继续安装。

[0090] 以上揭示关于确定目标应用的两种典型方式及其变通方式，本领域技术人员足以据此掌握，本发明的主动防御配置方法的首要步骤中涉及如何确定目标应用的多种途径，以及如何获得被确定的目标应用的安装包并将其保存至指定目录中的多种实现方式。

[0091] S12、利用目标应用的安装包配置外壳应用的安装包。

[0092] 确定需要建构主动防御环境的目标应用后，进一步创建外壳应用。请参阅图 2，该外壳应用的创建包括如下具体步骤：

[0093] S121、解析目标应用安装包，生成外壳应用镜像。

[0094] 众所周知，目标应用安装包为压缩文件，将该安装包解压，即可获得其中的文件。较佳的，将目标应用安装包解压至一个临时工作目录以完成解压工作。解压后，即可对目标应用安装包中的各个文件进行解析。另一种方式中，也可以在内存中直接解析该目标应用安装包。无论如何，本领域技术人员均能通过已知方式对目标应用进行解析，获得用于配置外壳应用的相关参数和资源，并据此生成外壳应用镜像。该镜像既可以是硬盘镜像，也可以是内存镜像，其功能在于在构造外壳应用过程中作为中间状态出现，故其具体存在形式应不影响本发明的实现，本领域技术人员可以结合公知常识灵活变通之，下不赘述。S122、修改或替换镜像中的代码文件，以注入所述的桩模块。

[0095] 公知的，apk 安装包的构成文件中包含代码文件 Classes.dex。本发明中，通过修改或者替换的方式，为外壳应用镜像构造新的 Classes.dex，使该新文件中包含本发明所提供的桩模块 nStub。该桩模块通过加载利用 HOOK 技术实现的监控单元 14，使监控单元 14 可在运行时实现对目标应用 15 所创建的进程的事件行为的监控捕获。

[0096] S123、修改镜像中的配置文件的配置参数，用于加载指定目录中的目标应用 15。

[0097] 同理，安装包的构成文件中还包含配置文件 Androidmanifest.xml，对该文件进行修改，对应修改外壳应用镜像中有关目标应用 15 的配置信息，使其适于加载指定目录中的目标应用 15。此外，本发明利用 Java 反射调用机制，将 LoadApk 与 ActivityThread 涉及的运行时配置信息用反射替换成指定目录中目标应用 15 安装包的 ClassLoader 与资源，从而实现外壳应用在运行时对目标应用 15 的加载。

[0098] 此外，图标作为一种可供人机识别的资源，在本发明中也作为配置文件之一被修改。为了使图标更易识别，本发明利用该目标应用 15 的原图标作为底稿，对其添加图戳，以原文件名保存替换原图标，如此，即可在外壳应用安装后，供用户通过该图戳识别其为已防御的应用。同一个目标应用 15 可能包括多个图标资源，可以仅对其中目标应用 15 所采用的主图标进行修改，也可对其包含的多个或所有图标进行类似的修改。

[0099] S124、完成该外壳应用的封装。

[0100] 本子步骤为本领域技术人员所知悉的常规步骤，在完成上述的修改后，对外壳应用镜像进行打包和签名，便可完成外壳应用的封装。签名时，参照公知方式，可以采用手机识别码 IME，或者采用随机码的方式进行签名。

[0101] 经过以上四个子步骤，便可基于目标应用 15 安装包构造相应的外壳应用安装包。可以理解，外壳应用属于轻应用，体积较小，其功能主要表现在对监控单元 14 和对目标应用 15 的先后加载。在运行时，监控单元 14 先被桩模块加载，加载后便开始挂钩后续被加载的目标应用 15 的所有或部分指定的事件行为，实际上相当于将目标应用 15 的事件行为的控制权交到监控单元 14 手中。

[0102] 需要指出的是，所述的监控单元 14，是通过从一后台沙箱 HOOK 框架中获取对应于特定的事件行为的挂钩插件，利用该挂钩插件监控目标应用 15 的特定事件行为而实现的。所述的后台沙箱 HOOK 框架，在云端进行集中管理，向各终端进行分发。其中，云端主要构造有 Java 挂钩插件库和 Native 挂钩插件库。监控单元 14 可以通过远程插件接口向后台沙箱 HOOK 框架发送请求，获得针对特定事件行为的 HOOK 函数，即所述的挂钩插件，借此建立对特定事件行为的监控捕获和处理。

[0103] 由于监控单元14与目标应用15的加载，均为外壳应用进程所驱动，且监控单元14先于目标应用15加载，因而，监控单元14在理论上可以建立对目标应用15一切事件行为的监控。以下概括说明几种典型的事件行为及其捕获实例：

[0104] (1)终端、联网有关的操作：

[0105] 获取运营商信息：目标应用15例如通过getSimOperatorName()函数可以获得移动终端的IMSI，由此可进一步判断运营商的名称，进一步可以向运营商发送约定指令，实现扣费之类的非法目的。监控平台通过挂钩与此相关的消息，便可以对事件行为的捕获。

[0106] 切换APN操作：同理，目标应用15通过与APN切换有关的函数实现ANP切换控制的操作，也可被监控单元14通过调用相应的挂钩插件进行监控。

[0107] 类似的操作，还包括获取手机识别码IME的操作，也与上述同理。

[0108] (2)通知栏广告操作：通知栏广告是最易被恶意程序利用的手段，监控单元14通过调用相应的挂钩插件对notify函数产生的事件消息进行监控，也可对其实施监控。

[0109] (3)通信操作：

[0110] 如电话拨打操作，通过StartActivity()函数可以监控拨打电话的事件行为，利用相应的挂钩插件可以对拔打电话操作建立事件行为监控。

[0111] 短信操作，对应于SendTextMessage()之类的函数，同理，可以借助挂钩插件对这类函数建立事件行为监控。

[0112] 联系人操作：一般对应于Query()、Insert()函数，监控单元14利用挂钩插件挂钩此类函数可以实现对此类事件行为的监控捕获。

[0113] (4)命令操作：

[0114] 如SU提权操作或执行命令操作，均需用到Execve()函数，监控单元14通过监控此函数的返回消息，便可实现该类事件行为的监控。

[0115] (5)界面及访问操作：

[0116] 如创造快捷方式的事件行为，则对应于SentBroacast()函数。同理，对于隐藏程序图标的操作，也可对应特定函数监控之。

[0117] 如HTTP网络访问操作，则对应于Sentto()、Write()等函数。

[0118] (6)程序操作：

[0119] 如应用加载操作，指当前目标应用15加载相关应用的操作，通过对dexclassloader()、loadlibrary()等函数进行挂钩监控，可以实现对此类事件行为的捕获。

[0120] 又如安装子包，则对应于StartActivity()函数。

[0121] (7)其它危险操作：

[0122] 例如，子进程侵入操作、衍生物操作、激活设备管理器操作等，分别对应于。

[0123] 其中，子进程是指目标应用15建立的子进程，在目标应用15创建子进程时，监控单元14将收到相应的消息，而判定其创建子进程的事件行为。由此，监控单元14进一步向该子进程以内联钩子的方式在该子进程中植入监控单元14，后续便可继续对该子进程的事件行为进行监控。因而，无论是目标应用15的自身进程，还是其创建的子进程，它们直接或间接所触发的事件行为，均能被本发明的监控单元14所监控，使主动防御效果更佳。

[0124] 而所述衍生物，是指目标应用15自行创建的文件，或者远程下载的文件，通常是

指敏感的衍生物，例如安装包。通过挂钩 fclose() 函数可以捕获该事件。需要指出的是，当监控单元 14 捕获该事件行为后，可以按照前述的方法，进一步利用远程规则库接口发送请求到云端，由云端利用其黑、白、灰的安全等级行为规则判断该衍生物的安全等级，本发明通过远程规则库接口获得云端判定结果后，进一步弹窗询问用户是否建立对该敏感衍生物的主动防御，由此便可进一步巩固主动防御的效果。

[0125] 上述的事件行为仅为摘录之用，不能理解为对本发明监控的事件行为的限制。结合上述对事件行为的分类可知，本发明的监控单元 14，可以对源自目标应用 15 的事件行为、无论是由目标应用 15 直接还是间接触发的事件行为进行监控。

[0126] 本发明的外壳应用安装包的文件名与目标应用 15 的安装包的文件名完全一致，因此，可以看出，外壳应用构成了目标应用 15 的伪装应用。外壳应用体积小，其构造过程较为迅速，对用户而言其构造和运行过程较为透明，基本不影响进行主动防御环境构建时的目标应用 15 安装和运行效率。

[0127] 此外，为了便于实现用户交互，本发明还为外壳应用配备一交互接口，通过该交互接口，可以向预注册的系统服务发送消息，通过系统服务向用户界面弹窗问询用户指令，系统服务获得用户指令后返回给本外壳应用的进程，外壳应用根据用户指令可以做前述所称的系列的后续处理，这一后续处理部分将在后续涉及主动防御方法部分进行详细的揭示。

[0128] S13、安装该外壳应用。

[0129] 完成该外壳应用的构造后，本发明将该外壳应用进行安装，而后，该目标应用 15 即具有了前述所阐述的主动防御环境，用户运行该目标应用 15，会被引导至运行文件名相同的外壳应用，外壳应用一旦运行，便能实现对该目标应用 15 的主动防御。

[0130] 由于本发明的方法的应用环境为非 ROOT 授权的环境，部分权限受限制，这种情况下，如果已安装目标应用 15 未卸载，则会先弹出一卸载该目标应用 15 的界面，引导用户卸载已装目标应用 15 ；继而弹出一安装该外壳应用的界面，引导客户安装该外壳应用。当然，如果系统已获 ROOT 授权，本发明的方法可径行将其旧应用卸载然后安装外壳应用。

[0131] 需要进一步的强调的是，前述提及的卸载目标应用 15 的子步骤，如本步骤所述，可以按需被后续处理，其卸载的时点，并不影响本发明的方法的实现。

[0132] 以上仅仅阐述了本发明的主动防御配置方法，进一步，可以利用该主动防御配置方法构建相应的主动防御配置装置。

[0133] 请参阅图 3，本发明的免 ROOT 主动防御配置装置与前述配置方法具有严密的对应性，包括确定装置 11、构造装置 12 以及安装装置 13，如下进行具体阐述 ：

[0134] 所述的确定装置 11，用于确定目标应用 15，并将该目标应用 15 的安装包保存至指定目录。

[0135] 所述的目标应用 15，即需要建构主动防御环境的目标应用 15 程序，就特定的处于非 ROOT 授权环境下的 Android 系统而言，出于权限限制的考虑，一般适用于第三方应用。

[0136] 本发明所称的指定目录，是指本发明出于文件组织、管理效率的考虑而为这些需要建构主动防御环境而提供的自定义默认目录，所有通过本发明建立了主动防御环境的目标应用 15 的安装包，均可被移动或复制保存到该指定目录中，进一步还可以对其进行加密或隐藏，以确保其安全性。需要指出的是，这里的指定目录，还可以是系统已经存在的目录。既可以是单个目录，也可以是多个目录。概括而言，是为本发明所采用的用于存放由本发明

建构主动防御环境的目标应用 15 安装包的目录。

[0137] 确定装置 11 的构造，非常灵活，以下提供几种构造该确定装置 11 的实施方式：

[0138] 方式一：

[0139] 对于已经完成安装的应用程序而言，本发明可以自动或受用户指令控制对这些已安装应用程序进行扫描，获得这些应用程序的安装信息，通过一选定单元将这些应用程序作为候选目标应用 15 列表如图 9 所示的显示在用户界面中，在图形用户界面的相应指示区域中为列表中的每个候选目标应用 15 提供相对应的选择开关，由用户对这些开关状态进行设定，从而获得用户对具体目标应用 15 的确定。具体而言，用户可以将某个目标应用 15 所对应的指示区域中的选择开关，从未选定状态切换至选定状态，这种情况下，即可视为用户完成了对该目标应用 15 的确定操作。

[0140] 众所周知的，Android 系统中，第三方应用的安装会涉及对如下目录做如下操作：data/app，第三方应用安装目录，安装时先把 apk 文件复制到此目录；data/dalvik-cache，将 apk 解压后的代码文件（.dex 文件）安装到该目录下；data/data，用于建立并存放应用程序所需的数据。基于上述原理可知，第三方应用的 apk 文件即为其安装包，在 data/app 中可以找到该安装包。因此，对于已安装的目标应用 15 而言，本发明进而为确定装置 11 构造一处理单元，由其从 data/app 中复制相应的 apk 文件到指定目录中，然后卸载该目标应用 15。

[0141] 方式二：

[0142] 参阅图 8，对于准备或者正在进行安装的应用程序而言，本发明可以通过将自身注册为默认安装器的形式，通过一选定单元获取该应用程序的安装广播信息。继而，将这个新装应用程序作为目标应用 15，将其安装包或签名之类的特征信息通过远程规则库接口发送到云端服务器中，由云端服务器对其做出安全性判断。一种实施例中，云端服务器为应用程序的安全级别设定黑、灰、白三种级别，分别代表不同危险程度，并设定对应的处理规则。例如，黑应用禁止安装，灰应用由用户自行选择，白应用则可径行安装。当然，可以进一步简化为灰、白两种，或者简化为黑、白两种。本领域技术人员熟悉服务器的这种云端控制技术，恕不赘述。无论如何，本发明将从本机远程规则库接口中获得云端服务器有关这些应用的处理规则的反馈，利用反馈结果做出相应的后续处理。具体而言，当针对当前目标应用 15 返回黑应用标识时，可以随即停止该目标应用 15 的安装；当标识为白应用或灰应用时，则可放行安装。出于交互性的考虑，当完成远程判断后，本发明将向用户界面弹窗提醒用户有关判断结果，并显示相应的处理建议，询问用户是否确定对当前新装应用建构主动防御环境，用户从中确定对当前新装目标应用 15 进行主动防御的标识后，即确定了该目标应用 15。

[0143] 同理，用户确定该目标应用 15 之后，本发明会将该目标应用 15 的安装包存放至所述的指定目录中。另外，出于本发明后续将为该已确定的目标应用 15 建构主动防御环境的考虑，本发明会由一处理单元立即停止该目标应用 15 的安装，停止安装的操作既可以发明在用户确定该目标应用 15 之前也可以发生在之后。

[0144] 其它变通方式：

[0145] 如前提供的两种典型的有关确定目标应用 15 的方式，可由本领域技术人员变通利用。例如，对于方式一中的已安装目标应用 15 而言，可以适用方式二中将已安装应用通过远程规则库接口发送到云端进行安全等级判断，并在返回结果后，参照方式二的处理方

式，对已安装应用进行处理。又如，如果当前应用属于黑应用，而用户仍然希望安装该应用，则仍可允许用户在建立主动防御环境的前提下保留该已安装应用程序，或者允许相应的新装应用继续安装。

[0146] 以上揭示关于确定装置 11 的两种典型构造方式及其变通方式，本领域技术人员足以据此掌握，本发明的主动防御配置装置的确定装置 11 中涉及如何确定目标应用 15 的多种途径，以及如何获得被确定的目标应用 15 的安装包并将其保存至指定目录中的多种实现方式。

[0147] 所述的构造装置 12，其利用目标应用 15 的安装包配置外壳应用的安装包。

[0148] 确定需要建构主动防御环境的目标应用 15 后，进一步创建外壳应用。该构造装置 12 包括解析单元、代码单元、配置单元以及封装单元，以下详细揭示这些单元的功能实现：

[0149] 所述的解析单元，用于解析目标应用 15 安装包，生成外壳应用镜像。

[0150] 众所周知，目标应用 15 安装包为压缩文件，将该安装包解压，即可获得其中的文件。较佳的，将目标应用 15 安装包解压至一个临时工作目录以完成解压工作。解压后，即可对目标应用安装包中的各个文件进行解析。另一种方式中，也可以在内存中直接解析该目标应用安装包。无论如何，本领域技术人员均能通过已知方式对目标应用进行解析，获得用于配置外壳应用的相关参数和资源，并据此生成外壳应用镜像。

[0151] 所述的代码单元，用于修改或替换镜像中的代码文件，以注入所述的桩模块。

[0152] 公知的，apk 安装包的构成文件中包含代码文件 Classes.dex。本发明中，通过修改或者替换的方式，构造新的 Classes.dex，使该新文件中包含本发明所提供的桩模块 nStub。该桩模块通过加载利用 HOOK 技术实现的监控单元 14，使监控单元 14 可在运行时实现对目标应用 15 所创建的进程的事件行为的监控捕获。

[0153] 所述的配置单元，用于修改镜像中的配置文件的配置参数，以用于加载指定目录中的目标应用 15。

[0154] 同理，安装包的构成文件中还包含配置文件 Androidmanifest.xml，对该文件进行修改，对应修改外壳应用镜像中有关目标应用 15 的配置信息，使其适于加载指定目录中的目标应用 15。此外，本发明利用 Java 反射调用机制，将 LoadApk 与 ActivityThread 涉及的运行时配置信息用反射替换成指定目录中目标应用 15 安装包的 ClassLoader 与资源，从而实现外壳应用在运行时对目标应用 15 的加载。

[0155] 此外，图标作为一种可供人机识别的资源，在本发明中也作为配置文件之一被修改。为了使图标更易识别，本发明利用该目标应用 15 的原图标作为底稿，对其添加图戳，以原文件名保存替换原图标，如此，即可在外壳应用安装后，供用户通过该图戳识别其为已防御的应用。同一个目标应用 15 可能包括多个图标资源，可以仅对其中目标应用 15 所采用的主图标进行修改，也可对其包含的多个或所有图标进行类似的修改。

[0156] 所述的封装单元，用于完成该外壳应用的封装。

[0157] 封装单元的功能实现为本领域技术人员所应理解。在完成上述的修改后，对外壳应用镜像进行打包和签名，便可完成外壳应用的封装。签名时，参照公知方式，可以采用手机识别码 IME，或者采用随机码的方式进行签名。

[0158] 通过执行该构造装置 12，便可基于目标应用 15 安装包构造相应的外壳应用安装包。可以理解，外壳应用属于轻应用，体积较小，其功能主要表现在对监控单元 14 和对目标

应用 15 的先后加载。在运行时，监控单元 14 先被桩模块加载，加载后便开始挂钩后续被加载的目标应用 15 的所有或部分指定的事件行为，实际上相当于将目标应用 15 的事件行为的控制权交到监控单元 14 手中。

[0159] 需要指出的是，所述的监控单元 14，是通过从一后台沙箱 HOOK 框架中获取对应于特定的事件行为的挂钩插件，利用该挂钩插件监控目标应用 15 的特定事件行为而实现的。所述的后台沙箱 HOOK 框架，在云端进行集中管理，向各终端进行分发。其中，云端主要构造有 Java 挂钩插件库和 Native 挂钩插件库。监控单元 14 可以通过远程插件接口向后台沙箱 HOOK 框架发送请求，获得针对特定事件行为的 HOOK 函数，即所述的挂钩插件，借此建立对特定事件行为的监控捕获和处理。

[0160] 由于监控单元 14 与目标应用 15 的加载，均为外壳应用进程所驱动，且监控单元 14 先于目标应用 15 加载，因而，监控单元 14 在理论上可以建立对目标应用 15 一切事件行为的监控。

[0161] 有关本发明的主动防御配置装置中监控单元 14 所处理的事件行为，由于与上述主动防御配置方法具有严密对应性，故不赘述。

[0162] 同理，本发明的外壳应用安装包的文件名与目标应用 15 的安装包的文件名完全一致，因此，可以看出，外壳应用构成了目标应用 15 的伪装应用。外壳应用体积小，其构造过程较为迅速，对用户而言构造和运行过程较为透明，基本不影响进行主动防御环境构建时的目标应用 15 安装和运行效率。

[0163] 此外，为了便于实现用户交互，本发明还为外壳应用配备一交互接口，通过该交互接口，可以向预注册的系统服务发送消息，通过系统服务向用户界面弹窗问询用户指令，系统服务获得用户指令后返回给本外壳应用的进程，外壳应用根据用户指令可以做前述所称的系列的后续处理，这一后续处理部分将在后续涉及主动防御方法部分进行详细的揭示。

[0164] 所述的安装装置 13，用于安装该外壳应用。

[0165] 完成该外壳应用后，执行安装装置 13，以便直接安装该外壳应用，安装完成后，该目标应用 15 即具有了前述所阐述的主动防御环境，用户运行该目标应用 15，会被引导至运行文件名相同的外壳应用，外壳应用一时运行，便能实现对该目标应用 15 的主动防御。

[0166] 由于本发明的方法的应用环境为非 ROOT 授权的环境，部分权限受限制，这种情况下，如果已安装目标应用 15 未卸载，则会先弹出一卸载该目标应用 15 的界面，引导用户卸载已装目标应用 15；继而弹出一安装该外壳应用的界面，引导客户安装该外壳应用。当然，如果系统已获 ROOT 授权，本发明的方法可径行将其旧应用卸载然后安装外壳应用。

[0167] 需要进一步的强调的是，前述提及的卸载目标应用 15 的子步骤，如本步骤所述，可以按需被后续处理，其卸载的时点，并不影响本发明的方法的实现。

[0168] 本发明在前述的方法和装置中为应用程序建构了主动防御环境，在此基础上，从程序执行的视角，还提供了一种免 ROOT 主动防御方法和一种免 ROOT 主动防御装置。

[0169] 请参阅图 4，本发明的免 ROOT 主动防御方法，是前述主动防御配置方法中构建的主动防御环境的具体应用，该方法建基于构建了主动防御环境配置的目标应用 15，对目标应用 15 实施安全防护。结合图 7，该方法包括如下步骤：

[0170] S31、响应运行目标应用 35 的指令，运行相应的外壳应用。

[0171] 参阅前述配置方法的说明可知，外壳应用被安装后，其文件名与原来的目标应用

35 的文件名相同，伪装成目标应用 35，用户对目标应用 35 的操作，实际上，通过桌面图标指引的快捷方式，将被引导至运行预先伪装的外壳应用，此时，用户在用户界面上的点选操作便构成运行该外壳应用的运行指令。需要指出的是，运行目标应用 35 的指令并不局限于由用户触发，也包括如前所述的，由应用程序、定时任务或者通过其它公知途径以函数调用的方式执行的加载指令。外壳应用为轻应用，可以快速加载至内存中运行，对用户而言，其启动过程是透明的。

[0172] 外壳应用的图标是从目标应用 35 的默认图标改进的，一般是以该默认图标加图戳来实现这种改进，因而，从视觉效果上，还可起到一定的示警作用。

[0173] 一旦产生运行目标应用 35 的指令，本发明即作出响应，外壳应用随即被加载到 JAVA 虚拟机中运行。

[0174] S32、外壳应用的加载过程。

[0175] 如前的配置方法所述，本发明的外壳应用中，其代码文件 Classes.dex 配置有桩模块 nstub，通过该桩模块可以加载监控模块；其配置文件 Androidmanifest.xml 运用 Java 反射调用原理，对其中的配置参数进行修改，使其适于加载保存在所述指定目录中的目标应用 35，此外，还对目标应用 35 的运行时配置参数进行了适应性的修改，确定目标应用 35 能正常运行。

[0176] 因此，请参阅图 5，外壳应用运行后，如步骤 S321 所揭示，首先通过桩模块调用加载监控单元 34，所述监控单元 34 从一后台沙箱 HOOK 框架中获取对应于特定的事件行为的挂钩插件，利用该挂钩插件挂钩并监控目标应用 35 的特定事件行为。所述的后台沙箱 HOOK 框架，在云端进行集中管理，向各终端进行分发。其中，云端主要构造有 Java 挂钩插件库和 Native 挂钩插件库。监控单元 34 需要挂钩具体事件行为时，通过远程插件接口向后台沙箱 HOOK 框架发送请求，获得针对特定事件行为的 HOOK 函数，即所述的挂钩插件，借此建立对特定事件行为的监控捕获和处理。

[0177] 进而，如步骤 S322 所揭示，运行中的外壳应用将进一步加载所述位于指定目录中的目标应用 35。如前所述，目标应用 35 调用，是利用公知的 Java 反射调用机制实现的。外壳应用的进程将 LoadApk 与 ActivityThread 涉及的运行时配置信息用反射替换成指定目录中目标应用 35 安装包的 ClassLoader 与资源，从而实现对目标应用 35 的加载。

[0178] 如步骤 S323 显示，目标应用 35 被加载时，已被监控单元 34 利用挂钩插件建立了监控，因此，目标应用 35 的一切事件行为均在监控单元 34 的监控范围之内。位于目标应用 35 的安装包是完整未经修改的，因此，目标应用 35 被外壳应用加载后，能够完全合法、正常地运行，实现目标应用 35 原本能实现的所有功能。

[0179] 由于监控单元 34 与目标应用 35 的加载，均为外壳应用进程所驱动，同为外壳应用进程的一部分，且监控单元 34 先于目标应用 35 加载，因而，运行中的监控单元 34 即建立了对目标应用 35 一切事件行为的监控。目标应用 35 运行过程中产生的任何事件行为，其事件消息均会被监控单元 34 捕获并进行相应的处理。

[0180] S33、捕获事件行为之后的处理过程。

[0181] 请结合图 6，步骤 S331 显示，目标应用 35 产生的特定事件行为被监控单元 34 捕获，实质上是触发特定事件行为时，所产生的事件消息被监控单元 34 中相应的挂钩插件（钩子函数）所捕获。捕获该事件消息，即可知晓该事件的意图，继而可以进行后续的处理。

[0182] 步骤 S332 显示，对特定事件行为进行处理，需要获取事件行为处理策略。在这一子步骤中，可以进一步借助系统服务来实现人机交互功能。为了实现人机交互效果，本发明预先将一交互模块注册为系统服务，外壳应用可以通过其交互接口与该交互模块通信，从而实现外壳应用对用户指令或预设指令的获取。

[0183] 事件行为策略的获取方式非常灵活多样，以下列举几种为本发明所择一或任意组合使用的策略：

[0184] (1) 监控单元 34 捕获特定事件行为后，通过外壳应用内建的交互接口，向所述交互模块发送请求，由交互模块向用户界面弹窗问询用户处理策略，如图 11 和图 12 所示，该弹窗界面可以直接告知用户有关事件行为的内容及其风险，由用户选择相应的选项作为处理策略。用户选择相应选项并确定后，交互模块获得针对该特定事件行为的处理策略，将其反馈给监控单元 34，监控单元 34 即可根据该用户指令所产生的处理策略对目标应用 35 的相应事件行为进行下一步的处理。

[0185] (2) 在某些已被公认为相对低风险的事件行为发生时，例如对联系人的只读操作行为，或者在用户为本发明设置了自行检索针对特定事件行为所应采取的处理策略时，本发明利用一本地策略数据库检索相应的针对特定事件行为的处理策略。例如，如图 10 所示，某个应用的所有事件行为的默认处理策略可以被以表单的形式给出。也就是说，该本地策略数据库中，建立了特定事件行为与相应的处理策略之间的关联，并且存储了多种事件行为与相应的处理策略之间对应关系的记录数据，可以供本发明检索使用。本发明从本地策略数据库中获取相应的处理策略后，方能对相应事件行为做下一步的处理。

[0186] (3) 如果用户为本发明设置了远程获取处理策略的选项，或者默认在本地策略数据库检索不到特定事件行为的具体策略时可以远程获取，又或通过前述第 (1) 种情况进行交互而在规定时限内得不到用户对弹窗的响应，诸如此类的情况，外壳应用均可通过其内建的远程策略接口，向预架构的云端发送请求，获得对应于该特定事件行为的相应的处理策略，并用于后续的处理。

[0187] 需要指出的是，有关以上三种获取处理策略的方式，可以交叉配合使用，例如，一旦交互模块接收到监控单元 34 传递的事件消息的特征，即可依照默认设置，参照第 (2) 种方式先行检索本地策略数据库，获得系统推荐的处理策略（如果不能从本地策略数据库中获得，甚至可以进一步按第 (3) 种方式从云端策略数据库中获取）。继而，参照第 (1) 种方式，在弹窗界面设置系统推荐的处理策略为默认选项。如果用户未在规定时限内确认该默认选项，则以系统推荐的处理策略为准执行后续指令；如果用户将之改变为新的默认选项，则向监控单元 34 返回用户设置的处理策略。可见，人机交互过程是可以更为灵活自由地实现的。

[0188] 所述的本地策略数据库，可以是云端策略数据库的一个复件，因此，本发明中，设置一个更新步骤，用于下载云端策略数据库用于更新本地策略数据库。

[0189] 一般情况下，针对特定事件行为的策略可以设置为“拒绝”、“运行”、“询问”三个常见选项，其表征的具体意向为：

[0190] 拒绝：针对该特定事件行为，向目标应用 35 发送事件行为已经执行完毕的虚假消息，以禁止该事件行为实际发生；

[0191] 运行：针对该特定事件行为不做任何改变，将相应的事件消息直接转送给系统消

息机制，允许目标应用 35 继续其事件行为；

[0192] 询问：独立或依附于前述两个选项任意之一，针对该特定事件行为，标记其状态为未知状态，后续重复发生该行为时，需要再行弹窗询问用户。

[0193] 实际应用中，选项“询问”可被忽略，仅需考虑是否拒绝或允许当前事件行为发生即可。

[0194] 所述的事件行为，多种多样，具体包括如下几大类型：

[0195] (1) 终端、联网有关的操作：

[0196] 获取运营商信息：目标应用 35 例如通过 getSimOperatorName() 函数可以获得移动终端的 IMSI，由此可进一步判断运营商的名称，进一步可以向运营商发送约定指令，实现扣费之类的非法目的。监控平台通过挂钩与此相关的消息，便可以对事件行为的捕获。

[0197] 切换 APN 操作：同理，目标应用 35 通过与 APN 切换有关的函数实现 ANP 切换控制的操作，也可被监控单元 34 通过调用相应的挂钩插件进行监控。

[0198] 类似的操作，还包括获取手机识别码 IME 的操作，也与上述同理。

[0199] (2) 通知栏广告操作：通知栏广告是最易被恶意程序利用的手段，监控单元 34 通过调用相应的挂钩插件对 notify 函数产生的事件消息进行监控，也可对其实施监控。

[0200] (3) 通信操作：

[0201] 如电话拨打操作，通过 StartActivity() 函数可以监控拨打电话的事件行为，利用相应的挂钩插件可以对拨打电话操作建立事件行为监控。

[0202] 短信操作，对应于 SendTextMessage() 之类的函数，同理，可以借助挂钩插件对这类函数建立事件行为监控。

[0203] 联系人操作：一般对应于 Query()、Insert() 函数，监控单元 34 利用挂钩插件挂钩此类函数可以实现对此类事件行为的监控捕获。

[0204] (4) 命令操作：

[0205] 如 SU 提权操作或执行命令操作，均需用到 Execve() 函数，监控单元 34 通过监控此函数的返回消息，便可实现该类事件行为的监控。

[0206] (5) 界面及访问操作：

[0207] 如创造快捷方式的事件行为，则对应于 SentBroacast() 函数。同理，对于隐藏程序图标的操作，也可对应特定函数监控之。

[0208] 如 HTTP 网络访问操作，则对应于 Sentto()、Write() 等函数。

[0209] (6) 程序操作：

[0210] 如应用加载操作，指当前目标应用 35 加载相关应用的操作，通过对 dexclassloader()、loadlibrary() 等函数进行挂钩监控，可以实现对此类事件行为的捕获。

[0211] 又如安装子包，则对应于 StartActivity() 函数。

[0212] (7) 其它危险操作：

[0213] 例如，子进程侵入操作、衍生物操作、激活设备管理器操作等，分别对应于。

[0214] 其中，子进程是指目标应用 35 建立的子进程，在目标应用 35 创建子进程时，监控单元 34 将收到相应的消息，而判定其创建子进程的事件行为。由此，监控单元 34 进一步向该子进程以内联钩子的方式在该子进程中植入监控单元 34，后续便可继续对该子进程的事

件行为进行监控。因而，无论是目标应用 35 的自身进程，还是其创建的子进程，它们直接或间接所触发的事件行为，均能被本发明的监控单元 34 所监控，使主动防御效果更佳。

[0215] 而所述衍生物，是指目标应用 35 自行创建的文件，或者远程下载的文件，通常是指敏感的衍生物，例如安装包。通过挂钩 fclose() 函数可以捕获该事件。需要指出的是，当监控单元 34 捕获该事件行为后，可以按照前述的方法，进一步利用远程规则库接口发送请求到云端，由云端利用其黑、白、灰的安全等级行为规则判断该衍生物的安全等级，本发明通过远程规则库接口获得云端判定结果后，进一步弹窗询问用户是否建立对该敏感衍生物的主动防御，由此便可进一步巩固主动防御的效果。

[0216] 上述的事件行为仅为摘录之用，不能理解为对本发明监控的事件行为的限制。

[0217] 步骤 S333 显示，依据上述的处理策略和上述关于事件行为的说明，本发明的主动防御方法便可对各种事件行为进行相应的处理，其处理过程的概括在前文中已散列给出，以下进一步列举几种典型的应用实例 ：

[0218] (1) 对目标应用 35 的精细拦截的应用 ：

[0219] 部分恶意程序被安装后，在相当长的一段时间内处于正常使用的状态，麻痹用户的安全意识。但是，运行一段长时间之后，该目标应用 35 尝试从后台插入一短信引起用户的关注，达到广告和诈骗的效果。参阅图 12，对该目标应用 35 建立主动防御机制后，本发明如前所述，通过监控单元 34 中相应的挂钩插件对短信操作函数的监控，一旦目标应用 35 产生短信操作的事件行为，便可捕获这一事件行为，继而，监控单元 34 通过其交互接口通知作为系统服务运行的交互模块，由交互模块向用户界面弹窗示警。用户点选“拒绝”的处理策略后，被逆反馈给监控单元 34，其中相应的挂钩插件便能阻该事件行为的实际发生，达到防范风险的目的。

[0220] (2) 对目标应用 35 释放恶意文件的应用。

[0221] 目标应用 35 为一游戏软件，通过检查更新的方式下载并释放恶意子包，并且调用系统功能安装该子包。本发明对该目标应用 35 建立了主动防御之后，可以监控到其下载完文件而产生的事件行为，据此通过交互模块弹窗告警。用户指令拒绝之后，监控单元 34 中相应的挂钩插件便可直接删除该文件，或者仅仅拒绝该文件的安装行为。

[0222] 本发明中，对于诸如此类的恶意子包，视为敏感衍生物，对衍生物是否存在恶意的判断，参照前述防御配置方法中所述及的确定安全等级的方式进行远程判断。具体而言，当检测到产生衍生物时，将相应的文件或者其签名之类的特征信息通过远程规则库接口发送给云端，并从云端获得其安全等级，如果为黑、灰应用，则在弹窗中建议用户拒绝安装 ；如果为白应用，则可允许其通行。通过这种方法，便可实现对敏感衍生物的安全防御。如果云端检测不到该衍生物的相关记录，可以要求本方法为其上传该文件，并由云端标示为未知应用，相应的，以灰应用予以标记，以备后用。

[0223] (3) 对子进程侵入的应用。

[0224] 被监控的目标应用 35 在运行过程中创建子进程，而子进程进一步释放恶意事件行为。监控单元 34 监控到目标应用 35 创建子进程时，即获得子进程的入口，然后向该子进程植入本发明的监控单元 34，所有 HOOK 插件（挂钩插件）都会被以内联钩子的方式加载到该子进程中并初始化好实现挂钩，以便建立对该子进程的事件行为的监控。由此，可以看出，无论是由目标应用 35 进程直接触发的事件行为，还是由目标应用 35 进程所创建的子进

程所触发的间接事件行为，均能被监控单元 34 成功监控。

[0225] 以上通过 S31、S32、S33 共三个关键步骤，详细描述了本发明的主动防御方法的实现及其应用，可以看出，以该方法工作的主动防御技术，具有充分的可行性。

[0226] 进一步，适应上述免 ROOT 主动防御方法，本发明进一步提供一种免 ROOT 主动防御装置，两者也自然具有严密的对应性，以下对该装置进行具体揭示：

[0227] 本发明的免 ROOT 主动防御装置，包括启动模块 31、安防模块 32 以及处理模块 33，各模块的具体功能及实现如下所示：

[0228] 所述的启动模块 31，用于响应运行目标应用 35 的指令，运行相应的外壳应用。

[0229] 参阅前述配置方法的说明可知，外壳应用被安装后，其文件名与原来的目标应用 35 的文件名相同，伪装成目标应用 35，用户对目标应用 35 的操作，实际上，通过桌面图标指引的快捷方式，将被引导至运行预先伪装的外壳应用，此时，用户在用户界面上的点选操作便构成运行该外壳应用的运行指令。需要指出的是，运行目标应用 35 的指令并不局限于由用户触发，也包括如前所述的，由应用程序、定时任务或者通过其它公知途径以函数调用的方式执行的加载指令。外壳应用为轻应用，可以快速加载至内存中运行，对用户而言，其启动过程是透明的。

[0230] 外壳应用的图标是从目标应用 35 的默认图标改进的，一般是以该默认图标加图戳来实现这种改进，因而，从视觉效果上，还可起到一定的示警作用。

[0231] 一旦产生运行目标应用 35 的指令，本发明即作出响应，外壳应用随即被加载到 JAVA 虚拟机中运行。

[0232] 所述的安防模块 32，其主要实现外壳应用的加载过程，利用外壳应用先后加载监控单元 34 及所述目标应用 35，由该监控单元 34 对该目标应用 35 的事件行为进行监控。

[0233] 本发明的外壳应用中，其代码文件 Classes.dex 配置有桩模块 nstub，通过该桩模块可以加载监控模块；其配置文件 Androidmanifest.xml 运用 Java 反射调用原理，对其中的配置参数进行修改，使其适于加载保存在所述指定目录中的目标应用 35，此外，还对目标应用 35 的运行时配置参数进行了适应性的修改，确定目标应用 35 能正常运行。

[0234] 因此，外壳应用运行后，首先通过桩模块调用监控单元 34，所述监控单元 34 从一后台沙箱 HOOK 框架中获取对应于特定的事件行为的挂钩插件，利用该挂钩插件挂钩并监控目标应用 35 的特定事件行为。所述的后台沙箱 HOOK 框架，在云端进行集中管理，向各终端进行分发。其中，云端主要构造有 Java 挂钩插件库和 Native 挂钩插件库。监控单元 34 需要挂钩具体事件行为时，通过远程插件接口向后台沙箱 HOOK 框架发送请求，获得针对特定事件行为的 HOOK 函数，即所述的挂钩插件，借此建立对特定事件行为的监控捕获和处理。

[0235] 进而，运行中的外壳应用将进一步加载所述位于指定目录中的目标应用 35。如前所述，目标应用 35 调用，是利用公知的 Java 反射调用机制实现的。安防模块 32 中构造有配置模块，其由外壳应用的进程将 LoadApk 与 ActivityThread 涉及的运行时配置信息用反射替换成指定目录中目标应用 35 安装包的 ClassLoader 与资源，从而实现对目标应用 35 的加载。目标应用 35 被加载时，已被监控单元 34 利用挂钩插件建立了监控，因此，目标应用 35 的一切事件行为均在监控单元 34 的监控范围之内。位于目标应用 35 的安装包是完整未经修改的，因此，目标应用 35 被外壳应用加载后，能够完全合法、正常地运行，实现目

标应用 35 原本能实现的所有功能。

[0236] 由于监控单元 34 与目标应用 35 的加载，均为外壳应用进程所驱动，同为外壳应用进程的一部分，且监控单元 34 先于目标应用 35 加载，因而，运行中的监控单元 34 即建立了对目标应用 35 一切事件行为的监控。目标应用 35 运行过程中产生的任何事件行为，其事件消息均会被监控单元 34 捕获并进行相应的处理。

[0237] 所述的处理模块 33，用于执行捕获事件行为之后的处理过程。

[0238] 目标应用 35 产生的特定事件行为被监控单元 34 捕获，实质上是触发特定事件行为时，所产生的事件消息被监控单元 34 中相应的挂钩插件（钩子函数）所捕获。捕获该事件消息，即可知晓该事件的意图，继而可以进行后续的处理。

[0239] 对特定事件行为进行处理，需要获取事件行为处理策略。在这一子步骤中，可以进一步借助系统服务来实现人机交互功能。为了实现人机交互效果，本发明预先将一交互模块注册为系统服务，外壳应用可以通过其交互接口与该交互模块通信，从而实现外壳应用对用户指令或预设指令的获取。

[0240] 如前所述，事件行为策略的获取方式非常灵活多样，通过构造一策略生成装置来执行，以下列举几种为本发明所择一或任意组合使用的策略：

[0241] (1) 监控单元 34 捕获特定事件行为后，通过外壳应用内建的交互接口，向所述交互模块发送请求，由交互模块向用户界面弹窗问询用户处理策略，该弹窗界面可以直接告知用户有关事件行为的内容及其风险，由用户选择相应的选项作为处理策略。用户选择相应选项并确定后，交互模块获得针对该特定事件行为的处理策略，将其反馈给监控单元 34，监控单元 34 即可根据该用户指令所产生的处理策略对目标应用 35 的相应事件行为进行下一步的处理。

[0242] (2) 在某些已被公认为相对低风险的事件行为发生时，例如对联系人的只读操作行为，或者在用户为本发明设置了自行检索针对特定事件行为所应采取的处理策略时，本发明利用一本地策略数据库检索相应的针对特定事件行为的处理策略。也就是说，该本地策略数据库中，建立了特定事件行为与相应的处理策略之间的关联，并且存储了多种事件行为与相应的处理策略之间对应关系的记录数据，可以供本发明检索使用。本发明从本地策略数据库中获取相应的处理策略后，方能对相应事件行为做下一步的处理。

[0243] (3) 如果用户为本发明设置了远程获取处理策略的选项，或者默认在本地策略数据库检索不到特定事件行为的具体策略时可以远程获取，又或通过前述第 (1) 种情况进行交互而在规定时限内得不到用户对弹窗的响应，诸如此类的情况，外壳应用均可通过其内建的远程策略接口，向预架构的云端发送请求，获得对应于该特定事件行为的相应的处理策略，并用于后续的处理。

[0244] 需要指出的是，有关以上三种获取处理策略的方式，可以交叉配合使用，例如，一旦交互模块接收到监控单元 34 传递的事件消息的特征，即可依照默认设置，参照第 (2) 种方式先行检索本地策略数据库，获得系统推荐的处理策略（如果不能从本地策略数据库中获得，甚至可以进一步按第 (3) 种方式从云端策略数据库中获取）。继而，参照第 (1) 种方式，在弹窗界面设置系统推荐的处理策略为默认选项。如果用户未在规定时限内确认该默认选项，则以系统推荐的处理策略为准执行后续指令；如果用户将之改变为新的默认选项，则向监控单元 34 返回用户设置的处理策略。可见，人机交互过程是可以更为灵活自由地实

现的。

[0245] 所述的本地策略数据库，可以是云端策略数据库的一个复件，因此，本发明中，设置一个更新步骤，用于下载云端策略数据库用于更新本地策略数据库。

[0246] 一般情况下，针对特定事件行为的策略可以设置为“拒绝”、“运行”、“询问”三个常见选项，其表征的具体意向为：

[0247] 拒绝：针对该特定事件行为，向目标应用 35 发送事件行为已经执行完毕的虚假消息，以禁止该事件行为实际发生；

[0248] 运行：针对该特定事件行为不做任何改变，将相应的事件消息直接转送给系统消息机制，允许目标应用 35 继续其事件行为；

[0249] 询问：独立或依附于前述两个选项任意之一，针对该特定事件行为，标记其状态为未知状态，后续重复发生该行为时，需要再行弹窗询问用户。

[0250] 实际应用中，选项“询问”可被忽略，仅需考虑是否拒绝或允许当前事件行为发生即可。

[0251] 所述的事件行为，多种多样，具体包括如下几大类型：

[0252] (1) 终端、联网有关的操作：

[0253] 获取运营商信息：目标应用 35 例如通过 getSimOperatorName() 函数可以获得移动终端的 IMSI，由此可进一步判断运营商的名称，进一步可以向运营商发送约定指令，实现扣费之类的非法目的。监控平台通过挂钩与此相关的消息，便可以对事件行为的捕获。

[0254] 切换 APN 操作：同理，目标应用 35 通过与 APN 切换有关的函数实现 ANP 切换控制的操作，也可被监控单元 34 通过调用相应的挂钩插件进行监控。

[0255] 类似的操作，还包括获取手机识别码 IME 的操作，也与上述同理。

[0256] (2) 通知栏广告操作：通知栏广告是最易被恶意程序利用的手段，监控单元 34 通过调用相应的挂钩插件对 notify 函数产生的事件消息进行监控，也可对其实施监控。

[0257] (3) 通信操作：

[0258] 如电话拨打操作，通过 StartActivity() 函数可以监控拨打电话的事件行为，利用相应的挂钩插件可以对拨打电话操作建立事件行为监控。

[0259] 短信操作，对应于 SendTextMessage() 之类的函数，同理，可以借助挂钩插件对这类函数建立事件行为监控。

[0260] 联系人操作：一般对应于 Query()、Insert() 函数，监控单元 34 利用挂钩插件挂钩此类函数可以实现对此类事件行为的监控捕获。

[0261] (4) 命令操作：

[0262] 如 SU 提权操作或执行命令操作，均需用到 Execve() 函数，监控单元 34 通过监控此函数的返回消息，便可实现该类事件行为的监控。

[0263] (5) 界面及访问操作：

[0264] 如创造快捷方式的事件行为，则对应于 SentBroacast() 函数。同理，对于隐藏程序图标的操作，也可对应特定函数监控之。

[0265] 如 HTTP 网络访问操作，则对应于 Sentto()、Write() 等函数。

[0266] (6) 程序操作：

[0267] 如应用加载操作，指当前目标应用 35 加载相关应用的操作，通过对

dexclassloader()、loadlibrary() 等函数进行挂钩监控，可以实现对此类事件行为的捕获。

[0268] 又如安装子包，则对应于 StartActivity() 函数。

[0269] (7) 其它危险操作 ：

[0270] 例如，子进程侵入操作、衍生物操作、激活设备管理器操作等，分别对应于。

[0271] 其中，子进程是指目标应用 35 建立的子进程，在目标应用 35 创建子进程时，监控单元 34 将收到相应的消息，而判定其创建子进程的事件行为。由此，监控单元 34 进一步向该子进程以内联钩子的方式在该子进程中植入监控单元 34，后续便可继续对该子进程的事件行为进行监控。因而，无论是目标应用 35 的自身进程，还是其创建的子进程，它们直接或间接所触发的事件行为，均能被本发明的监控单元 34 所监控，使主动防御效果更佳。

[0272] 而所述衍生物，是指目标应用 35 自行创建的文件，或者远程下载的文件，通常是指敏感的衍生物，例如安装包。通过挂钩 fclose() 函数可以捕获该事件。需要指出的是，当监控单元 34 捕获该事件行为后，可以按照前述的方法，进一步利用远程规则库接口发送请求到云端，由云端利用其黑、白、灰的安全等级行为规则判断该衍生物的安全等级，本发明通过远程规则库接口获得云端判定结果后，进一步弹窗询问用户是否建立对该敏感衍生物的主动防御，由此便可进一步巩固主动防御的效果。

[0273] 上述的事件行为仅为摘录之用，不能理解为对本发明监控的事件行为的限制。

[0274] 依据上述的处理策略和上述关于事件行为的说明，本发明的主动防御方法便可对各种事件行为进行相应的处理。以下列举几种典型的应用实例 ：

[0275] (1) 对目标应用 35 的精细拦截的应用 ：

[0276] 部分恶意程序被安装后，在相当长的一段时间内处于正常使用的状态，麻痹用户的安全意识。但是，运行一段长时间之后，该目标应用 35 尝试从后台插入一短信引起用户的关注，达到广告和诈骗的效果。对该目标应用 35 建立主动防御机制后，本发明如前所述，通过监控单元 34 中相应的挂钩插件对短信操作函数的监控，一旦目标应用 35 产生短信操作的事件行为，便可捕获这一事件行为，继而，监控单元 34 通过其交互接口通知作为系统服务运行的交互模块，由交互模块向用户界面弹窗示警。用户点选“拒绝”的处理策略后，被逆反馈给监控单元 34，其中相应的挂钩插件便能阻该事件行为的实际发生，达到防范风险的目的。

[0277] (2) 对目标应用 35 释放恶意文件的应用。

[0278] 目标应用 35 为一游戏软件，通过检查更新的方式下载并释放恶意子包，并且调用系统功能安装该子包。本发明对该目标应用 35 建立了主动防御之后，可以监控到其下载完文件而产生的事件行为，据此通过交互模块弹窗告警。用户指令拒绝之后，监控单元 34 中相应的挂钩插件便可直接删除该文件，或者仅仅拒绝该文件的安装行为。

[0279] 本发明中，对于诸如此类的恶意子包，视为敏感衍生物，对衍生物是否存在恶意的判断，参照前述防御配置方法中所述及的确定安全等级的方式进行远程判断。具体而言，当检测到产生衍生物时，将相应的文件或者其签名之类的特征信息通过远程规则库接口发送给云端，并从云端获得其安全等级，如果为黑、灰应用，则在弹窗中建议用户拒绝安装 ；如果为白应用，则可允许其通行。通过这种方法，便可实现对敏感衍生物的安全防御。如果云端检测不到该衍生物的相关记录，可以要求本方法为其上传该文件，并由云端标示为未知应

用，相应的，以灰应用予以标记，以备后用。

[0280] (3) 对子进程侵入的应用。

[0281] 被监控的目标应用 35 在运行过程中创建子进程，而子进程进一步释放恶意事件行为。监控单元 34 监控到目标应用 35 创建子进程时，即获得子进程的入口，然后向该子进程植入本发明的监控单元 34，所有 HOOK 插件（挂钩插件）都会被以内联钩子的方式加载到该子进程中并初始化好实现挂钩，以便建立对该子进程的事件行为的监控。由此，可以看出，无论是由目标应用 35 进程直接触发的事件行为，还是由目标应用 35 进程所创建的子进程所触发的间接事件行为，均能被监控单元 34 成功监控。

[0282] 由上述的分析可见，本发明的主动防御装置，相应于主动防御方法，具有高效的可行性。

[0283] 为便于本领域技术人员进一步实现本发明，以下进一步揭示云端服务器与终端设备如何相互配合实现安装包安全等级判断的相关内容：

[0284] 如前所述，由客户端通过远程规则库接口发送到云端服务器的特征信息，包括：Android 安装包的包名，和 / 或，版本号，和 / 或，数字签名，和 / 或，Android 组件 receiver 的特征，和 / 或，Android 组件 service 的特征，和 / 或，Android 组件 activity 的特征，和 / 或，可执行文件中的指令或字符串，和 / 或，Android 安装包目录下各文件的 MD5 值（签名）。

[0285] 实现了本发明的方法或装置的客户端，将指定的特征信息上传到服务器（云端），在服务器预置的规则库中查找与指定的单个特征信息或其组合相匹配的特征记录；其中，所述服务器预置的规则库中包含特征记录及特征记录对应的安全级别，每条特征记录中包含单个特征信息或特征信息的组合；

[0286] 服务器端规则库中预置了数千条特征记录，其中，第一条特征记录中列出了某种病毒的 Android 安装包包名，第二条特征记录中列出了某个正常应用的 Android 安装包版本号及其数字签名的 MD5 值，第三条特征记录中列出了某个正常应用的 Android 安装包包名及其 receiver 特征，第四条特征记录中列出了某种木马的 Android 安装包包名、版本号及其 ELF 文件中的特定字符串，等等。

[0287] 关于安全等级的标识，即黑，白（安全）或者灰（未知，可疑）三种标识，可以进一步的表示为：

[0288] 安全：该应用是一个正常的应用，没有任何威胁用户手机安全的行为；

[0289] 危险：该应用存在安全风险，有可能该应用本身就是恶意软件；也有可能该应用本来是正规公司发布的正常软件，但是因为存在安全漏洞，导致用户的隐私、手机安全受到威胁；

[0290] 谨慎：该应用是一个正常的应用，但是存在一些问题，例如会让用户不小心被扣费，或者有不友好的广告遭到投诉等；当发现这类应用之后，会提示用户谨慎使用并告知该应用可能的行为，但是由用户自行决定是否清除该应用；

[0291] 木马：该应用是病毒、木马或者其他恶意软件，此处为了简单统称为木马，但并不表示该应用仅仅是木马。

[0292] 应当理解，云端与客户端之间的配合，可以由本领域技术人员根据本发明所揭示的内容进一步扩充、变换、增删而改善。因而，以上揭示的内容不应理解为实现本发明的方

法和装置的限制。

[0293] 经过测试，本发明相对于现有技术有了较宽广的应用范围和应用效果，以下略加阐述：

[0294] 由于本发明已经将 HOOK 框架做成了服务平台，以挂钩插件的方式为终端配置监控单元 34，因此，其加载仅需依赖于相应的配置文件，管理高效且易于实现，对技术人员而言，一些简单的函数调用仅需编写配置文件即可实现挂钩插件的配置，HOOK 重入、并发性能高。

[0295] 采用外壳应用先后实现对监控单元 34 和目标应用 35 的加载，继而借助监控单元 34 对目标应用 35 的事件行为建立监控，可以实现对 Java 函数、Native 函数的挂钩。

[0296] 本发明不仅适用于 Dalvik 模式，也适用于 ART 模式，功能表现上两者无异，使用者不需适应不同模式编写不同的代码，简化开发工作（小范围内测试 Android 版本号 4.4.2、4.4.3、4.4.4）。

[0297] 经实测，有如下数据佐证本发明的实例的优越性：

[0298] (1) 本发明的开发实例，在 16 部手机上对 107 款主流应用软件（如 QQ、微信，微博，手机卫士，支付类、多种团购 app，各视频播放软件等）进行了稳定性深度测试，均能正常运行。

[0299] (2) 本发明的开发实例，测试涵盖手机 Android 操作系统版本号从 2.3 到 4.4.3。机型包括 nexus4/5、7，三星，小米，华为，联想，索尼，HTC 及部分山寨手机，均获得较为优异的表现。

[0300] (3) 本发明的开发实例，支持加固应用，如支持 360 加固、网秦加固，腾讯加固、梆梆和爱加密、APKProtect 等，对于以上各家厂商提供的加固应用的测试显示，本发明的实例均可正常运行。

[0301] (4) 本发明的开发实例的测试效果显示，手机端生成外壳包的成功率为 99.7％（基数为 100W）。

[0302] 综上所述，本发明所提供的主动防御技术更为安全高效。

[0303] 以上所述仅是本发明的部分实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本发明原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也应视为本发明的保护范围。

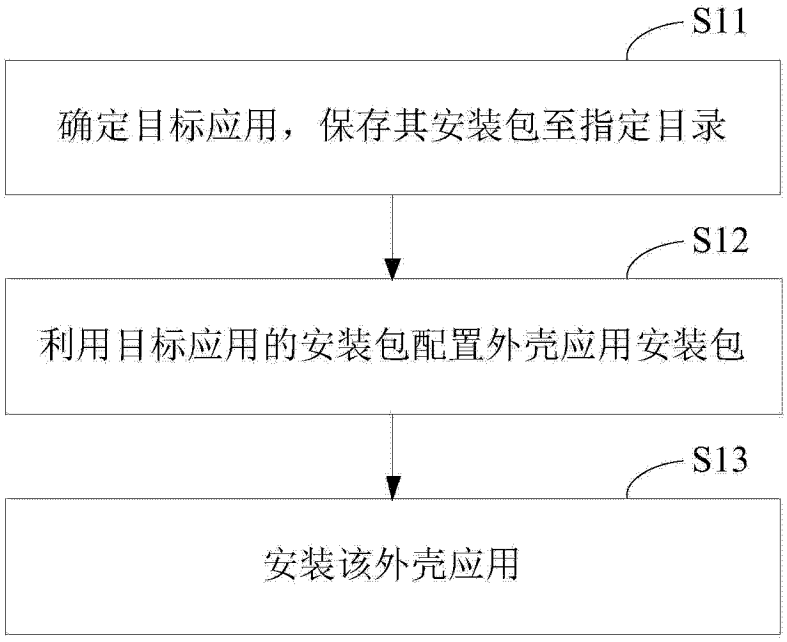

S11
确定目标应用，保存其安装包至指定目录
S12
利用目标应用的安装包配置外壳应用安装包
S13
安装该外壳应用

图 1

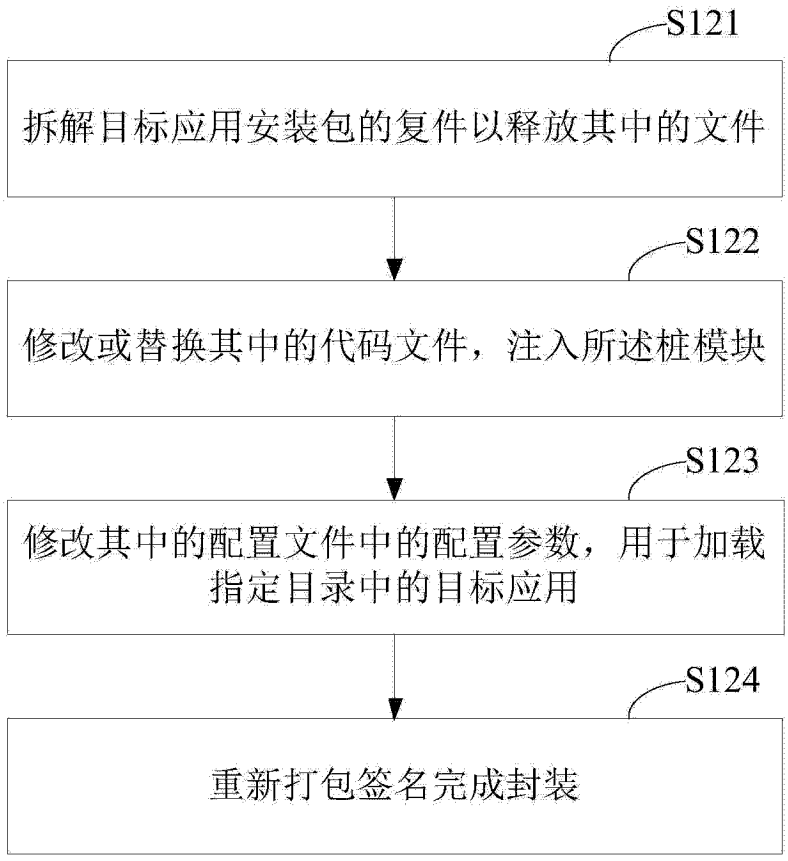

S121
拆解目标应用安装包的复件以释放其中的文件
S122
修改或替换其中的代码文件，注入所述桩模块
S123
修改其中的配置文件中的配置参数，用于加载指定目录中的目标应用
S124
重新打包签名完成封装

图 2

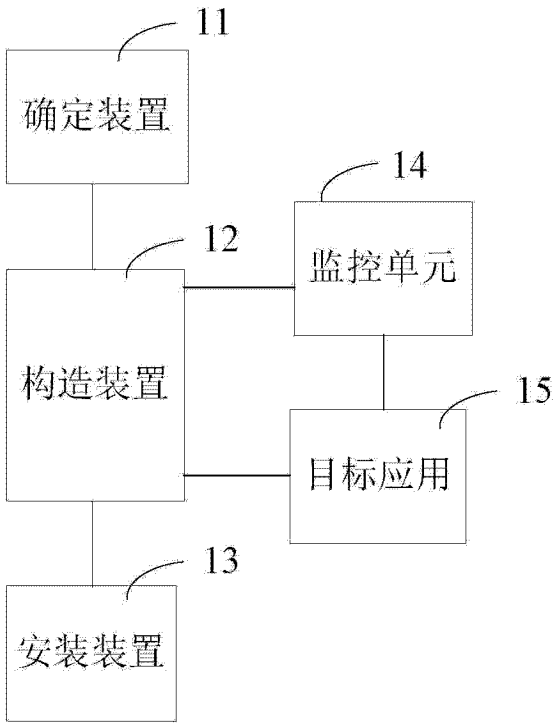

11
确定装置
14
监控单元
12
构造装置
15
目标应用
13
安装装置

图 3

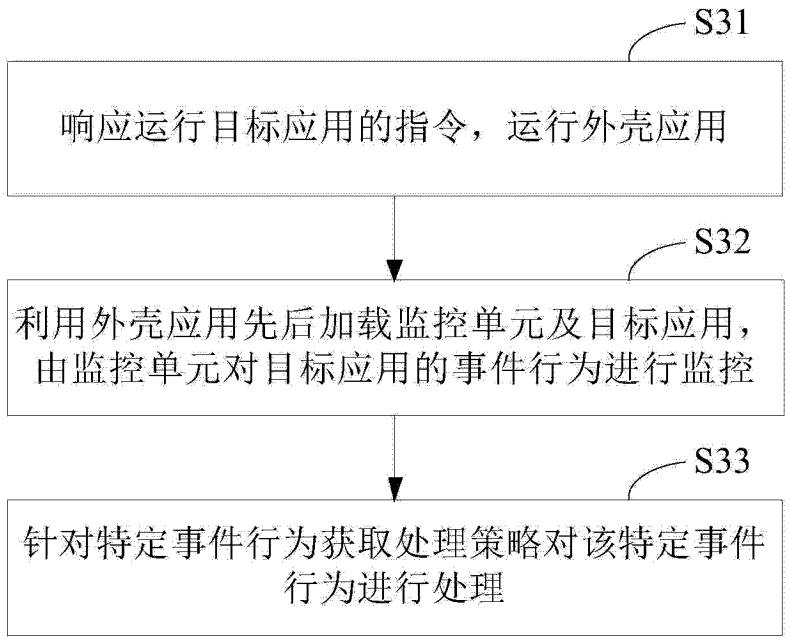

S31
响应运行目标应用的指令，运行外壳应用
S32
利用外壳应用先后加载监控单元及目标应用，由监控单元对目标应用的事件行为进行监控
S33
针对特定事件行为获取处理策略对该特定事件行为进行处理

图 4

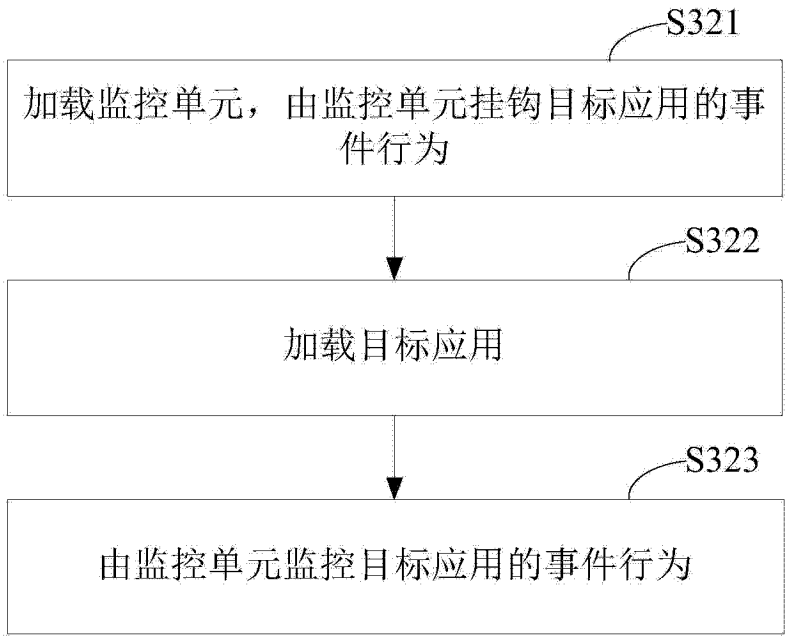

S321
加载监控单元，由监控单元挂钩目标应用的事件行为
S322
加载目标应用
S323
由监控单元监控目标应用的事件行为

图 5

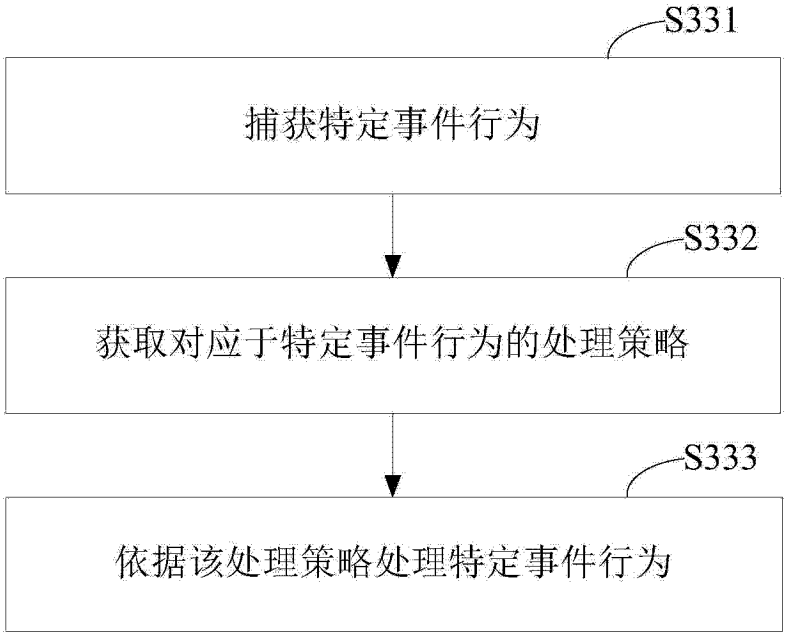

S331
捕获特定事件行为
S332
获取对应于特定事件行为的处理策略
S333
依据该处理策略处理特定事件行为

图 6

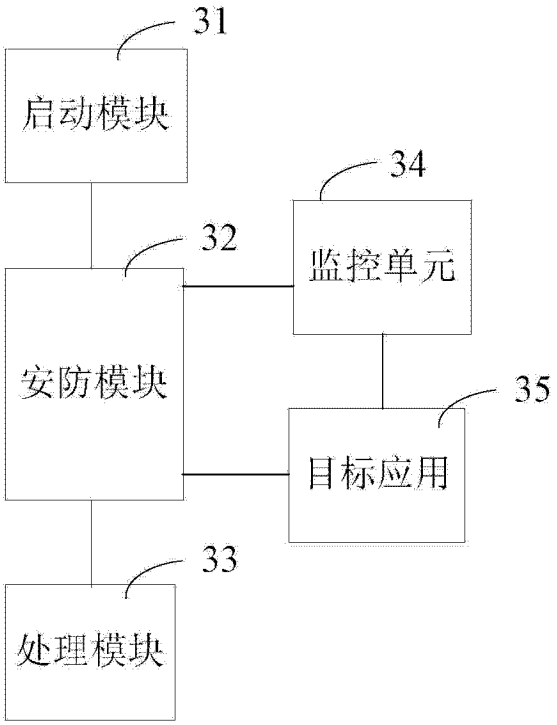

31
启动模块
34
监控单元
32
安防模块
35
目标应用
33
处理模块

图 7


手机正在快速充电
360护盾提示
是否对以下新安装应用进行监控
取消
监控

图 8


360手机护盾
execTes
点击监控
跳跳虎
已监控
移动掌上客户端
已监控
Receipts
点击监控
Q版泡泡堂攻略
已监控
360手机助手
点击监控
微博
已监控
ENERGY.DE 2.0
点击监控
手机淘宝
点击监控
豌豆荚
点击监控

图 9


6:08
360手机护盾
com.sina.weibo
微博
4.5.5
权限管理
行为管理
发送文本短信
询问 允许 拒绝
发送文本信息
询问 允许 拒绝
发送数据信息
询问 允许 拒绝
启动设备管理器
询问 允许 拒绝
拨打电话
询问 允许 拒绝
安装应用
询问 允许 拒绝
录音
询问 允许 拒绝
加载so文件
询问 允许 拒绝
加载dex文件
询问 允许 拒绝
重启应用
询问 允许 拒绝
创建快捷方式
询问 允许 拒绝
删除快捷方式
询问 允许 拒绝

图 10


5:24
移动掌上客户端
360护盾提示
移动掌上客户端
正在进行以下操作，是否允许？
发送短信：
号码：15064128842
内容：软件已安装。
当前手机型号Nexus 4
软件到期时间:2014-06-30
记住我的选择
拒绝(30s)
允许

图 11


5:32
测试内容显示区域
请输入电话号码
360护盾提示
ApiTest
正在进行以下操作，是否允许？
插入短信
号码：15810531959
内容：同程旅游是中国领先的一站式旅游预订平台，提供20000多家特惠酒店、特价机票、景点门票、出国旅游、邮轮旅行、自助游预订服务。详情请致电1101301581417
记住我的选择
拒绝
允许(25s)
短信
通讯..
通话..
书签
其它..
地理..
APN..
安装..
快捷..
通知..

图 12