(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 108280343 A
(43)申请公布日 2018.07.13

(21)申请号 201710012122.6

(22)申请日 2017.01.06

(71)申请人 广州市动景计算机科技有限公司
地址 510627 广东省广州市天河区黄埔大道西平云路163号广电平云广场B塔14层

(72)发明人 方爱强 骆智彬 卢声康

(74)专利代理机构 北京展翼知识产权代理事务所(特殊普通合伙) 11452
代理人 张阳

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

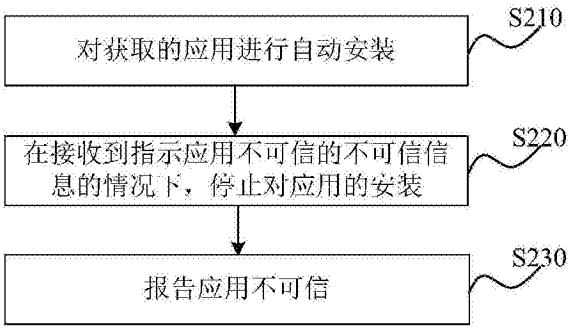
权利要求书2页 说明书6页 附图3页

(54)发明名称

安卓环境下检测应用安全性的方法、装置及系统

(57)摘要

本发明公开了一种安卓环境下检测应用安全性的方法、装置及系统。方法包括:对获取的应用进行自动安装;在接收到指示应用不可信的不可信信息的情况下,停止对应用的安装;以及报告应用不可信,其中,不可信信息是安全服务器响应于应用在安装时自动上报的验证信息没有通过验证而发出的。由此,利用本发明可以自动完成应用的整个检测过程,并可以将检测为不可信应用的不可信信息报告给用户、检测人员或服务器。



CN 108280343 A

1.一种安卓环境下检测应用安全性的方法,包括:

对获取的应用进行自动安装;

在接收到指示所述应用不可信的不可信信息的情况下,停止对所述应用的安装;以及

报告所述应用不可信,

其中,所述不可信信息是安全服务器响应于所述应用在安装时自动上报的验证信息没有通过验证而发出的。

2.如权利要求1所述的方法,还包括:

在没有接收到指示所述应用不可信的不可信信息使得应用安装完成的情况下,对完成安装的所述应用进行卸载。

3.如权利要求1所述的方法,其中,对获取的应用进行自动安装包括:

获取Accessibility权限读取屏幕中应用的安装键并进行自动点击安装。

4.如权利要求1所述的方法,其中,报告所述应用不可信包括如下至少一项:

向从其下载所述应用的应用下载服务器报告所述应用不可信;

向所述安卓环境的用户报告所述应用不可信;以及

报告所述应用不可信以及所述不可信类型。

5.如权利要求1所述的方法,还包括:

删除被指示为不可信的所述应用的安装文件。

6.一种安卓环境下检测应用安全性的方法,包括:

对获取的多个应用逐一进行自动安装;

在接收到指示当前应用不可信的不可信信息的情况下,停止对所述当前应用的安装;

报告所述当前应用不可信;以及

开始对下一应用的安全性检测直到多个应用都完成检测为止,

其中,所述不可信信息是安全服务器响应于所述当前应用在安装时自动上报的验证信息没有通过验证而发出的。

7.一种安卓环境下检测应用安全性的装置,包括:

应用安装单元,用于对获取的应用进行自动安装;

安装停止单元,用于在接收到指示所述应用不可信的不可信信息的情况下,停止对所述应用的安装;以及

报告单元,用于报告所述应用不可信,

其中,所述不可信信息是安全服务器响应于所述应用在安装时自动上报的验证信息没有通过验证而发出的。

8.如权利要求7所述的装置,还包括:

应用卸载单元,用于在没有接收到指示所述应用不可信的不可信信息使得应用安装完成的情况下,对完成安装的所述应用进行卸载。

9.如权利要求7所述的装置,还包括:

安装文件删除单元,用于删除被指示为不可信的所述应用的安装文件。

10.一种检测安卓环境下应用安全性的系统,包括应用下载服务器和客户端,所述应用下载服务器用于:

向所述客户端下发应用安装文件;

获取所述客户端报告的所述应用是否可信的日志;以及

根据所述日志对应用安装文件进行处理,

所述客户端用于:

对由所述应用下载服务器下发的所述应用安装文件进行自动安装;

在接收到指示所述应用不可信的不可信信息的情况下,停止对所述应用的安装;以及

向所述应用下载服务器报告所述应用不可信,

其中,所述不可信信息是安全服务器响应于所述应用在安装时自动上报的验证信息没有通过验证而发出的。

安卓环境下检测应用安全性的方法、装置及系统

技术领域

[0001] 本发明涉及应用检测领域，特别是涉及一种安卓环境下检测应用安全性的方法、装置及系统。

背景技术

[0002] 安卓系统是开放源代码的操作系统，开发者便于在此基础上开发安卓系统的应用，因此安卓系统一经推出便发展得极为迅速，各种安卓应用也层出不穷。由于安卓系统的开放性，用户不仅能从谷歌官方市场下载和安装应用程序，也可以从其它第三方市场甚至网站和论坛下载和安装应用程序。

[0003] 多渠道下载应用的方式为用户提供了便利性，但是同时也带来了安全性问题。例如部分不法分子利用安卓系统的开放性开发出带有病毒或者木马程序的恶意应用，并将这些恶意应用安装程序上传到多种下载渠道，给用户造成潜在的危害。

[0004] 由此，需要一种能够对安卓环境下应用的安全性进行检测的方案。

发明内容

[0005] 本发明主要解决的技术问题是提供一种安卓环境下检测应用安全性的方法、装置及系统，其能够快速有效地检测出应用是否安全。

[0006] 根据本发明的一个方面，提出了一种安卓环境下检测应用安全性的方法，包括：对获取的应用进行自动安装；在接收到指示应用不可信的不可信信息的情况下，停止对应用的安装；以及报告应用不可信，其中，不可信信息是安全服务器响应于应用在安装时自动上报的验证信息而发出的。

[0007] 由此，利用本发明可以自动完成应用的整个检测过程，并可以将检测为不可信应用的不可信信息报告给用户、检测人员或相应的应用下载服务器等。

[0008] 优选地，该方法还可以包括：在没有接收到指示应用不可信的不可信信息的情况下，对完成安装的应用进行卸载。

[0009] 由此，本发明可以用于专门用于测试的客户端设备，如果测试中没有接收到指示其不可信的不可信信息，表明该应用安全，可以在客户端设备上正常安装，并且在安装完成后可以对其进行卸载，以释放占用空间。本发明也可以用于用户正常使用的客户端，以便以对用户透明的方式实现对应用安装文件的安全性检测。

[0010] 优选地，对获取的应用进行自动安装可以包括：获取Accessibility权限读取屏幕中应用的安装键并进行自动点击安装。

[0011] 由此，可以利用安卓系统所提供的Accessibility(无障碍)功能来实现自动点击安装按钮，并调用包安装器(PackageInstaller)实现应用的自动安装。

[0012] 优选地，报告应用不可信可以包括如下至少一项：向从其下载应用的应用下载服务器报告应用不可信；向安卓环境的用户报告应用不可信；以及报告应用不可信以及不可信类型。

[0013] 由此，可以将检测出的应用的不可信信息通知给下载了该应用的安卓用户，也可以将应用的不可信信息上报给应用下载服务器，便于应用下载服务器对不可信应用执行限制下载、下架等处理。其中，在将不可信信息报告给用户或应用下载服务器时，可以将不可信类型一并告知。

[0014] 优选地，该方法还可以包括：删除被指示为不可信的应用的安装文件。由此也可以释放占用空间。

[0015] 根据本发明的另一个方面，还提供了一种安卓环境下检测应用安全性的方法，包括：对获取的多个应用逐一进行自动安装；在接收到指示当前应用不可信的不可信信息的情况下，停止对当前应用的安装；报告当前应用不可信；以及开始对下一应用的安全性检测直到多个应用都完成检测为止，其中，不可信信息是安全服务器响应于当前应用在安装时自动上报的验证信息而发出的。由此，利用本发明可以对多个应用的安全性进行检测。

[0016] 根据本发明的又一个方面，还提供了一种安卓环境下检测应用安全性的装置，包括：应用安装单元，用于对获取的应用进行自动安装；安装停止单元，用于在接收到指示应用不可信的不可信信息的情况下，停止对应用的安装；以及报告单元，用于报告应用不可信，其中，不可信信息是安全服务器响应于应用在安装时自动上报的验证信息没有通过验证而发出的。

[0017] 优选地，该装置还可以包括：应用卸载单元，用于在没有接收到指示应用不可信的不可信信息的情况下，对完成安装的应用进行卸载。

[0018] 优选地，该装置还可以包括：安装文件删除单元，用于删除被指示为不可信的应用的安装文件。

[0019] 根据本发明的再一个方面，还提供了一种检测安卓环境下应用安全性的系统，包括应用下载服务器和客户端，应用下载服务器用于：向客户端下发应用安装文件；获取客户端报告的应用是否可信的日志；以及根据日志对应用安装文件进行处理，客户端用于：对由应用下载服务器下发的应用安装文件进行自动安装；在接收到指示应用不可信的不可信信息的情况下，停止对应用的安装；以及向应用下载服务器报告应用不可信，其中，不可信信息是安全服务器响应于应用在安装时自动上报的验证信息没有通过验证而发出的。

[0020] 利用本发明的安卓环境下检测应用安全性的方法、装置及系统，可以自动实现对应用的安全性检测，并可以将检测为不可信的应用的不可信信息报告给用户、检测人员或服务器。

附图说明

[0021] 通过结合附图对本公开示例性实施方式进行更详细的描述，本公开的上述以及其它目的、特征和优势将变得更加明显，其中，在本公开示例性实施方式中，相同的参考标号通常代表相同部件。

[0022] 图1是示出了根据本发明一实施例的检测安卓环境下应用安全性的系统的功能框图。

[0023] 图2是示出了根据本发明一实施例的检测安卓环境下应用安全性的方法的示意性流程图。

[0024] 图3是示出了根据本发明一实施例的检测安卓环境下应用安全性的装置的功能框

图。

[0025] 图4是示出了利用本发明实现监控的一个具体方案的流程图。

具体实施方式

[0026] 下面将参照附图更详细地描述本公开的优选实施方式。虽然附图中显示了本公开的优选实施方式，然而应该理解，可以以各种形式实现本公开而不应被这里阐述的实施方式所限制。相反，提供这些实施方式是为了使本公开更加透彻和完整，并且能够将本公开的范围完整地传达给本领域的技术人员。

[0027] 为了实现对安卓应用的安全性的检测，本发明基于安卓系统的特性，提供了一种可以自动对应用的安全性进行检测的方案。利用本发明可以自动完成应用的整个检测过程，并可以将检测为不可信应用的不可信信息上报。

[0028] 下面将参照图1至图3来具体地描述本发明的实施例。图1是示出了根据本发明一实施例的检测安卓环境下应用安全性的系统的功能框图。

[0029] 如图1所示，检测系统100包括应用下载服务器110和一个或多个客户端120。客户端120可以是专门用于检测应用是否安全的安卓设备，也可以是用户使用的安卓设备。

[0030] 客户端120可以向应用下载服务器110发送安装文件获取请求，响应于客户端120的获取请求，应用下载服务器110可以向客户端120下发相应的应用安装文件。其中，图1示出的是一个应用下载服务器110，应该知道，客户端120还可以从多个不同的应用下载服务器110获取应用安装文件。

[0031] 客户端120在获取了应用安装文件后，就可以执行检测应用安全性的处理(例如，通过上报安全服务器)，以确定应用安装文件是否可信，并可以向应用下载服务器110上报关于应用安装文件是否可信的日志，应用下载服务器110可以根据客户端上报的日志对应用安装文件执行相应的处理。例如，在应用下载服务器110从客户端120获取的日志指示应用不可信时，就可以对应用安装文件执行限制下载、下架等处理。

[0032] 下面详细说明客户端120执行检测应用安全性的处理的具体流程，该流程可以参见图2。图2是示出了根据本发明一实施例的检测应用安全性的方法的示意性流程图。

[0033] 如图2所示，检测应用安全性处理开始于步骤S210，客户端120对获取的应用进行自动安装。由于客户端120的操作系统是安卓系统，因此，可以通过在安卓环境下获取Accessibility(Android无障碍功能，借助该功能可以实现自动点击等操作)权限来实现自动安装应用。具体地，可以利用Accessibility权限读取屏幕中应用的安装键并进行自动点击，以唤起包安装程序(PackageInstaller，Android默认的应用程序，用它来交互式地安装普通包文件)对应用进行安装。

[0034] 对于安卓系统而言，在应用的安装过程中，还可以向安全服务器自动上报验证信息，安全服务器根据验证信息可以判断应用是否可信，在判定不可信的情况下，安全服务器会向客户端发送用于指示应用不可信的信息(为了便于描述，称为不可信信息)甚至可以直接拦截不可信应用的安装。

[0035] 在安卓环境下，本文中的安全服务器优选地是由Google(谷歌公司)设置地、用于提供在线应用服务的服务器。具体来说，自2012年起，Google在安卓系统中新增了应用验证(Verify Apps)功能，应用验证功能可以定期检查设备上的活动，一旦发现恶意应用(PHA，

Potentially Harmful Applications,Google对于恶意应用的定义),就会加以阻止或发出警告。因此可以通过Google Play服务提供的应用验证功能向对应的安全服务器发送验证信息,以验证新安装的应用是否有害。Google Play服务可以用于更新Google应用和Google Play提供的其他应用,此组件可提供多种核心功能,例如对支持Verify Apps功能,实现对PHA的拦截或警告等。验证信息可以包括包名、版本号、md5等等用于标识应用身份的信息。对于被Google的安全服务器确定为恶意的应用,安全服务器可以将指示该应用为恶意应用的信息以及该恶意应用的类型(即PHA的类型)一并下发,甚至能够对客户端上正安装的应用进行拦截。由此在安卓环境的客户端上安装应用时,可以利用应用的自动验证功能向安全服务器发送验证信息,安全服务器可以根据验证信息验证应用是否可信。客户端可以监听来自应用服务器的用于指示应用是否有害及具体有害类型的信息。

[0036] 对于步骤S210来说,也可以基于ROOT的自动安装检测方式来完成应用的自动安装、卸载等操作,但是Google对ROOT设备的安全策略有所特殊化,无法保证检测恶意应用的有效性。

[0037] 另外,也可以基于前台进程监听的方式来获取安全服务器下发的指示应用是否有害的信息。但是这种方式无法在设备层面完成自主地自动安装、卸载的需求,且仅仅能够获取应用是否被拦截,无法获取具体的拦截信息。也就是说仅能够判定应用是否为恶意应用,但是无法判定属于哪一类的恶意应用。

[0038] 因此基于上述考虑,本发明优选地利用Accessibility权限来实现应用的自动安装以及监听来自安全服务器下发的用于指示应用是否可信的信息。

[0039] 在接收到来自安全服务器的不可信信息时,就可以停止对应用的安装(步骤S220),并可以报告应用不可信(步骤S230)。

[0040] 在实际操作中,可以是安全服务器在检测到应用不可信的情况下,以拦截相应应用在客户端上的安装的拦截指令的形式下发不可信信息。例如,安全服务器可以向客户端发出拦截安装的指令,并相应地在客户端的显示界面上弹出指示应用不可信的警告窗口。客户端响应于来自安全服务器的拦截,停止对目标应用的安装。由于客户端120是自行启动对应用安装文件安全性的检测,因此客户端120优选地拦截上述警告窗口在客户端显示界面上的显示,以便保证整个安全性检测过程以用户透明的方式进行。

[0041] 在确认安装文件不可行的情况下,客户端120可以向下载该应用的应用下载服务器110报告应用不可信,也可以向用户或检测人员报告应用不可信,并且在报告应用不可信时还可以将不可信的类型一并告知。

[0042] 具体地,在向用户报告应用不可信时,可以向安卓环境下的广大用户报告应用不可信。以用下载服务器为华为应用市场下载服务器为例,在执行步骤S210、步骤S220后发现应用市场下某个应用软件不可信时,可以在应用市场下该应用的下方进行标记(例如叉形或是其他有害或疑似有害的标记)以告知用户该应用为不可信应用。在向应用下载服务器110报告应用不可信时,可以向应用下载服务器110上报应用不可信的日志,应用下载服务器110可以根据指示应用不可信的日志对应用安装文件进行下架、取消下载权限等处理。

[0043] 在没有接收到指示应用不可信的不可信信息的情况下,可以确定该应用安全,此时会在客户端上完成应用的正常安装。出于执行应用的安全性检测(而非真正安装)的考虑,优选在应用通过检测后,卸载已安装的应用。而在接收到不可信信息时,可以停止安装

并直接删除应用的安装文件。上述安装文件的删除可以可选地通知客户端的用户。

[0044] 客户端可以是用户设备，上述安全性检测过程需要以用户透明的方式进行。例如，对于通过安全服务器验证的应用安装文件，客户端可以在应用完成安装后进行自动卸载，并为此安装文件添加一个表示通过安全性验证的标记(比如，绿色勾形)。而对被判定为不可信的应用，客户端可以自动在后台拦截安全服务器(例如，谷歌应用服务)发出的警告并将由安全服务器拦截安装的应用安装文件删除。对于该应用安装文件有害的判定以及具体类型，可以报告客户端的用户、应用下载服务器或是其他相关方或检测人员。

[0045] 至此，结合图2对检测应用的安全性的过程做了详细说明。在利用本发明检测多个应用的安全性时，可以对获取的多个应用逐一执行检测应用安全性处理，在执行完第一个应用的检测后，再开始对下一个应用的安全性检测，直到多个应用都完成检测为止。

[0046] 图3是示出了根据本发明一实施例的安卓环境下检测应用安全性的装置的功能框图。其中，检测装置300的功能模块可以由实现本发明原理的硬件、软件或硬件和软件的结合来实现。本领域技术人员可以理解的是，图3所描述的功能模块可以组合起来或者划分成子模块，从而实现上述发明的原理。因此，本文的描述可以支持对本文描述的功能模块的任何可能的组合、或者划分、或者更进一步的限定。

[0047] 图3所示的检测装置300可以用来实现图2所示的监控方法，下面仅就检测装置300可以具有的功能模块以及各功能模块可以执行的操作做简要说明，对于其中涉及的细节部分可以参见上文结合图2的描述，这里不再赘述。

[0048] 如图3所示，检测装置300包括应用安装单元310、安装停止单元320以及报告单元330。

[0049] 安装单元310用于对获取的应用进行自动安装。安装停止单元320用于在接收到指示应用不可信的不可信信息的情况下，停止对应用的安装。报告单元330用于报告应用不可信。其中，不可信信息是安全服务器响应于应用在安装时自动上报的验证信息而发出的。

[0050] 可选地，检测装置300还可以包括应用卸载单元340。应用卸载单元340用于在没有接收到指示应用不可信的不可信信息的情况下，对完成安装的应用进行卸载。

[0051] 可选地，检测装置300还可以包括安装文件删除单元350，安装文件删除单元350用于删除被指示为不可信的应用的安装文件。

[0052] 上文中已经参考附图详细描述了根据本发明的安卓环境下检测应用安全性的方法、装置及系统。如下将给出利用本发明实现监控的一个具体应用。

[0053] 应用例

[0054] 图4是示出了利用本发明实现监控的一个具体方案的流程图。其中图4所示的前端可以是客户端，后端可以是服务端，例如前述的应用下载服务器。方案具体实现步骤如下所示。

[0055] 步骤1，任务准备阶段。后端可以获取待检测的应用程序，生成检测任务列表，通过特定的接口下发给前端。前端在联入网络状态下，可以从后端获取任务列表所对应的应用程序文件(应用安装文件)。

[0056] 步骤2，自动安装阶段。当前端下载应用安装程序完毕时，可以自动调起应用程序管理服务PackageInstaller对目标应用程序进行安装。具体地，前端可以利用Accessibility权限来自动识别设备屏幕上的“安装”按钮，以调用PackageInstaller实现

应用的自动安装过程。

[0057] 步骤3，拦截检测与识别。前端可以在应用安装过程中检测Google Play服务是否对该应用程序进行拦截，若是拦截则可以读取其具体的警告信息内容。

[0058] 步骤4。检测结果上报，对步骤3中的检测结果进行上报。后端获取上报信息之后进行解析、存储，以便实现后续处置。

[0059] 另外，对于通过Google Play服务的验证而顺利完成安装的应用，由于本发明是用于安全性检测的方案(换句话说，用户本人并没有真的点击“安装”按钮)，因此需要将顺利完成安装的应用进行卸载，并且可以优选地为通过验证的安装文件附上标记。

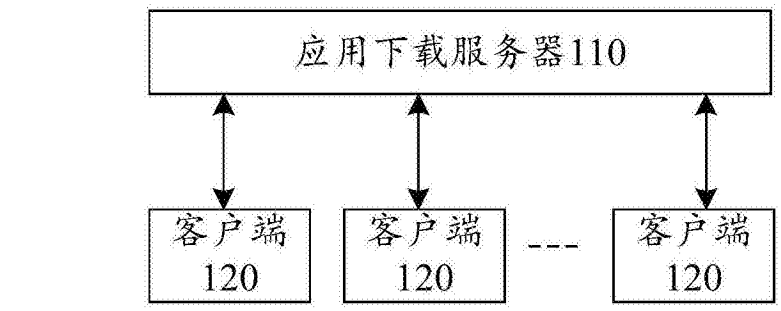
[0060] 在实际测试中，利用本发明可以有效识别特定应用安装包是否有害以及有害的具体类型。目前利用本发明的方案可以支持识别14类有害应用的拦截状态，单个客户端设备的日检验能力达1200～1400个/台。

[0061] 上文中已经参考附图详细描述了根据本发明的安卓环境下检测应用安全性的方法、装置及系统。

[0062] 此外，根据本发明的方法还可以实现为一种计算机程序，该计算机程序包括用于执行本发明的上述方法中限定的上述各步骤的计算机程序代码指令。或者，根据本发明的方法还可以实现为一种计算机程序产品，该计算机程序产品包括计算机可读介质，在该计算机可读介质上存储有用于执行本发明的上述方法中限定的上述功能的计算机程序。本领域技术人员还将明白的是，结合这里的公开所描述的各种示例性逻辑块、模块、电路和算法步骤可以被实现为电子硬件、计算机软件或两者的组合。

[0063] 附图中的流程图和框图显示了根据本发明的多个实施例的系统和方法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中，方框中所标记的功能也可以以不同于附图中所标记的顺序发生。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操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。

[0064] 以上已经描述了本发明的各实施例，上述说明是示例性的，并非穷尽性的，并且也不限于所披露的各实施例。在不偏离所说明的各实施例的范围和精神的情况下，对于本技术领域的普通技术人员来说许多修改和变更都是显而易见的。本文中所用术语的选择，旨在最好地解释各实施例的原理、实际应用或对市场中的技术的改进，或者使本技术领域的其它普通技术人员能理解本文披露的各实施例。


100 检测系统
应用下载服务器110
客户端
120
客户端
120

客户端
120

图1

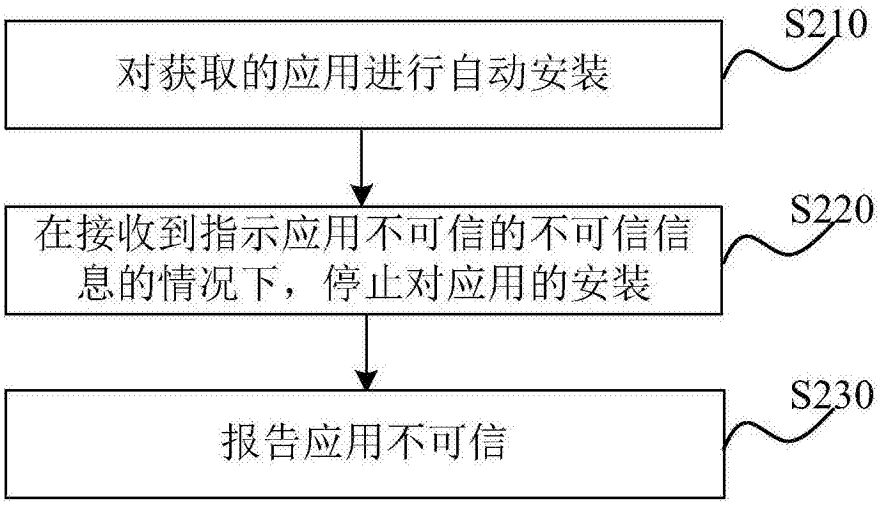

对获取的应用进行自动安装
S210
在接收到指示应用不可信的不可信信息的情况下，停止对应用的安装
S220
报告应用不可信
S230

图2

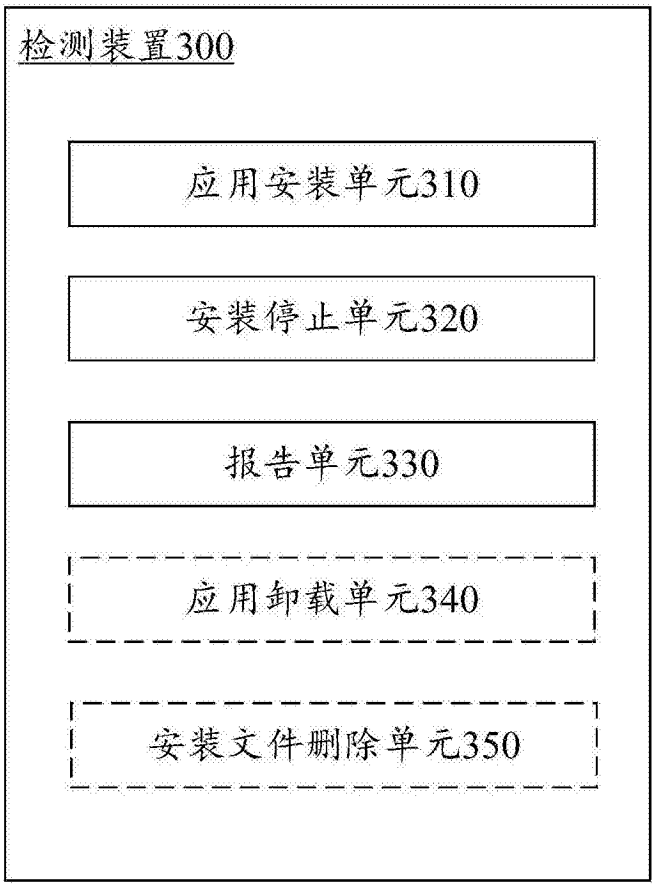

检测装置300
应用安装单元310
安装停止单元320
报告单元330
应用卸载单元340
安装文件删除单元350

图3

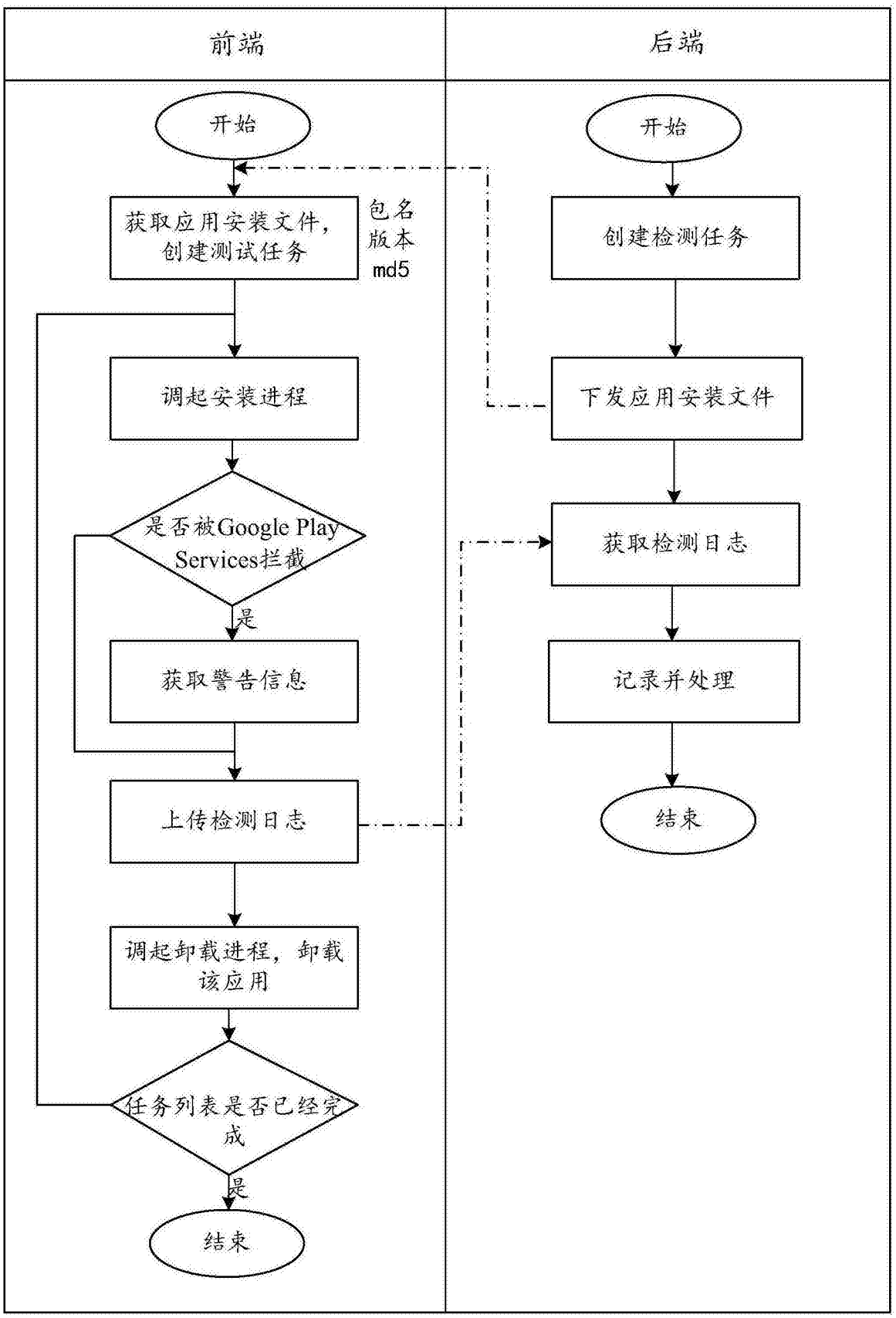

前端
后端
开始
开始
获取应用安装文件，创建测试任务
包名
版本
md5
创建检测任务
调起安装进程
下发应用安装文件
是否被Google Play Services拦截
获取检测日志
是
获取警告信息
记录并处理
上传检测日志
结束
调起卸载进程，卸载该应用
任务列表是否已经完成
是
结束

图4