(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 104391729 B
(45)授权公告日 2018.05.01

(21)申请号 201410806477.9

(22)申请日 2014.12.19

(65)同一申请的已公布的文献号
申请公布号 CN 104391729 A

(43)申请公布日 2015.03.04

(73)专利权人 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)
专利权人 奇智软件(北京)有限公司

(72)发明人 符传坚 陈俊 邹勇 马金亭

(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 8/65(2018.01)
G06F 21/57(2013.01)

(56)对比文件
CN 101958933 A,2011.01.26,
CN 101958933 A,2011.01.26,
CN 1818867 A,2006.08.16,
CN 102981835 A,2013.03.20,

审查员 张昕

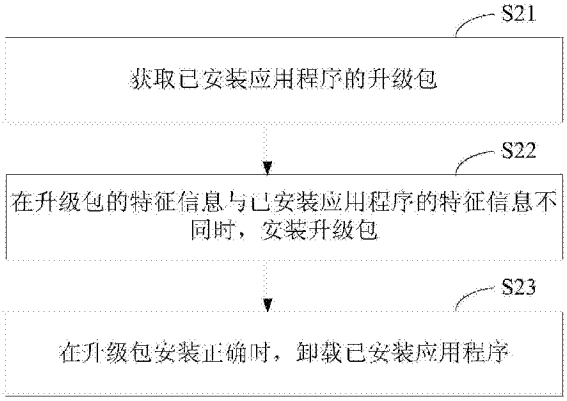
权利要求书2页 说明书9页 附图4页

(54)发明名称

基于Root权限的程序升级方法及装置

(57)摘要

本发明提供一种基于Root权限的程序升级方法,包括以下步骤:获取已安装应用程序的升级包;在升级包的特征信息与已安装应用程序的特征信息不同时,安装升级包;在升级包安装正确时,卸载已安装应用程序。本发明还提供一种基于Root权限的程序升级装置。通过上述方式,在具有Root权限时,可对应用程序进行跨包名和/或签名升级,以避免用户的流失。



CN 104391729 B

1.一种基于Root权限的程序升级方法，其特征在于，所述方法在客户端中执行，包括以下步骤：

获取本地已安装应用程序的升级包；

在所述升级包的特征信息与已安装应用程序的特征信息不同时，安装所述升级包；所述特征信息包括包名和签名；

在所述升级包安装正确时，卸载所述已安装应用程序。

2.根据权利要求1所述的基于Root权限的程序升级方法，其特征在于，在所述升级包的特征信息与已安装应用程序的特征信息不同时，安装所述升级包的步骤具体包括：

发送获取升级包特征信息的请求；

接收响应请求所返回的升级包特征信息；

判断所述升级包的特征信息与已安装应用程序的特征信息是否一致；

若不一致，则安装所述升级包。

3.根据权利要求2所述的基于Root权限的程序升级方法，其特征在于，在所述升级包安装正确时，卸载所述已安装应用程序的步骤具体包括：

所述升级包安装完成后，发送启动该应用的请求；

若安装后的升级包可启动，则所述升级包安装正确，卸载所述已安装应用程序。

4.根据权利要求3所述的基于Root权限的程序升级方法，其特征在于，所述特征信息还包括应用大小和应用中各文件的MD5值。

5.根据权利要求4所述的基于Root权限的程序升级方法，其特征在于，安装的升级包为合法的升级包。

6.根据权利要求5所述的基于Root权限的程序升级方法，其特征在于，接收响应请求所返回的升级包特征信息后，判断该特征信息与升级包当前特征信息是否一致，若一致，则获取的升级包合法，所述升级包当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。

7.根据权利要求6所述的基于Root权限的程序升级方法，其特征在于，在所述升级包的特征信息与已安装应用程序的特征信息相同时，对所述已安装应用程序进行普通升级。

8.根据权利要求7所述的基于Root权限的程序升级方法，其特征在于，应用程序的升级通过静默或用户触发的方式实现。

9.根据权利要求1所述的基于Root权限的程序升级方法，其特征在于，所述升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，所述数字标识为0，当升级前后的包名和/或签名不同时，所述数字标识为其他数字。

10.一种基于Root权限的程序升级装置，其特征在于，所述装置在客户端中执行，包括：

获取模块，用于获取本地已安装应用程序的升级包；

安装模块，用于在所述升级包的特征信息与已安装应用程序的特征信息不同时，安装所述升级包；所述特征信息包括包名和签名；

卸载模块，用于在所述升级包安装正确时，卸载所述已安装应用程序。

11.根据权利要求10所述的基于Root权限的程序升级装置，其特征在于，所述安装模块具体用于：

发送获取升级包特征信息的请求；

接收响应请求所返回的升级包特征信息；

判断所述升级包的特征信息与已安装应用程序的特征信息是否一致；

若不一致，则安装所述升级包。

12.根据权利要求11所述的基于Root权限的程序升级装置，其特征在于，所述卸载模块具体用于：

所述升级包安装完成后，发送启动该应用的请求；

若安装后的升级包可启动，则所述升级包安装正确，卸载所述已安装应用程序。

13.根据权利要求12所述的基于Root权限的程序升级装置，其特征在于，所述特征信息还包括应用大小和应用中各文件的MD5值。

14.根据权利要求13所述的基于Root权限的程序升级装置，其特征在于，安装的升级包为合法的升级包。

15.根据权利要求14所述的基于Root权限的程序升级装置，其特征在于，所述安装模块具体用于：

接收响应请求所返回的升级包特征信息后，判断该特征信息与升级包当前特征信息是否一致，若一致，则获取的升级包合法，所述升级包当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。

16.根据权利要求15所述的基于Root权限的程序升级装置，其特征在于，所述装置包括：

普通升级模块，用于在所述升级包的特征信息与已安装应用程序的特征信息相同时，对所述已安装应用程序进行普通升级。

17.根据权利要求16所述的基于Root权限的程序升级装置，其特征在于，应用程序的升级通过静默或用户触发的方式实现。

18.根据权利要求10所述的基于Root权限的程序升级装置，其特征在于，所述升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，所述数字标识为0，当升级前后的包名和/或签名不同时，所述数字标识为其他数字。

基于Root权限的程序升级方法及装置

技术领域

[0001] 本发明涉及计算机领域，具体而言，本发明涉及一种基于Root权限的程序升级方法及装置。

背景技术

[0002] 为满足用户需求，对应用程序升级是非常必要的，通过升级可增加新功能以提高用户体验，还可修复程序中的漏洞。Android系统是一种以Linux为基础的开放源代码操作系统，主要应用于移动设备，如：手机和平板电脑等。目前，基于Android平台的应用程序升级，要求应用程序升级前后的包名和签名一致，即只允许应用程序在同一个包名和签名进行覆盖安装和升级。

[0003] 但应用程序在使用过程中，存在如下情况：签名泄露、签名被破解；包名有关键词侵权；包名或签名更新。在上述情况下，若将应用程序升级至新的包名和/或签名，则很难实现。导致应用程序无法维护、用户无法得到新的服务、用户流失、不合法的包名或泄露的签名继续流通，给产品带来巨大的损失。

发明内容

[0004] 本发明的目的旨在至少解决上述技术缺陷之一，特别是在具有Root权限时，可对应用程序进行跨包名和/或签名升级，以避免用户流失。

[0005] 本发明提供一种基于Root权限的程序升级方法，包括以下步骤：获取已安装应用程序的升级包；在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包；在升级包安装正确时，卸载已安装应用程序。

[0006] 其中，特征信息包括包名和签名。

[0007] 其中，在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包的步骤具体包括：发送获取升级包特征信息的请求；接收响应请求所返回的升级包特征信息；判断升级包的特征信息与已安装应用程序的特征信息是否一致；若不一致，则安装升级包。

[0008] 其中，在升级包安装正确时，卸载已安装应用程序的步骤具体包括：升级包安装完成后，发送启动该应用的请求；若安装后的升级包可启动，则升级包安装正确，卸载已安装应用程序。

[0009] 其中，特征信息还包括应用大小和应用中各文件的MD5值。

[0010] 其中，安装的升级包为合法的升级包。

[0011] 其中，接收响应请求所返回的升级包特征信息后，判断该特征信息与升级包当前特征信息是否一致，若一致，则获取的升级包合法，升级包当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。

[0012] 其中，在升级包的特征信息与已安装应用程序的特征信息相同时，对已安装应用程序进行普通升级。

[0013] 其中，应用程序的升级通过静默或用户触发的方式实现。

[0014] 其中，升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，数字标识为其他数字。

[0015] 本发明还提供一种基于Root权限的程序升级方法，包括以下步骤：接收已安装应用程序的升级指令；根据升级指令推送相应的升级包；接收获取升级包特征信息的请求；根据请求返回升级包的特征信息。

[0016] 其中，特征信息包括包名和签名。

[0017] 其中，特征信息还包括应用大小和应用中各文件的MD5值。

[0018] 其中，升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，数字标识为其他数字。

[0019] 本发明提供一种基于Root权限的程序升级装置，包括：获取模块，用于获取已安装应用程序的升级包；安装模块，用于在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包；卸载模块，用于在升级包安装正确时，卸载已安装应用程序。

[0020] 其中，特征信息包括包名和签名。

[0021] 其中，安装模块具体用于：发送获取升级包特征信息的请求；接收响应请求所返回的升级包特征信息；判断升级包的特征信息与已安装应用程序的特征信息是否一致；若不一致，则安装升级包。

[0022] 其中，卸载模块具体用于：升级包安装完成后，发送启动该应用的请求；若安装后的升级包可启动，则升级包安装正确，卸载已安装应用程序。

[0023] 其中，特征信息还包括应用大小和应用中各文件的MD5值。

[0024] 其中，安装的升级包为合法的升级包。

[0025] 其中，安装模块具体用于：接收响应请求所返回的升级包特征信息后，判断该特征信息与升级包当前特征信息是否一致，若一致，则获取的升级包合法，升级包当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。

[0026] 其中，装置还包括：普通升级模块，用于在升级包的特征信息与已安装应用程序的特征信息相同时，对已安装应用程序进行普通升级。

[0027] 其中，应用程序的升级通过静默或用户触发的方式实现。

[0028] 其中，升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，数字标识为其他数字。

[0029] 本发明还提供一种基于Root权限的程序升级装置，包括：第一接收模块，用于接收已安装应用程序的升级指令；推送模块，用于根据升级指令推送相应的升级包；第二接收模块，用于接收获取升级包特征信息的请求；返回模块，用于根据请求返回升级包的特征信息。

[0030] 其中，特征信息包括包名和签名。

[0031] 其中，特征信息还包括应用大小和应用中各文件的MD5值。

[0032] 其中，升级包的特征信息还包括数字标识，当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，数字标识为其他数字。

[0033] 与现有技术相比，本发明具有以下优点：

[0034] 1.将服务器端升级包的特性信息与下载后升级包的当前特征信息进行比较，当二者的应用签名或应用中各文件的MD5值相同时，该下载的升级包合法。其中，应用中各文件

的MD5值为重新生成。通过合法性验证，可确保下载的升级包没有被篡改，为官方的升级包。

[0035] 2.将服务器端升级包的特性信息与已安装应用程序的特征信息进行比较，若二者的包名和/或签名不同，在具有Root权限的情况下，进行跨包名和/或签名升级，若二者的包名和签名相同，进行普通升级。通过该方式，只要开发者对同款软件升级，不论包名和/或签名相同与否，均可进行升级。

[0036] 3.升级方式可采用静默的方式在后台升级，方便快捷。

[0037] 本发明提出的上述方案，通过跨包名和/或签名升级，可维持原应用程序的用户，避免用户的流失。

[0038] 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

[0039] 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：

[0040] 图1为本发明系统结构原理图；

[0041] 图2为本发明基于Root权限的程序升级方法一实施例的流程示意图；

[0042] 图3为本发明基于Root权限的程序升级方法另一实施例的流程示意图；

[0043] 图4为本发明系统基于Root权限的程序升级方法一实施例的流程示意图；

[0044] 图5为本发明基于Root权限的程序升级装置一实施例的结构示意图。

具体实施方式

[0045] 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。

[0046] 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。

[0047] 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。

[0048] 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。

[0049] 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。

[0050] 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。

[0051] 请参阅图1,图1为本发明系统结构原理图,如图1所示,包括客户端11和服务器端12。

[0052] 图1所示系统为基于网络环境所构建的系统,客户端11为安装有应用程序的智能终端(如:移动终端、电脑),服务器端12设有应用程序升级所需的升级包。其中,客户端11为运行Android系统的智能终端,服务器端12可为云端。

[0053] 客户端11涉及Android系统,但不局限于该操作系统,本领域技术人员可以合理预见,可适应本发明构思的操作系统均可。

[0054] 请参阅图2,图2为本发明基于Root权限的程序升级方法一实施例的流程示意图,如图2所示,包括以下步骤:

[0055] S21,获取已安装应用程序的升级包。

[0056] 本实施例的方法在图1所示的客户端11实施。应用程序的升级具有多种方式,以手机为例,如:根据手机助手推送的升级信息,发送升级请求;根据软件自身推送的升级信息,发送升级请求;在没有升级提示下,用户主动查看是否可升级,若可升级,发送升级请求。通过升级请求,获取图1所示服务器端12推送的升级包。其中,升级请求中通常携带有需升级应用程序的相关信息,如包名等。

[0057] 客户端11中安装的应用程序包括用户安装的应用程序和系统内置的应用程序,为

方便用户管理或数据读取，升级包可存储至flash盘或SD卡，通常优先存储至SD卡，以防止设备变慢。

[0058] S22，在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包。

[0059] 本实施例客户端11系统中注册有service，用于进行步骤S22、S23的工作。在升级包安装前，需要判断升级包是否合法，采用何种方式升级。

[0060] 判断已安装应用程序升级方式的方法如下：

[0061] A.发送获取升级包特征信息的请求。

[0062] 通过URL发送获取升级包特征信息的请求至服务器端12，服务器端12根据该请求查找此升级包在服务器端12存储的特征信息，并以JSON字符串的方式返回。

[0063] 返回的特征信息包括：包名和签名，还可包括以下一种或多种信息：数字标识、应用大小、应用中各文件的MD5值。

[0064] 包名源自于Java的package的概念，按照package的命名风格，如某个应用程序的包名为com.qihoo360.mobilesafe，Android系统要求每个应用程序都声明一个唯一的包名。如果需安装的应用程序的包名与已安装的应用程序的包名重复，则该需安装的应用程序无法安装。

[0065] 根据包名的原理，Android系统的应用程序在升级时，首先卸载已安装的应用程序，再安装其升级包。

[0066] 出于安全性的目的，Android系统要求每个应用程序都包含开发者签名，签名也可称为代码签名，附加于应用程序上，用于防伪和防篡改。如果应用程序的签名与其官方的签名不一致，则认为应用程序可能被篡改。在提取签名时，对于安卓应用而言，可以从程序中的元信息(META-INF)目录下提取。

[0067] 数字标识如：0、1、2等，当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，所述数字标识为其他数字。

[0068] 应用程序目录下各文件的MD5值，是对各文件利用现有的校验算法(如MD5算法)计算产生的一个校验值，校验值可用于验证应用程序的合法性，即完整且未被篡改。

[0069] 现有技术中，Android系统应用程序的升级需已安装应用程序与其升级包的包名和签名相同，否则无法升级。

[0070] B.接收响应请求所返回的升级包特征信息。

[0071] C.判断升级包的特征信息与已安装应用程序的特征信息是否一致。

[0072] D.若不一致，采用跨包名和/或签名的方式升级。

[0073] 已安装应用程序的特征信息从已安装应用程序的安装包提取，包括签名和包名。

[0074] 将升级包的包名和签名与已安装应用程序的包名和签名进行对比，若二者的包名和/或签名不相同，则需采用跨包名和/或签名的方式升级，若二者的包名和/或签名相同，则采用普通升级方式。

[0075] 还可利用升级包特征信息中的数字标识进行判断，若数字标识为0，则采用普通升级方式，若数字标识为其他数字，则采用跨包名和/或签名的方式升级。

[0076] 虽然已安装应用程序的包名和签名与其升级包的包名和签名不一致，但服务器端12会注明二者的联系，当已安装应用程序发送升级请求时，服务器端12会推送该升级包。

[0077] 在其他实施例中，还可在服务器端12判断升级包与已安装应用程序的包名、签名

是否相同，具体为，当已安装应用程序发送升级请求时，请求包括该应用相应的包名和签名，服务器端12将接收的包名和签名与其存储的升级包的包名和签名进行比较，并将比较结果返回至客户端11，客户端11根据比较结果采用相应的升级方式进行升级。

[0078] 不论采用跨包名和/或签名的方式升级，还是采用普通方式升级，在安装升级包前，需判断升级包是否合法，即是否下载完整、是否被篡改。

[0079] 判断获取的升级包是否合法的方法如下：

[0080] A.接收响应请求所返回的升级包特征信息。

[0081] 此步骤在判断已安装应用程序升级方式的方法中已有阐述，在此不再赘述。

[0082] B.判断返回的升级包的特征信息与下载后升级包的当前特征信息是否一致。

[0083] 返回的升级包的特征信息包括应用大小、签名或应用大小、应用中各文件的MD5值。升级包的当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。将二者的应用大小、签名进行比较，可先对应用大小进行初步判断，再判断签名，若应用大小不同，就可说明升级包不合法。或将二者的应用大小、应用中各文件的MD5值进行比较，由于应用大小的比较较为方便，可先对应用大小进行初步判断，再判断应用中各文件的MD5值。或将二者的应用大小、签名、应用中各文件的MD5值综合比较，以较精确的判断结果。

[0084] 当签名不同，说明升级包被恶意篡改，当MD5值不同，说明升级包没有下完整或被恶意篡改。

[0085] 在某些情况下，以手机为例，升级包的签名可能在SD卡上被篡改，这时还需验证签名的MD5值。具体为：首先根据升级包重新生成签名MD5值，然后跟升级包中的原签名MD5值进行比较，若一致，则该签名合法。

[0086] 在其他实施例中，还可采用应用程序的MD5值进行判断。

[0087] 在其他实施例中，还可将下载后升级包的当前特征信息发送至服务器端12进行判断。

[0088] C.若一致，则下载的升级包合法。

[0089] 当返回的升级包的特征信息与下载后升级包的当前特征信息一致时，则下载的升级包合法。

[0090] 以上所述，升级包合法性的判断与已安装应用程序升级方式的判断可同时进行，也可优先判断合法性。若下载的升级包不合法，将下载的升级包删除。

[0091] 采用跨包名和/或签名的方式升级，其过程具体为，首先安装升级包，然后卸载已安装的应用程序。此过程的实现需要在具有Root权限的情况下实施，因此需要获得Root权限，Root权限的获取方式如下：

[0092] 目前有多种提权方案用于获取Android系统的Root权限，依提权后权限作用的生命周期来看，包括永久Root权限和临时Root权限。永久Root权限情况下，应用程序一经Root授权，以后可不必再进行Root提权；而临时Root权限情况下，权限作用的生命周期只是操作系统的一次从开机到关机的过程，下次开机依然需要进行Root。

[0093] 无论采用何种Root方式，提权的基本原理均是通过向系统植入用于接收权限请求的su，再结合SuperUser.apk应用程序实现人机交互。Root提权操作的过程具体为：把su文件放到/system/bin/中，把Superuser.apk放到system/app下面，前者用于监听用户的权限

请求并与后者通信，后者主要是在与前者通信的基础上实现人机交互，从而允许用户做出相关指示。理论上，如果su可以实现默认通过所有权限请求，则SuperUser.apk可以舍弃。此外还需要设置/system/bin/su可以让任意用户可运行，使其具有set uid和set gid的权限，具体可通过在android机器上运行命令：adbshell chmod 4755/system/bin/su实现。

[0094] 对于Root方案，应理解为包括：与破解相关的代码文件及其配置参数，以“su”、“SuperUser.apk”命名或实现的文件。

[0095] 当需获得Root权限时，发送Root请求，云端接收该请求后，根据请求中相关的机型信息，选择适合该机型的Root方案，并推送至客户端11，客户端11根据此Root方案获取Root权限。

[0096] 客户端11的service运行于后台，可调用PackageManagerService对升级包进行安装。

[0097] S23，在升级包安装正确时，卸载已安装应用程序。

[0098] 卸载已安装应用程序前，需进行以下判断：

[0099] A.升级包安装完成后，发送启动该应用的请求。

[0100] B.若安装后的升级包可启动，则升级包安装正确，卸载已安装应用程序。

[0101] 以上所述完成应用程序的跨包名和/签名升级，在升级过程中，通过状态机判断每一步的状态，如：升级包下载是否成功、升级包是否合法等状态，当状态满足时，执行下一步的操作，若状态不满足条件，则退出升级。

[0102] 本实施例在升级过程中，采用静默的方式升级。在其他实施例中，可采用用户触发，即用户确认的方式升级。

[0103] 需要指出的是，本实施例的方法可作为独立产品实现，也可作为附加功能添加至其他产品，如：360手机助手。

[0104] 以上所述，本实施例可对应用程序进行跨包名和/或签名升级，避免由于应用程序升级包改变包名和/或签名，使得应用程序无法维护而导致用户流失的问题。且利用本实施例的方法，可对应用程序的包名和/或签名进行更新，以避免应用程序由其包名和/或签名带来的问题，确保应用程序的持续开发。

[0105] 请参阅图3，图3为本发明基于Root权限的程序升级方法另一实施例的流程示意图，如图3所示，包括以下步骤：

[0106] S31，接收已安装应用程序的升级指令。

[0107] 本实施例的方法在图1所示的服务器端12实施，服务器端12设有升级包及与该升级包相关的特征信息。

[0108] S32，根据升级指令推送相应的升级包。

[0109] 服务器端12接收已安装应用程序的升级指令后，对该指令进行解析，获得相应升级包的特征信息，如：是否跨包名和/或签名、路径、MD5值、应用大小等信息，根据路径确定升级包的位置，并将升级包推送至图1所示的客户端11。

[0110] S33，接收获取升级包特征信息的请求。

[0111] S34，根据请求返回升级包的特征信息。

[0112] 服务器端12返回的升级包的特征信息包括包名和签名，还可包括以下一种或多种信息：数字标识、应用大小、应用中各文件的MD5值。

[0113] 其中，利用包名、签名、数字标识可判断已安装应用程序是否为跨包名和/或签名升级，利用签名、应用大小、应用中各文件的MD5值可判断下载后的升级包是否合法。

[0114] 以上所述，服务器端12与客户端11相互配合，共同完成已安装应用程序的跨包名和/或签名升级，避免用户断层。

[0115] 请参阅图4，图4为本发明系统基于Root权限的程序升级方法一实施例的流程示意图，如图4所示，包括以下步骤：

[0116] S41，服务器端接收客户端发送的已安装应用程序的升级指令。

[0117] S42，根据升级指令推送相应的升级包。

[0118] S43，客户端获取已安装应用程序的升级包。

[0119] S44，服务器端接收客户端发送的获取升级包特征信息的请求。

[0120] S45，根据请求返回升级包的特征信息。

[0121] S46，在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包。

[0122] S47，在升级包安装正确时，卸载已安装应用程序。

[0123] 上述步骤在图2和图3所示的实施例中，均由详细的描述，在此不再赘述。

[0124] 请参阅图5，图5为本发明基于Root权限的程序升级装置一实施例的结构示意图，如图5所示，包括获取模块51、安装模块52、卸载模块53、普通升级模块54、第一接收模块55、推送模块56、第二接收模块57及返回模块58。

[0125] 上述各模块的功能如下：

[0126] 获取模块51用于获取已安装应用程序的升级包。安装模块52用于在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包。卸载模块53用于在升级包安装正确时，卸载已安装应用程序。普通升级模块54用于在升级包的特征信息与已安装应用程序的特征信息相同时，对已安装应用程序进行普通升级。

[0127] 第一接收模块55用于接收已安装应用程序的升级指令。推送模块56用于根据升级指令推送相应的升级包。第二接收模块57用于接收获取升级包特征信息的请求。返回模块58用于根据请求返回升级包的特征信息。

[0128] 在本实施例中，结合图1，获取模块51、安装模块52、卸载模块53及普通升级模块54位于客户端11中，第一接收模块55、推送模块56、第二接收模块57及返回模块58位于服务器端12中，客户端11与服务器端12相互交互，下面详细阐述在交互过程中，各模块的工作过程。

[0129] 客户端11中已安装应用程序可升级时，由用户发出升级指令，服务器端12第一接收模块55接收该升级指令，对该升级指令进行解析，查找相应的升级包，推送模块56将查找的升级包推送至客户端11。客户端11获取模块51获取推送模块56推送的升级包后，安装模块52发送获取该升级包特征信息的请求，第二接收模块57接收该请求后，查找服务器端12存储的该升级包的特征信息，返回模块58将此特征信息返回至客户端11。安装模块52接收返回的升级包特征信息后，判断该升级包的特征信息与已安装应用程序的特征信息是否一致，主要判断二者的包名和签名，当判断结果为不一致时，对已安装应用程序采用跨包名和/或签名的方式升级，当判断结果为一致时，对已安装应用程序采用普通升级方式升级。跨包名和/或签名的方式升级具体为，首先安装模块52安装该升级包，升级包安装完成后，卸载模块53发送启动安装后的升级包的请求，若安装后的升级包可启动，卸载模块53卸载

原来的已安装应用程序，至此完成跨包名和/或签名升级。普通升级的方式具体为，普通升级模块54对已安装应用程序进行升级。

[0130] 其中，安装模块52接收返回的升级包特征信息后，还需判断该特征信息与下载后的升级包的当前特征信息是否一致，若一致，则获取的升级包合法，才能进行安装或判断以何种方式升级。在此判断过程中，升级包当前特征信息包括应用大小、签名或应用大小、重新生成的应用中各文件的MD5值。

[0131] 其中，服务器端12返回的特征信息包括：包名和签名，还可包括以下一种或多种信息：数字标识、应用大小、应用中各文件的MD5值。当升级前后的包名和签名相同时，数字标识为0，当升级前后的包名和/或签名不同时，数字标识为其他数字，因此，数字标识也可判断是否为跨包名和/或签名升级。

[0132] 其中，应用程序的升级通过静默或用户触发的方式实现。

[0133] 本实施例可对应用程序进行跨包名和/或签名升级，使本来由于包名和/或签名原因不能升级的应用程序具有新的功能，避免用户流失。

[0134] 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。


11
客户端
12
服务器端

图1

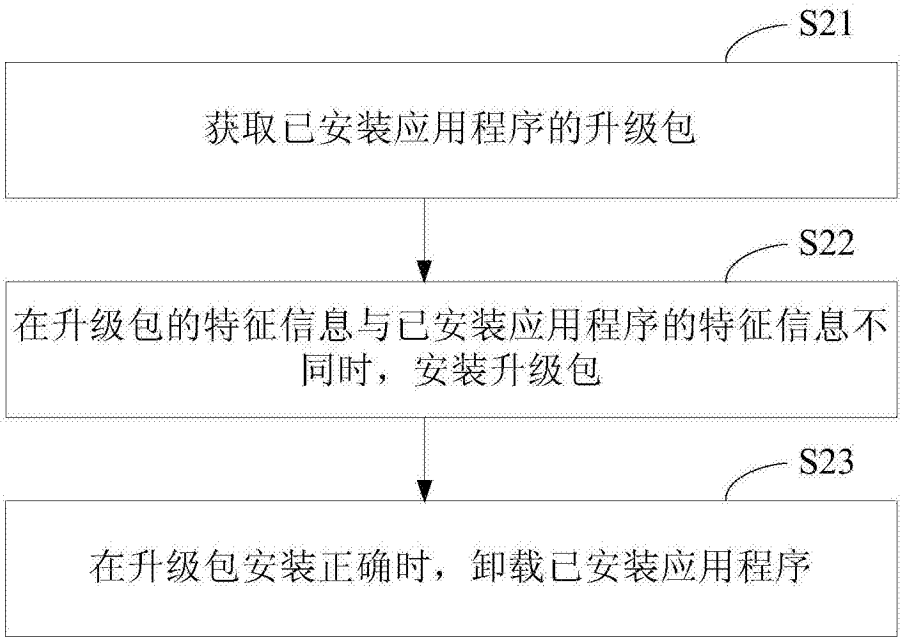

S21
获取已安装应用程序的升级包
S22
在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包
S23
在升级包安装正确时，卸载已安装应用程序

图2

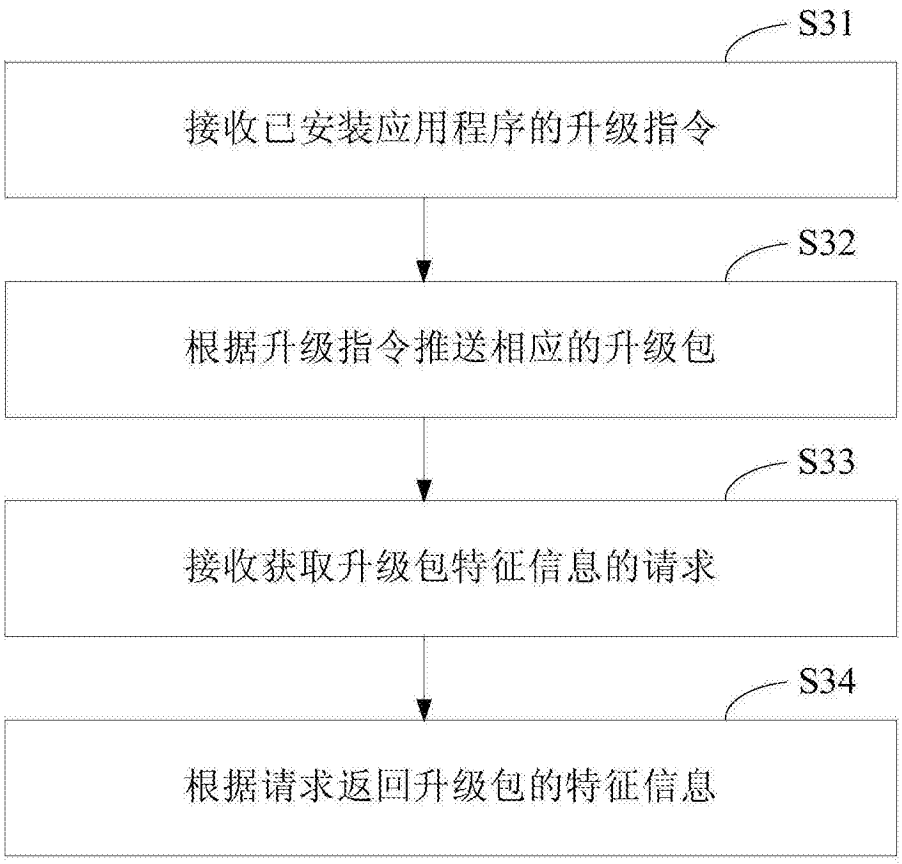

S31
接收已安装应用程序的升级指令
S32
根据升级指令推送相应的升级包
S33
接收获取升级包特征信息的请求
S34
根据请求返回升级包的特征信息

图3

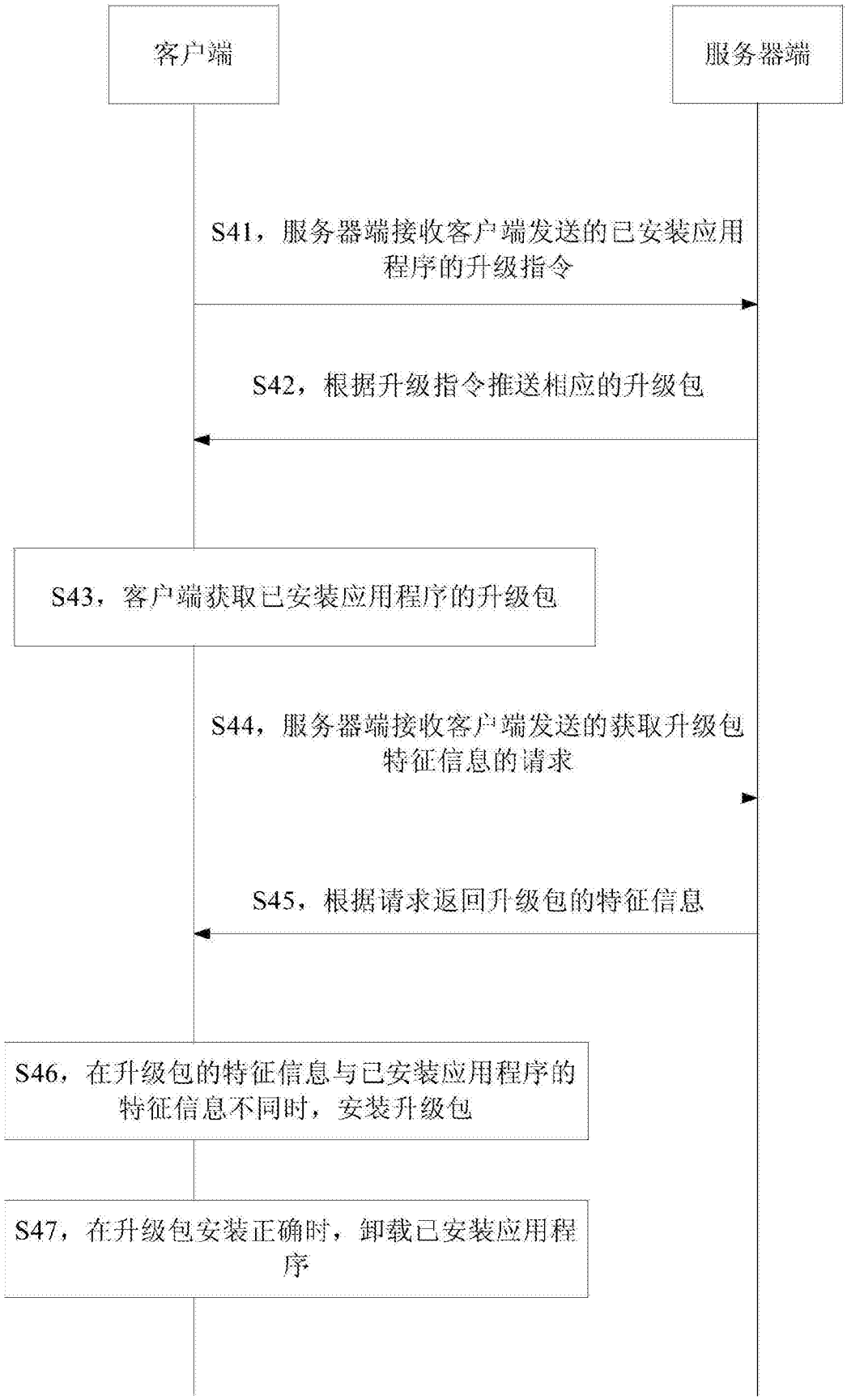

客户端
服务器端
S41，服务器端接收客户端发送的已安装应用程序的升级指令
S42，根据升级指令推送相应的升级包
S43，客户端获取已安装应用程序的升级包
S44，服务器端接收客户端发送的获取升级包特征信息的请求
S45，根据请求返回升级包的特征信息
S46，在升级包的特征信息与已安装应用程序的特征信息不同时，安装升级包
S47，在升级包安装正确时，卸载已安装应用程序

图4

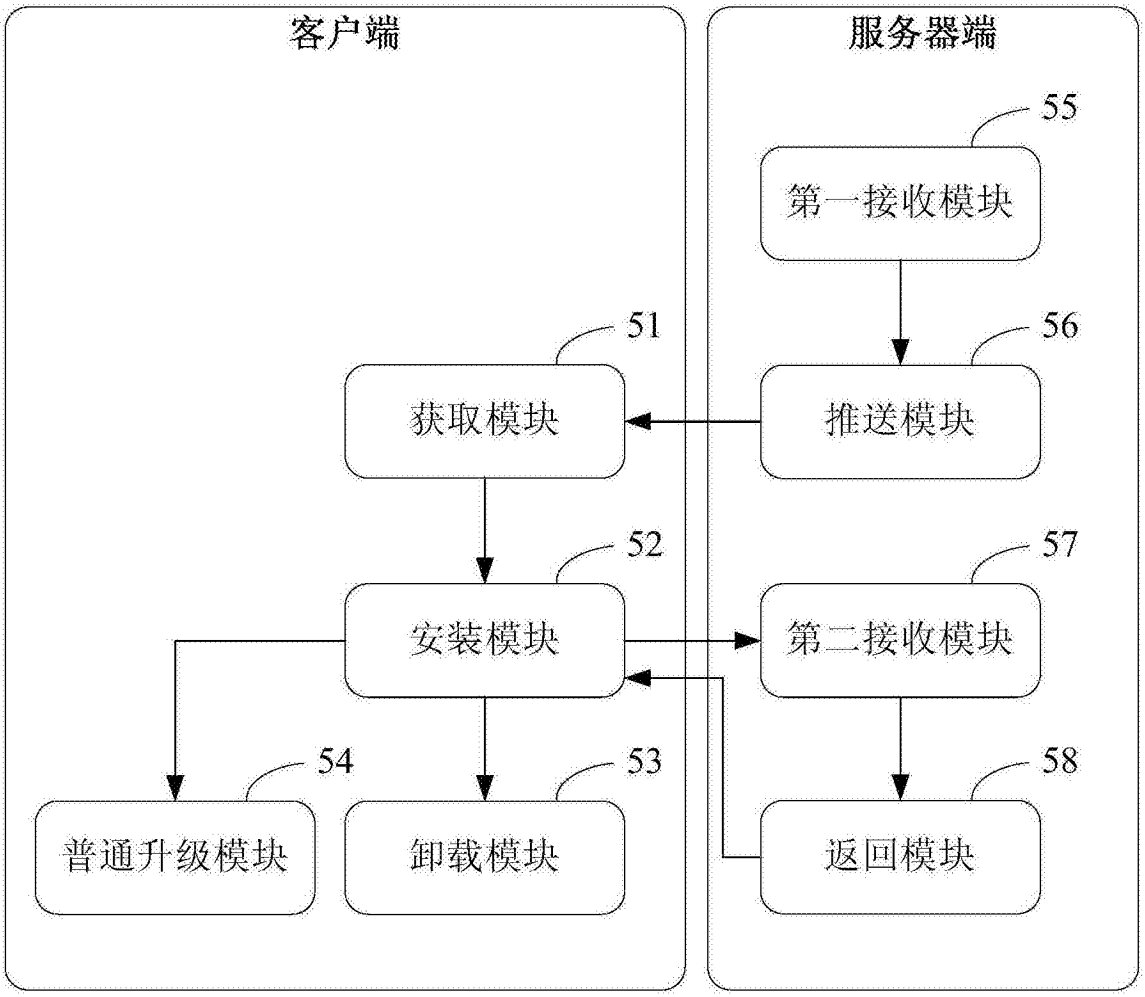

客户端
服务器端
55
第一接收模块
51
56
获取模块
推送模块
52
57
安装模块
第二接收模块
54
53
58
普通升级模块
卸载模块
返回模块

图5