



(12) 发明专利

(10) 授权公告号 CN 110780915 B

(45) 授权公告日 2022. 01. 25

(21) 申请号 201910759708.8

(22) 申请日 2019.08.16

(65) 同一申请的已公布的文献号  
申请公布号 CN 110780915 A

(43) 申请公布日 2020.02.11

(73) 专利权人 腾讯科技(深圳)有限公司  
地址 518044 广东省深圳市南山区高新区  
科技中一路腾讯大厦35层

(72) 发明人 彭秦中 王菲

(74) 专利代理机构 北京同达信恒知识产权代理  
有限公司 11291  
代理人 郭润湘 李娟

(51) Int. Cl.  
G06F 8/71 (2018.01)

(56) 对比文件

CN 201248052 Y, 2009.05.27

CN 110032358 A, 2019.07.19

CN 108288196 A, 2018.07.17

CN 109214159 A, 2019.01.15

US 2018309746 A1, 2018.10.25

审查员 陈亭玉

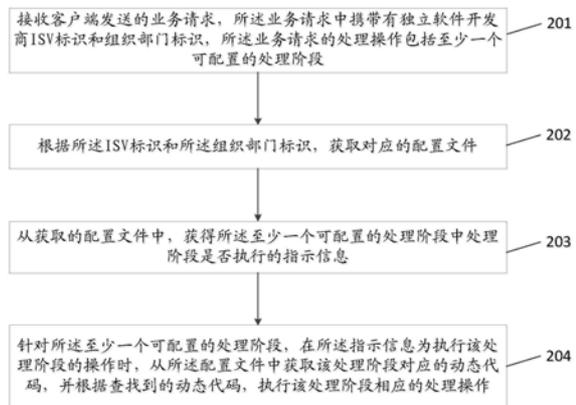
权利要求书2页 说明书15页 附图5页

(54) 发明名称

一种数据处理方法、装置和存储介质

(57) 摘要

本申请实施例提供一种数据处理方法、装置和存储介质,用以解决ISV适配方案复杂,维护成本大的问题,涉及互联网技术领域。该方法中,接收客户端发送的业务请求,根据ISV标识和组织部门标识,获取对应的配置文件;从获取的配置文件中,获得至少一个可配置的处理阶段中处理阶段是否执行的指示信息;针对至少一个可配置的处理阶段,从配置文件中获取该处理阶段对应的动态代码,并根据查找到的动态代码,执行该处理阶段相应的处理操作。这样,由于通过为业务请求的可配置的处理阶段配置动态代码,在新增一个ISV时,只需要在配置文件中新增一条配置即可,不需要修改代码,也无需增加实现类,故而适配难度小,便于维护。



1. 一种数据处理方法,其特征在于,所述方法包括:

接收客户端发送的业务请求,所述业务请求中携带有独立软件开发商ISV标识和组织部门标识,所述业务请求的处理操作包括至少一个可配置的处理阶段;其中,所述可配置的处理阶段是根据不同ISV对所述业务请求的处理差异确定的;

根据所述ISV标识和所述组织部门标识,获取对应的配置文件;

从获取的配置文件中,获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息;

针对所述至少一个可配置的处理阶段,在所述指示信息为执行该处理阶段的操作时,从所述配置文件中获取该处理阶段对应的代码,并根据查找到的代码,执行该处理阶段相应的处理操作。

2. 根据权利要求1所述的方法,其特征在于,所述组织部门标识为医院标识。

3. 根据权利要求1所述的方法,其特征在于,所述至少一个可配置的处理阶段为根据不同业务类型的ISV接口的处理差异确定的。

4. 根据权利要求1所述的方法,其特征在于,所述至少一个可配置的处理阶段包括以下中的至少一种:

签名处理阶段、签名检验处理阶段、请求包格式转换阶段以及响应包格式转换阶段,其中,所述请求包为调用ISV接口发送的数据包,所述响应包为从所述ISV接口接收到的数据包。

5. 根据权利要求4所述的方法,其特征在于,在处理完签名处理阶段和请求包格式转换阶段的处理操作之后,还包括:

在所述配置文件中,获取所述ISV标识对应的ISV接口的访问地址;

通过所述访问地址调用所述ISV接口将格式转换后的请求包以及签名,发送给组织部门标识对应的业务处理设备。

6. 根据权利要求5所述的方法,其特征在于,接收客户端发送的业务请求之后,还包括:

在所述配置文件中,查找所述业务请求对应的调用者标识和调用接口标识;

根据所述调用者标识和所述调用接口标识,确定所述调用者标识对应的调用者具有所述ISV标识对应的接口的访问权限,继续执行所述根据所述ISV标识和所述组织部门标识,获取对应的配置文件的步骤。

7. 根据权利要求4所述的方法,其特征在于,所述配置文件包括路由配置表、签名配置表和格式配置表;

在获取的配置文件中,查找用于分别表征各处理阶段是否执行的指示信息,包括:

在所述路由配置表中,查找所述指示信息;

若所述至少一个可配置的处理阶段为签名处理阶段或签名检验处理阶段,若指示信息为执行相应处理阶段,则从所述配置文件中获取相应处理阶段对应的动态代码,包括:

根据所述ISV标识和所述组织部门标识获取对应的签名配置表;

在所述签名配置表中,查找所述签名处理阶段或所述签名检验处理阶段的动态代码;

若所述至少一个可配置的处理阶段为请求包格式转换阶段或响应包格式转换阶段,若指示信息为执行相应处理阶段,则从所述配置文件中获取相应处理阶段对应的动态代码,包括:

根据所述ISV标识和所述组织部门标识获取对应的格式配置表；  
在所述格式配置表中，查找所述请求包格式转换阶段或响应包格式转换阶段的动态代码。

8. 根据权利要求1所述的方法，其特征在于，所述方法还包括：

显示配置界面；

在所述配置界面中，配置ISV标识和组织部门标识的至少一个可配置的处理阶段对应的指示信息至所述配置文件中；

配置至少一个可配置的处理阶段的动态代码至所述配置文件中。

9. 一种数据处理装置，其特征在于，所述装置包括：

接收模块，用于接收客户端发送的业务请求，所述业务请求中携带有独立软件开发商ISV标识和组织部门标识，所述业务请求的处理操作包括至少一个可配置的处理阶段；其中，所述可配置的处理阶段是根据不同ISV对所述业务请求的处理差异确定的；

获取模块，用于根据所述ISV标识和所述组织部门标识，获取对应的配置文件；

所述获取模块还用于，从获取的配置文件中，获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息；

所述获取模块还用于，针对所述至少一个可配置的处理阶段，在所述指示信息为执行该处理阶段的操作时，从所述配置文件中获取该处理阶段对应的代码；并，

处理模块，用于根据查找到的代码，执行该处理阶段相应的处理操作。

10. 一种计算机可读介质，存储有计算机可执行指令，其特征在于，所述计算机可执行指令用于执行如权利要求1-8中任一权利要求所述的方法。

11. 一种电子设备，其特征在于，包括：至少一个处理器；以及与所述至少一个处理器通信连接的存储器；其中，所述存储器存储有可被所述至少一个处理器执行的指令，所述指令被所述至少一个处理器执行，以使所述至少一个处理器能够执行如权利要求1-8中任一权利要求所述的方法。

## 一种数据处理方法、装置和存储介质

### 技术领域

[0001] 本申请涉及互联网技术领域,尤其涉及一种数据处理方法、装置和存储介质。

### 背景技术

[0002] 现有的网关服务,主要接入协议是http(HyperText Transfer Protocol,超文本传输协议)或者是https(Hyper Text Transfer Protocol over Secure Socket Layer,超文本传输安全协议),主要的功能是做到路由转发、统一接入功能,不能对同一业务的不同ISV(Independent Software Vendors,独立软件开发商)提供的接口做统一适配。

[0003] 现有技术中,为同一业务的不同ISV提供的接口做统一适配的主流方案,主要是使用设计模式中的工厂方法,针对不同的ISV定义不同的实现类,通过创建不同的实现类来完成差异化的ISV适配。

[0004] 然而通过工厂方法进行适配的方案,存在着维护成本非常大,每增加一个ISV就需要修改代码,增加一个实现类。另外,每增加一个ISV的适配,需要重新做一次应用发布,在发布过程中,其他ISV的接口调用也会受到影响。

### 发明内容

[0005] 本申请实施例提供一种数据处理方法和装置,用以解决ISV适配方案复杂,维护成本大的问题。

[0006] 第一方面,本申请实施例中提供一种数据处理方法,包括:

[0007] 接收客户端发送的业务请求,所述业务请求中携带有独立软件开发商ISV标识和组织部门标识,所述业务请求的处理操作包括至少一个可配置的处理阶段;

[0008] 根据所述ISV标识和所述组织部门标识,获取对应的配置文件;

[0009] 从获取的配置文件中,获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息;

[0010] 针对所述至少一个可配置的处理阶段,在所述指示信息为执行该处理阶段的操作时,从所述配置文件中获取该处理阶段对应的动态代码,并根据查找到的动态代码,执行该处理阶段相应的处理操作。

[0011] 可选的,所述组织部门标识为医院标识。

[0012] 可选的,所述至少一个可配置的处理阶段为根据不同业务类型的ISV接口的处理差异确定的。

[0013] 可选的,所述至少一个可配置的处理阶段包括以下中的至少一种:

[0014] 签名处理阶段、签名检验处理阶段、请求包格式转换阶段以及响应包格式转换阶段,其中,所述请求包为调用ISV接口发送的数据包,所述响应包为从所述ISV接口接收到的数据包。

[0015] 可选的,在处理完签名处理阶段和请求包格式转换阶段的处理操作之后,还包括:

[0016] 在所述配置文件中,获取所述ISV标识对应的ISV接口的访问地址;

- [0017] 通过所述访问地址调用所述ISV接口将格式转换后的请求包以及签名,发送给组织部门标识对应的业务处理设备。
- [0018] 可选的,接收终端发送的业务请求之后,还包括:
- [0019] 在所述配置文件中,查找所述业务请求对应的调用者标识和调用接口标识;
- [0020] 根据所述调用者标识和所述调用接口标识,确定所述调用者标识对应的调用者具有所述ISV标识对应的接口的访问权限,继续执行所述根据所述ISV标识和所述组织部门标识,获取对应的配置文件的步骤。
- [0021] 可选的,所述配置文件包括路由配置表、签名配置表和格式配置表;
- [0022] 在获取的配置文件中,查找用于分别表征各处理阶段是否执行的指示信息,包括:
- [0023] 在所述路由配置表中,查找所述指示信息;
- [0024] 若所述至少一个可配置的处理阶段为签名处理阶段或签名校验处理阶段,若指示信息为执行相应处理阶段,则从所述配置文件中获取相应处理阶段对应的动态代码,包括:
- [0025] 根据所述ISV标识和所述组织部门标识获取对应的签名配置表;
- [0026] 在所述签名配置表中,查找所述签名处理阶段或所述签名校验处理阶段的动态代码;
- [0027] 若所述至少一个可配置的处理阶段为请求包格式转换阶段或响应包格式转换阶段,若指示信息为执行相应处理阶段,则从所述配置文件中获取相应处理阶段对应的动态代码,包括:
- [0028] 根据所述ISV标识和所述组织部门标识获取对应的格式配置表;
- [0029] 在所述格式配置表中,查找所述请求包格式转换阶段或响应包格式转换阶段的动态代码。
- [0030] 可选的,所述方法还包括:
- [0031] 显示配置界面;
- [0032] 在所述配置界面中,配置ISV标识和组织部门标识的至少一个可配置的处理阶段对应的指示信息至所述配置文件中;
- [0033] 配置至少一个可配置的处理阶段的动态代码至所述配置文件中。
- [0034] 第二方面,本申请实施例提供一种数据处理装置,包括:
- [0035] 接收模块,用于接收客户端发送的业务请求,所述业务请求中携带有独立软件开发商ISV标识和组织部门标识,所述业务请求的处理操作包括至少一个可配置的处理阶段;
- [0036] 获取模块,用于根据所述ISV标识和所述组织部门标识,获取对应的配置文件;
- [0037] 所述获取模块还用于,从获取的配置文件中,获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息;
- [0038] 所述获取模块还用于,针对所述至少一个可配置的处理阶段,在所述指示信息为执行该处理阶段的操作时,从所述配置文件中获取该处理阶段对应的动态代码;并,
- [0039] 处理模块,用于根据查找到的动态代码,执行该处理阶段相应的处理操作。
- [0040] 可选的,所述组织部门标识为医院标识。
- [0041] 可选的,所述至少一个可配置的处理阶段为根据不同业务类型的ISV接口的处理差异确定的。
- [0042] 可选的,所述至少一个可配置的处理阶段包括以下中的至少一种:

[0043] 签名处理阶段、签名检验处理阶段、请求包格式转换阶段以及响应包格式转换阶段,其中,所述请求包为调用ISV接口发送的数据包,所述响应包为从所述ISV接口接收到的数据包。

[0044] 可选的,所述获取模块还用于,在处理完签名处理阶段和请求包格式转换阶段的处理操作之后,在所述配置文件中,获取所述ISV标识对应的ISV接口的访问地址;

[0045] 所述装置还包括:

[0046] 发送模块,用于通过所述访问地址调用所述ISV接口将格式转换后的请求包以及签名,发送给组织部门标识对应的业务处理设备。

[0047] 可选的,所述装置还包括:

[0048] 查找模块,用于接收终端发送的业务请求之后,在所述配置文件中,查找所述业务请求对应的调用者标识和调用接口标识;

[0049] 确定模块,用于根据所述调用者标识和所述调用接口标识,确定所述调用者标识对应的调用者具有所述ISV标识对应的接口的访问权限,继续执行所述根据所述ISV标识和所述组织部门标识,获取对应的配置文件的步骤。

[0050] 可选的,所述配置文件包括路由配置表、签名配置表和格式配置表;

[0051] 所述查找模块具体用于:在所述路由配置表中,查找所述指示信息;

[0052] 若所述至少一个可配置的处理阶段为签名处理阶段或签名校验处理阶段,若指示信息为执行相应处理阶段,则所述获取模块具体用于:

[0053] 根据所述ISV标识和所述组织部门标识获取对应的签名配置表;

[0054] 在所述签名配置表中,查找所述签名处理阶段或所述签名校验处理阶段的动态代码;

[0055] 若所述至少一个可配置的处理阶段为请求包格式转换阶段或响应包格式转换阶段,若指示信息为执行相应处理阶段,则所述获取模块具体用于:

[0056] 根据所述ISV标识和所述组织部门标识获取对应的格式配置表;

[0057] 在所述格式配置表中,查找所述请求包格式转换阶段或响应包格式转换阶段的动态代码。

[0058] 可选的,所述装置还包括:

[0059] 显示模块,用于显示配置界面;

[0060] 配置模块,用于在所述配置界面中,配置ISV标识和组织部门标识的至少一个可配置的处理阶段对应的指示信息至所述配置文件中;

[0061] 所述配置模块还用于,配置至少一个可配置的处理阶段的动态代码至所述配置文件中。

[0062] 第三方面,本申请另一实施例还提供了一种电子设备,包括至少一个处理器;以及与所述至少一个处理器通信连接的存储器;其中,所述存储器存储有可被所述至少一个处理器执行的指令,所述指令被所述至少一个处理器执行,以使所述至少一个处理器能够执行本申请实施例提供的任一数据处理方法。

[0063] 第四方面,本申请另一实施例还提供了一种计算机存储介质,其中,所述计算机存储介质存储有计算机可执行指令,所述计算机可执行指令用于使计算机执行本申请实施例中的任一数据处理方法。

[0064] 本申请实施例提供的数据处理方法、装置和存储介质,由于通过为业务请求的可配置的处理阶段配置动态代码,在新增一个ISV时,只需要在配置文件中新增一条配置即可,不需要修改代码,也无需增加实现类,故而适配难度小,便于维护,由于只需要更新配置所以无需重新做一次应用发布,不影响其它ISV的接口调用。

[0065] 本申请的其它特征和优点将在随后的说明书中阐述,并且,部分地从说明书中变得显而易见,或者通过实施本申请而了解。本申请的目的和其他优点可通过在所写的说明书、权利要求书、以及附图中所特别指出的结构来实现和获得。

## 附图说明

[0066] 为了更清楚地说明本发明实施例的技术方案,下面将对本发明实施例中所需要使用的附图作简单地介绍,显而易见地,下面所介绍的附图仅仅是本发明的一些实施例,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的附图。

[0067] 图1为本申请实施例中一种数据处理方法应用场景示意图;

[0068] 图2为本申请实施例中一种数据处理方法的流程示意图;

[0069] 图3为本申请实施例中的配置界面示意图;

[0070] 图4为本申请实施例中一种数据处理方法的时序图;

[0071] 图5为申请实施例中一种数据处理装置示意图;

[0072] 图6为根据本申请实施方式的电子设备框图。

## 具体实施方式

[0073] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述。

[0074] 如背景技术所言,现有技术中,为同一业务不同ISV提供的接口做统一适配的主流方案,存在着维护成本大,且每增加一个ISV的适配,需要重新做一次应用发布。

[0075] 有鉴于此,发明人在研究中发现,可将一次ISV请求进行分阶段并拆分。划分的阶段可以包含“接收业务请求”、“生成请求签名”、“内部请求包格式转ISV请求包格式”、“发送http请求”、“接收http响应”、“ISV响应包转内部响应包格式”、“验证响应包签名参数”、“响应业务请求”等几个阶段。不同的ISV请求差异主要在于“生成请求签名”、“内部请求包格式转ISV请求包格式”、“ISV响应包转内部响应包格式”、“验证响应包签名参数”等几个阶段。存在差异的主要阶段可以视为可配置阶段,并对不同处理阶段的处理方式配置。

[0076] 有鉴于此,本申请实施例提供一种数据处理方法。该方法中,将业务请求的处理操作分成了不同的可配置的处理阶段。参阅图1,为本申请实施例中一种数据处理方法应用场景示意图。该场景例如是医院的挂号业务。如图1所示,可包括:终端000中安装的客户端100,中间服务器101以及目的服务器102。

[0077] 客户端100例如可以是即时通讯应用,该应用中安装有各类轻应用(例如是用户分诊的小程序),这些轻应用依附于客户端100的后台即中间服务器101进行开发。用户可以通过客户端触发查看指定医院的指定医生的剩余号源,客户端100根据用户的触发,可发送相关的业务请求给中间服务器101。

[0078] 实施时,客户端通过中间服务器101访问医院的服务器102来获取医院的相关信息。而客户端的开发者、中间服务器的开发者以及医院的服务器开发者存在开发差异。导致客户端和目的服务器之间收发的信息格式不统一。例如,客户端需要调用医院的ISV接口访问医院的服务器时,客户端开发的请求包不能直接转发给医院的服务器处理。不同开发商的客户端的业务请求需要进行适配。故此,本申请对业务请求的整个处理过程进行分阶段的划分,将其中存在主要差异的阶段设置为可配置的处理阶段,以便于适配不同的应用以及不同的医院。

[0079] 故此,中间服务器101在接收客户端100发送的业务请求后进行解析,得到ISV标识和组织部门标识。然后中间服务器101根据ISV标识和组织部门标识,获取对应的配置文件,并从获取的配置文件中,获得前述各可配置的处理阶段是否执行的指示信息。针对各可配置的处理阶段,若其指示信息为执行该处理阶段,则在执行到该处理阶段时从配置文件中获取对应的动态代码执行相应的处理操作。

[0080] 其中,终端000又可以称为用户设备(user equipment,UE)。终端可以是智能手机、平板电脑、各类可穿戴设备、车载设备等。

[0081] 参阅图2,为本申请实施例中一种数据处理方法的流程示意图,可以包括以下步骤:

[0082] 步骤201:接收客户端发送的业务请求,所述业务请求中携带有独立软件开发商ISV标识和组织部门标识,所述业务请求的处理操作包括至少一个可配置的处理阶段。

[0083] 具体实施时,至少一个可配置的处理阶段是根据不同业务类型的ISV接口的处理差异确定的。例如,如前文所述不同ISV的业务请求主要差异在于“生成请求签名”、“内部请求包格式转ISV请求包格式”、“ISV响应包转内部响应包格式”、“验证响应包签名参数”等几个处理阶段。故此,本申请实施例中的至少一个可配置的处理阶段包括以下中的至少一种:签名处理阶段、签名检验处理阶段、请求包格式转换阶段以及响应包格式转换阶段,其中,所述请求包为调用ISV接口发送的数据包,所述响应包为从所述ISV接口接收到的数据包。

[0084] 上述方法,将业务请求的处理操作拆分成可配置的处理阶段,针对各处理阶段则可以根据配置文件中的配置以及动态代码进行相应的处理。即使,新增一个ISV,也只需要更新配置文件即可,不需要修改代码。

[0085] 在一个可能的实施例中,ISV接口携带的组织部门标识可以是医院标识,则ISV的业务请求可以是预约挂号、分诊等业务。组织部门标识还可以是航空公司,则ISV的业务请求可以是预约购票等。

[0086] 步骤202:根据所述ISV标识和所述组织部门标识,获取对应的配置文件。

[0087] 步骤203:从获取的配置文件中,获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息。

[0088] 步骤204:针对所述至少一个可配置的处理阶段,在所述指示信息为执行该处理阶段的操作时,从所述配置文件中获取该处理阶段对应的动态代码,并根据查找到的动态代码,执行该处理阶段相应的处理操作。

[0089] 上述方法,由于通过为业务请求的可配置的处理阶段配置动态代码,在新增一个ISV时,只需要在配置文件中新增一条配置即可,不需要修改代码,也无需增加实现类,故而适配难度小,便于维护,由于只需要更新配置所以无需重新做一次应用发布,不影响其它

ISV的接口调用。

[0090] 本申请实施例中,以功能的接口为最小粒度进行ISV的接口适配。在数据库中存储有配置文件。具体实施时,可以多个ISV的实现类对应一个配置文件,也可以是一个ISV的实现类对应一个配置文件。为了便于维护,配置文件可以包括路由配置表、签名配置表和格式配置表等多个表格。其中,路由配置表主要用于存储调用的ISV接口的访问地址,以及各可配置的处理阶段是否执行的指示信息。其他表,用于存储相应处理阶段在执行时所需的动态代码和相关参数信息等。各表中,ISV标识和组织部门标识组合起来查询相关信息。

[0091] 如表一所示,为本申请实施例中的路由配置表。其中,文本类型中int表示整型,varchar表示可变长字符型,bool表示布尔型,(32)表示字符串长度为32bit(比特)。后文提交的表格中与表一相同的字符的含义相同,之后不再赘述。

[0092] 具体的,Fneed\_signature字段可对应是否需要签名以及Fneed\_transform字段对应的是否需要做包格式转换表示处理阶段的指示信息。例如,若业务请求对应的路由配置表中,Fneed\_signature字段对应于需要做签名,则该业务请求的签名处理阶段和签名校验处理阶段的指示信息为执行处理阶段的处理操作。

[0093] 又例如,业务请求对应的路由配置表中,Fneed\_signature字段对应于不需要做签名,则该业务请求对应的签名处理阶段和签名校验处理阶段的指示信息为不执行处理阶段的处理操作。

[0094] 表一:t\_cmd\_config(路由配置表)

名称	文本类型	
[0095] Fid//调用者标识	int(11)	自增 ID
Fisv_id//ISV 标识	varchar(32)	isv 的唯一标识
Fhospital_id//组织部门标识	varchar(32)	医院唯一标识
Fcmd//调用接口标识	varchar(64)	命令字
[0096] Furl//访问路径	varchar(256)	ISV 提供的接口地址
Fneed_signature//是否需要签名	bool	是否需要签名
Fneed_transform//是否需要格式转换	bool	是否需要做包格式转换

[0097] 基于表一中确认是否执行的指示信息,可通过ISV标识以及组织部门标识从表二中定位到相应处理阶段所需的动态代码以及相关的参数。例如,若在表一中Fneed\_signature字段确定进行签名校验,则从表二中可获取对应的动态代码实现的签名方法以及签名检验方法,还可以获取到签名需要用的私钥。

[0098] 表二:t\_signature\_config(签名配置表)

名称	文本类型	
Fid//调用者标识	int(11)	自增 ID
Fcmd//调用接口标识	varchar(64)	命令字
Fisv_id//ISV 标识	varchar(32)	ISV 的唯一标识
Fhospital_id//组织部门标识	varchar(32)	医院的唯一标识, 如果为 0, 表示只按 ISV 划分
[0099] Fsignature_func//签名函数	varchar(4096)	Python 动态代码实现的签名方法
Fsignature_check//签名检验函数	varchar(4096)	Python 动态代码实现的签名检验方法
Fconfig//参数	varchar(512)	json 串, 常见的存的可能是签名需要用的私钥, 由签名函数使用

[0100] 实施时, 本申请实施例中签名函数的原型可定义如下:

[0101] def signature(json\_req, json\_sig\_conf), 表示生成请求签名;

[0102] return (ret, json\_sig), 表示签名结果。

[0103] 其中, json\_req 是标准的请求包 json 串; json\_sig\_conf 是 t\_signature\_config 里配置的 Fconfig; ret 是签名结果, 0 表示成功, 非 0 表示失败; json\_sig 是签名返回回来的 json 串, 例如 {"token": "xxx"}。

[0104] 本申请实施例中签名检验函数的原型定义如下:

[0105] def signature\_check(json\_sig, json\_sig\_conf), 表示进行签名检验;

[0106] return ret, 表示签名检验结果。

[0107] 其中, json\_sig 是从返回包里取出来的签名较验需要的信息的 json 串; json\_sig\_conf 是 t\_signature\_config 里配置的 Fconfig; ret 是签名校验结果, 0 表示成功, 非 0 表示失败。

[0108] 同理, 针对包格式转换的各处理阶段, 可通过表三获取相应的动态代码来实现。也即, 在表三中通过 ISV 标识和组织部门标识可唯一定位到相应的包格式转换的动态代码。如果只需要按照业务区分不同的包格式转换时, 医院的标识可定义为特殊的标识

[0109] , 例如为 0。这样, 不同医院的同一业务可以通过一张格式配置表实现。

[0110] 表三: t\_pack\_transform (格式配置表)

	名称	文本类型	
	Fid//调用者标识	int(11)	自增 ID
	Fcmd//调用接口标识	varchar(64)	命令字
	Fisv_id//ISV 标识	varchar(32)	isv 的唯一标识
[0111]	Fhospital_id//组织部门标识	varchar(32)	医院的唯一标识, 如果为 0, 表示只按 ISV 划分
	Fproxy2isv_func// 请求包格式转换函数	varchar(4096)	Python 动态代码实现的请求 ISV 的函数参数转换
	Fisv2proxy_func//响应包格式转换函数	varchar(4096)	Python 动态代码实现的 ISV 返回包的函数参数转换
[0112]			

[0113] 可选的, 本申请实施例中包括请求包和响应包各自的格式转换, 其中:

[0114] 1)、请求包格式转换函数的原型可定义如下:

[0115] def proxy2isv(json\_req,src\_url,json\_sig),表示进行格式转换;

[0116] return(ret,json\_dest\_req,dest\_url,header),表示格式转换结果。

[0117] 其中,json\_req是标准的请求包json串;src\_url是t\_cmd\_config里配置的Furl字段;json\_sig是签名返回回来的json串,比如{"token":"xxx"};ret是包转换结果,0表示成功,非0表示失败;json\_dest\_req是请求包转换逻辑处理完的json串;dest\_url是请求包转换逻辑处理完的URL(比如在URL参数里加上token=xxxx);header,是包请求时需要设置的http头部json串,若未使用,可以直接返回"{}"。

[0118] 2)、响应包格式转换函数的原型可定义如下:

[0119] def isv2proxy(json\_resp,json\_header),表示进行格式转换;

[0120] return(ret,json\_sig,json\_dest\_resp),表示返回的格式转换结果。

[0121] 其中,json\_resp是ISV返回的响应包;json\_header是ISV返回的http头部信息json串,目前未使用,传的是"{}";ret响应包包转换结果,0表示成功,非0表示失败;json\_dest\_resp是响应包格式转换之后的内部标准包的json,如图1中中间服务器内部标准的包。

[0122] 上述方法,以不同的配置表的形式存储配置文件,可以通过更新不同的配置表达到仅更新部分配置表内容的目的,方便维护。

[0123] 具体实施时,本申请实施例中的配置文件还可以包括权限配置表。其中,权限配置表用于对业务请求的调用者进行权限认证,表中存储的信息如表四所示:

[0124] 表四:t\_app\_permission(权限配置表)

	名称	文本类型	
	Fid//调用者标识	int(11)	自增 ID
[0125]	Fapp_id//程序标识	int(11)	app_id
	Fcmd//调用者接口标识	varchar(64)	有权限的 cmd, 一条记录就是一个命令字的权限

[0126] 下面结合表四,对本申请实施例中对业务请求的调用者标识进行权限认证的方法进行说明:

[0127] 在一个可能的实施例中,后台服务器在接收到客户端发送的业务请求之后,还需要对业务请求进行鉴权。

[0128] 具体可以实施为,在所述配置文件中,查找所述业务请求对应的调用者标识和调用接口标识。根据所述调用者标识和所述调用接口标识,确定所述调用者标识对应的调用者具有所述ISV标识对应的接口的访问权限,则继续执行所述根据所述ISV标识和所述组织部门标识,获取对应的配置文件的步骤。

[0129] 例如,后台服务器接收到业务请求之后,根据ISV标识和组织部门标识,获取对应的路由配置表(表一)。并,在路由配置表中获取Fcmd字段对应的命令字。以及,根据ISV标识和组织部门标识获取对应的权限配置表(表四)。并,在权限配置表中查找前述命令字,若查找到则确定该业务请求具有调用接口标识的访问权限。

[0130] 上述方法,通过对业务请求进行鉴权,能够避免ISV提供的接口被恶意访问,保证业务的安全性,亦或者根据实际的需求确定哪些调用者可以调用相应的接口。

[0131] 在对业务请求进行权限认证后,后台服务器根据ISV标识和组织部门标识,获取对应的路由配置表(表一)。并,在路由配置表中Fneed\_signature字段和Fneed\_transform字段,确定该业务请求至少一个可配置的处理阶段的指示信息。

[0132] 具体的,至少一个可配置的处理阶段可以是签名处理阶段或签名校验处理阶段。若路由配置表中指示信息为执行相应处理阶段,则获取ISV标识和组织部门标识对应的签名配置表。并,在签名配置表(表二)中查找签名处理阶段或签名校验处理阶段对应的动态代码,根据查找到的动态代码执行相应的处理操作,动态代码执行时所需的一些可配置参数,如二中的私钥可从相应的表中读取。

[0133] 例如,若业务请求对应的路由配置表中Fneed\_signature字段对应于需要做签名,则获取该业务请求对应的签名配置表。并,根据签名配置表中Fsignature\_func字段对应的python动态代码实现的签名方法对业务请求进行签名。

[0134] 或者,至少一个可配置的处理阶段还可以包括请求包格式转换阶段或响应包格式转换阶段。若指示信息为执行相应处理阶段,则根据ISV标识和组织部门标识获取对应的格式配置表(表三)。并,在格式配置表中查找请求包格式转换阶段或响应包格式转换阶段的动态代码。根据查找到的动态代码执行相应的处理操作。

[0135] 上述方法,通过不同的配置表分别维护各处理阶段的动态代码,易于维护。例如,在业务请求的签名方法发生变化时,仅更新该业务请求对应的签名配置表即可,不需要修改全部配置文件。

[0136] 为了实现灵活配置ISV的配置文件,本申请实施例中可以在配置界面中对配置文件进行配置。具体的,显示配置界面;在所述配置界面中,配置ISV标识和组织部门标识的至少一个可配置的处理阶段对应的指示信息至所述配置文件中;配置至少一个可配置的处理阶段的动态代码至所述配置文件中。

[0137] 参阅图3,为本申请实施例中的配置界面示意图。

[0138] 具体实施时,若可配置的处理阶段的动态代码为已存在的动态代码,则可以在配置界面中选择动态代码标识。例如,生成业务请求的请求签名的方法为已存在的方法,则在配置签名处理阶段和签名校验处理阶段时,在配置界面中选择python动态代码标识。

[0139] 如图3中的a图所示,为新增的ISV业务配置配置文件时,各可配置的处理阶段的动态代码为已存在的动态代码,则在配置界面中选择各处理阶段对应的动态代码标识配置该新增ISV业务配置文件。

[0140] 或者,若可配置的处理阶段的动态代码为新的动态代码,则在配置界面中可配置动态代码至配置文件中。例如,业请求包格式转换阶段和响应包格式转换阶段的动态代码为新的动态代码,则在配置界面中配置相应的自定义的动态代码至配置文件中,并生成该动态代码的标识,以及存储动态代码标识和动态代码的对应关系。

[0141] 如图3中的b图所示,为新增的ISV业务配置配置文件时,签名处理阶段的动态代码为新的动态代码,则在配置界面中点击新增按钮,并在悬浮框中生成签名处理阶段对应的动态代码。用户点击完成按钮,配置界面显示如图3中的c图所示的悬浮框,用户在该悬浮框输入生成的动态代码的标识。

[0142] 上述方法,通过配置界面配置并更新配置文件,能够灵活的更新ISV业务对应的配置文件。

[0143] 下面通过具体的场景对本申请提供的技术方案做进一步说明。本申请实施例中组织部门标识为医院标识,业务请求为预约挂号。参阅图4,为本申请实施例中一种数据处理方法的时序图,包括以下步骤:

[0144] 步骤401:客户端发送预约挂号的业务请求给后台服务器,执行步骤402。

[0145] 其中,所述业务请求中携带有ISV标识和医院标识以及app\_id。

[0146] 步骤402:后台服务器根据app\_id获取该业务请求的token信息,执行步骤403。

[0147] 步骤403:后台服务器判断token信息是否有效,若有效则执行步骤404;若无效则执行步骤426。

[0148] 步骤404:后台服务器根据ISV标识和医院标识,获取对应的路由配置表,执行步骤405。

[0149] 步骤405:后台服务器在获取的路由配置表中,查找Fcmd字段对应的命令字,以及Fneed\_signature字段对应的指示信息和Fneed\_transform字段对应的指示信息,执行步骤406。

[0150] 步骤406:后台服务器根据ISV标识和医院标识,获取对应的权限配置表,执行步骤407。

[0151] 步骤407:后台服务器判断权限配置表中是否包含有该业务请求的命令字,若包含则执行步骤408;若不包含,则执行步骤425。

[0152] 需要说明的是,步骤405中Fneed\_signature字段和Fneed\_transform字段的查找

操作也可以在步骤407中确定具有访问权限之后执行。

[0153] 步骤408:后台服务器确定Fneed\_signature字段对应的指示信息和Fneed\_transform字段对应的指示信息均为执行相应处理阶段,执行步骤409。

[0154] 步骤409:后台服务器根据ISV标识和医院标识获取对应的签名配置表和格式配置表,执行步骤410。

[0155] 步骤410:后台服务器根据签名配置表中Fsignature\_func字段对应的python动态代码实现的签名方法,生成业务请求的请求签名,执行步骤411。

[0156] 步骤411:后台服务器根据格式配置表中Fproxy2isv\_func字段对应的python动态代码实现的请求包ISV的函数参数转换,将请求包的格式转换为ISV的提供的接口的期望格式,执行步骤412。

[0157] 例如,将请求包的TCP (Transmission Control Protocol,传输控制协议) 格式转换为UDP (User Datagram Protocol,用户数据包协议) 格式。

[0158] 步骤412:后台服务器访问路由配置表中的Furl字段对应的ISV接口的地址,执行步骤413。

[0159] 步骤413:后台服务器通过ISV接口将进行格式转换后的请求包以及请求签名发送给医院标识对应的业务处理设备,执行步骤414。

[0160] 步骤414:业务处理设备对该业务请求进行处理,完成预约挂号,执行步骤415。

[0161] 步骤415:业务处理设备将响应包发送给后台服务器,执行步骤416。

[0162] 步骤416:后台服务器根据响应包携带的ISV标识和医院标识,获取对应的路由配置表,执行步骤417。

[0163] 步骤417:后台服务器在路由配置表中查找Fneed\_signature字段对应的指示信息和Fneed\_transform字段对应的指示信息,执行步骤418。

[0164] 步骤418:后台服务器确定Fneed\_signature字段对应的指示信息和Fneed\_transform字段对应的指示信息均为执行相应处理阶段,执行步骤419。

[0165] 实施时,步骤405和步骤408中关于Fneed\_signature字段的查找和确认操作可仅执行一次,即可以不执行步骤417-步骤418,根据步骤408中对Fneed\_signature字段的指示信息执行后续操作。

[0166] 步骤419:后台服务器根据ISV标识和医院标识,获取对应的签名配置表和格式配置表,执行步骤420。

[0167] 实施时,步骤419和步骤409中签名配置表和格式配置表的获取可执行一次。当然,若签名和校验签名的签名配置表不同,和/或,若请求包和响应包的格式配置表不同则可分别执行步骤409和步骤419获取不同处理节点的相应配置表。

[0168] 步骤420:后台服务器根据格式配置表中Fisv2proxy\_func字段对应的python实现的ISV返回包的函数参数转换,将响应包的格式转换为内部期望格式,执行步骤421。

[0169] 内部期望格式可为客户端期望的格式,例如,将响应包的UDP格式转换为TCP格式。

[0170] 步骤421:后台服务器根据签名配置表中Fsignature\_check字段对应的python动态代码实现的签名检验方法,对响应包的签名进行校验,执行步骤422。

[0171] 步骤422:后台服务器将确认校验通过,执行步骤423。

[0172] 步骤423:后台服务器将响应包的报文返回给客户端,执行步骤424。

- [0173] 步骤424:客户端显示报文。
- [0174] 步骤425:后台服务器拒绝该业务请求,执行步骤426。
- [0175] 步骤426:客户端显示无法访问的信息。
- [0176] 基于相同的发明构思,本申请提供一种数据处理装置。参阅图5,为申请实施例中一种数据处理装置示意图,包括:
- [0177] 接收模块501,用于接收客户端发送的业务请求,所述业务请求中携带有独立软件开发商ISV标识和组织部门标识,所述业务请求的处理操作包括至少一个可配置的处理阶段;
- [0178] 获取模块502,用于根据所述ISV标识和所述组织部门标识,获取对应的配置文件;
- [0179] 所述获取模块502还用于,从获取的配置文件中,获得所述至少一个可配置的处理阶段中处理阶段是否执行的指示信息;
- [0180] 所述获取模块502还用于,针对所述至少一个可配置的处理阶段,在所述指示信息为执行该处理阶段的操作时,从所述配置文件中获取该处理阶段对应的动态代码;并,
- [0181] 处理模块503,用于根据查找到的动态代码,执行该处理阶段相应的处理操作。
- [0182] 可选的,所述组织部门标识为医院标识。
- [0183] 可选的,所述至少一个可配置的处理阶段为根据不同业务类型的ISV接口的处理差异确定的。
- [0184] 可选的,所述至少一个可配置的处理阶段包括以下中的至少一种:
- [0185] 签名处理阶段、签名检验处理阶段、请求包格式转换阶段以及响应包格式转换阶段,其中,所述请求包为调用ISV接口发送的数据包,所述响应包为从所述ISV接口接收到的数据包。
- [0186] 可选的,所述获取模块502还用于,在处理完签名处理阶段和请求包格式转换阶段的处理操作之后,在所述配置文件中,获取所述ISV标识对应的ISV接口的访问地址;
- [0187] 所述装置还包括:
- [0188] 发送模块,用于通过所述访问地址调用所述ISV接口将格式转换后的请求包以及签名,发送给组织部门标识对应的业务处理设备。
- [0189] 可选的,所述装置还包括:
- [0190] 查找模块,用于接收终端发送的业务请求之后,在所述配置文件中,查找所述业务请求对应的调用者标识和调用接口标识;
- [0191] 确定模块,用于根据所述调用者标识和所述调用接口标识,确定所述调用者标识对应的调用者具有所述ISV标识对应的接口的访问权限,继续执行所述根据所述ISV标识和所述组织部门标识,获取对应的配置文件的步骤。
- [0192] 可选的,所述配置文件包括路由配置表、签名配置表和格式配置表;
- [0193] 所述查找模块具体用于:在所述路由配置表中,查找所述指示信息;
- [0194] 若所述至少一个可配置的处理阶段为签名处理阶段或签名校验处理阶段,若指示信息为执行相应处理阶段,则所述获取模块502具体用于:
- [0195] 根据所述ISV标识和所述组织部门标识获取对应的签名配置表;
- [0196] 在所述签名配置表中,查找所述签名处理阶段或所述签名校验处理阶段的动态代码;

[0197] 若所述至少一个可配置的处理阶段为请求包格式转换阶段或响应包格式转换阶段,若指示信息为执行相应处理阶段,则所述获取模块502具体用于:

[0198] 根据所述ISV标识和所述组织部门标识获取对应的格式配置表;

[0199] 在所述格式配置表中,查找所述请求包格式转换阶段或响应包格式转换阶段的动态代码。

[0200] 可选的,所述装置还包括:

[0201] 显示模块,用于显示配置界面;

[0202] 配置模块,用于在所述配置界面中,配置ISV标识和组织部门标识的至少一个可配置的处理阶段对应的指示信息至所述配置文件中;

[0203] 所述配置模块还用于,配置至少一个可配置的处理阶段的动态代码至所述配置文件中。

[0204] 在介绍了本申请示例性实施方式的一种数据处理方法和装置之后,接下来,介绍根据本申请的另一示例性实施方式的电子设备。

[0205] 所属技术领域的技术人员能够理解,本申请的各个方面可以实现为系统、方法或程序产品。因此,本申请的各个方面可以具体实现为以下形式,即:完全的硬件实施方式、完全的软件实施方式(包括固件、微代码等),或硬件和软件方面结合的实施方式,这里可以统称为“电路”、“模块”或“系统”。

[0206] 在一些可能的实施方式中,根据本申请的电子设备可以至少包括至少一个处理器、以及至少一个存储器。其中,存储器存储有程序代码,当程序代码被处理器执行时,使得处理器执行本说明书上述描述的根据本申请各种示例性实施方式的数据处理方法中的步骤。例如,处理器可以执行如图2中所示的步骤201-204或者如图4所示的步骤401-426。

[0207] 下面参照图6来描述根据本申请的这种实施方式的电子设备130。图6显示的电子设备130仅仅是一个示例,不应对本申请实施例的功能和使用范围带来任何限制。

[0208] 如图6所示,电子设备130以通用电子设备的形式表现。电子设备130的组件可以包括但不限于:上述至少一个处理器131、上述至少一个存储器132、连接不同系统组件(包括存储器132和处理器131)的总线133。

[0209] 总线133表示几类总线结构中的一种或多种,包括存储器总线或者存储器控制器、外围总线、处理器或者使用多种总线结构中的任意总线结构的局域总线。

[0210] 存储器132可以包括易失性存储器形式的可读介质,例如随机存取存储器(RAM) 1321和/或高速缓存存储器1322,还可以进一步包括只读存储器(ROM) 1323。

[0211] 存储器132还可以包括具有一组(至少一个)程序模块1324的程序/实用工具1325,这样的程序模块1324包括但不限于:操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0212] 电子设备130也可以与一个或多个外部设备134(例如键盘、指向设备等)通信,还可与一个或者多个使得用户能与电子设备130交互的设备通信,和/或与使得该电子设备130能与一个或多个其它电子设备进行通信的任何设备(例如路由器、调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口135进行。并且,电子设备130还可以通过网络适配器136与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,如因特网)通信。如图所示,网络适配器136通过总线133与用于电子设备130的其它模块通信。应当

理解,尽管图中未示出,可以结合电子设备130使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理器、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0213] 在一些可能的实施方式中,本申请提供的一种数据处理方法的各个方面还可以实现为一种程序产品的形式,其包括程序代码,当程序产品在计算机设备上运行时,程序代码用于使计算机设备执行本说明书上述描述的根据本申请各种示例性实施方式的一种数据处理方法中的步骤,例如,计算机设备可以执行如图2中所示的步骤201-204或者如图4所示的步骤401-426。

[0214] 程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0215] 本申请的实施方式的用于数据处理的程序产品可以采用便携式紧凑盘只读存储器(CD-ROM)并包括程序代码,并可以在电子设备上运行。然而,本申请的程序产品不限于此,在本文件中,可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0216] 可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了可读程序代码。这种传播的数据信号可以采用多种形式,包括——但不限于——电磁信号、光信号或上述的任意合适的组合。可读信号介质还可以是可读存储介质以外的任何可读介质,该可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。

[0217] 可读介质上包含的程序代码可以用任何适当的介质传输,包括——但不限于——无线、有线、光缆、RF等等,或者上述的任意合适的组合。

[0218] 可以以一种或多种程序设计语言的任意组合来编写用于执行本申请操作的程序代码,程序设计语言包括面向对象的程序设计语言—诸如Java、C++等,还包括常规的过程式程序设计语言—诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户电子设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户电子设备上部分在远程电子设备上执行、或者完全在远程电子设备或服务器上执行。在涉及远程电子设备的情形中,远程电子设备可以通过任意种类的网络——包括局域网(LAN)或广域网(WAN)—连接到用户电子设备,或者,可以连接到外部电子设备(例如利用因特网服务提供商来通过因特网连接)。

[0219] 应当注意,尽管在上文详细描述中提及了装置的若干单元或子单元,但是这种划分仅仅是示例性的并非强制性的。实际上,根据本申请的实施方式,上文描述的两个或更多单元的特征和功能可以在一个单元中具体化。反之,上文描述的一个单元的特征和功能可以进一步划分为由多个单元来具体化。

[0220] 此外,尽管在附图中以特定顺序描述了本申请方法的操作,但是,这并非要求或者暗示必须按照该特定顺序来执行这些操作,或是必须执行全部所示的操作才能实现期望的

结果。附加地或备选地,可以省略某些步骤,将多个步骤合并为一个步骤执行,和/或将一个步骤分解为多个步骤执行。

[0221] 本领域内的技术人员应明白,本申请的实施例可提供为方法、系统、或计算机程序产品。因此,本申请可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0222] 本申请是参照根据本申请实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0223] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0224] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0225] 尽管已描述了本申请的优选实施例,但本领域内的技术人员一旦得知了基本创造性概念,则可对这些实施例做出另外的变更和修改。所以,所附权利要求意欲解释为包括优选实施例以及落入本申请范围的所有变更和修改。

[0226] 显然,本领域的技术人员可以对本申请进行各种改动和变型而不脱离本申请的精神和范围。这样,倘若本申请的这些修改和变型属于本申请权利要求及其等同技术的范围之内,则本申请也意图包含这些改动和变型在内。

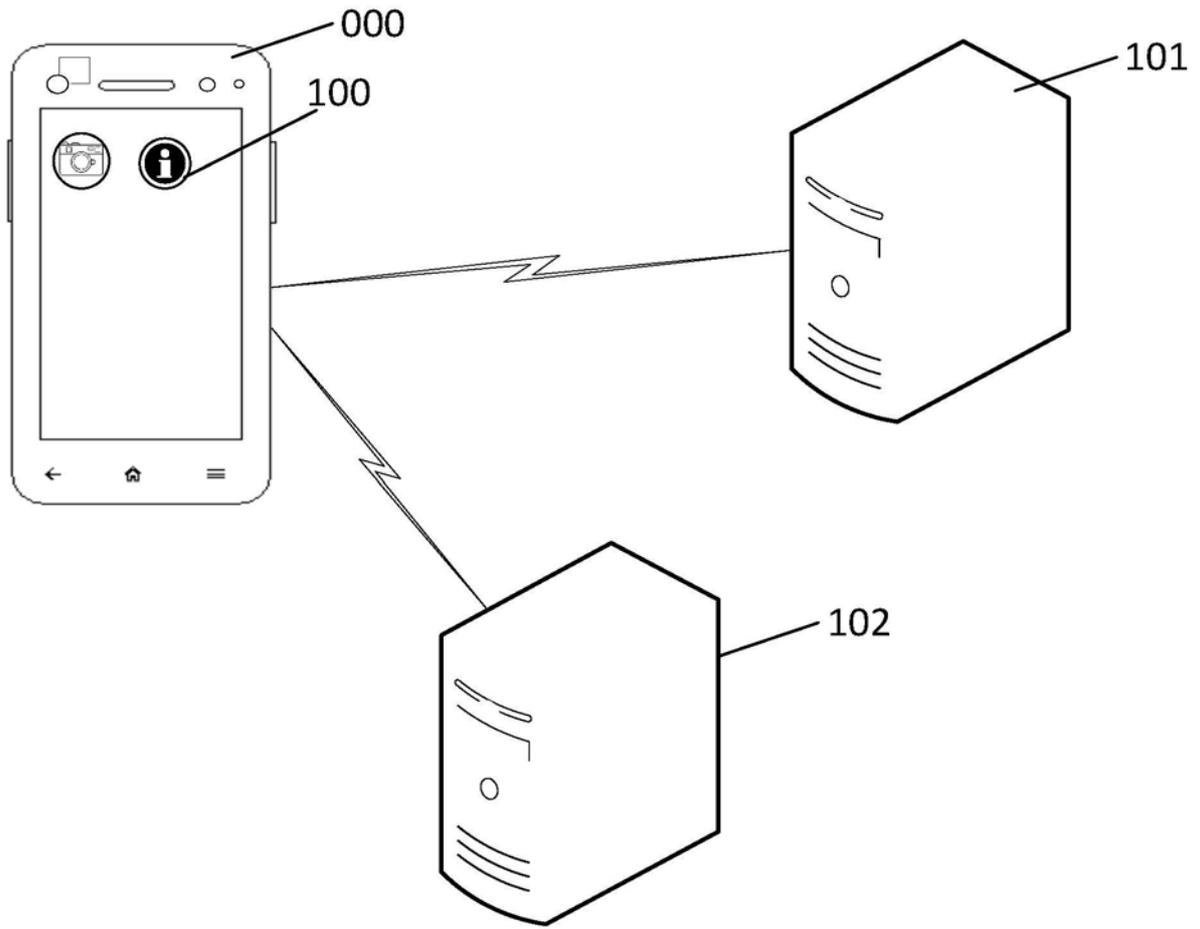


图1

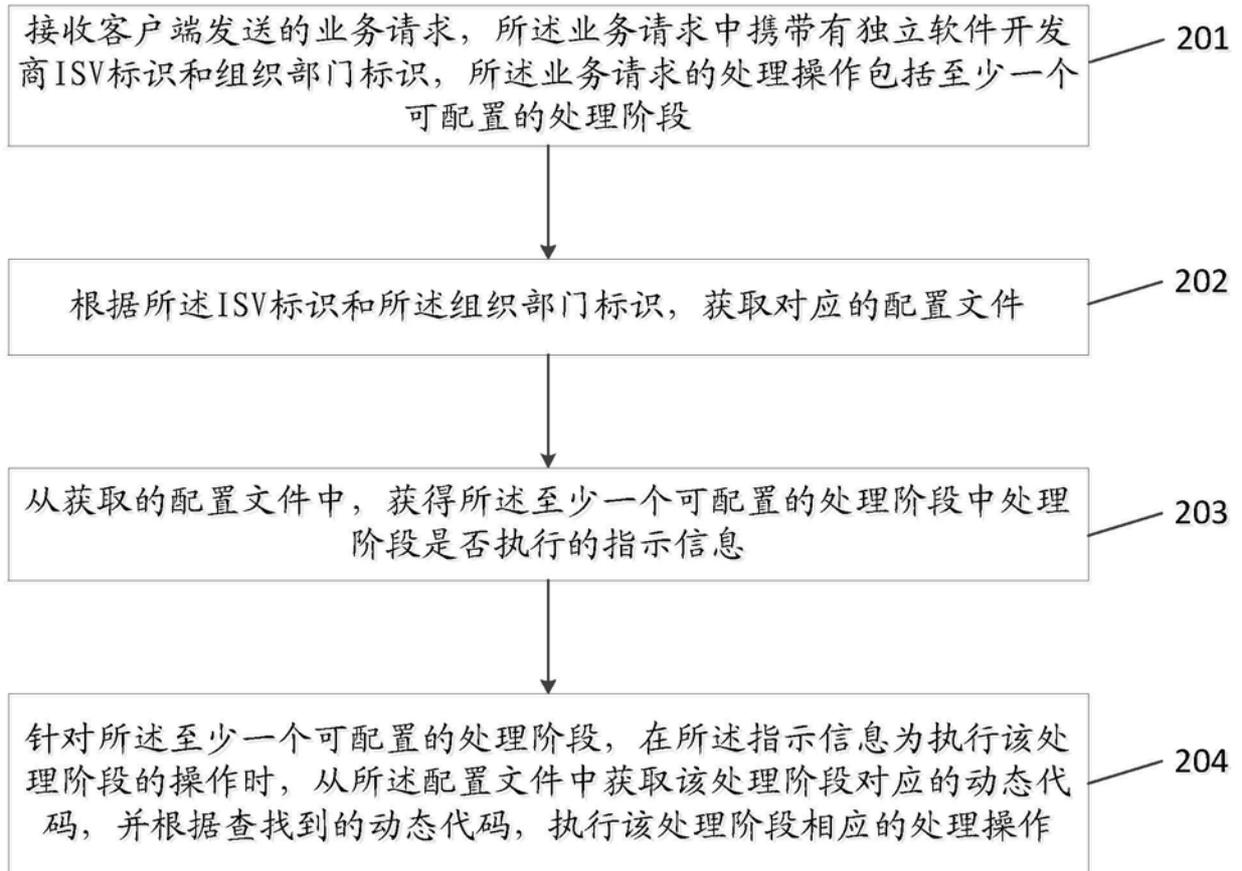


图2

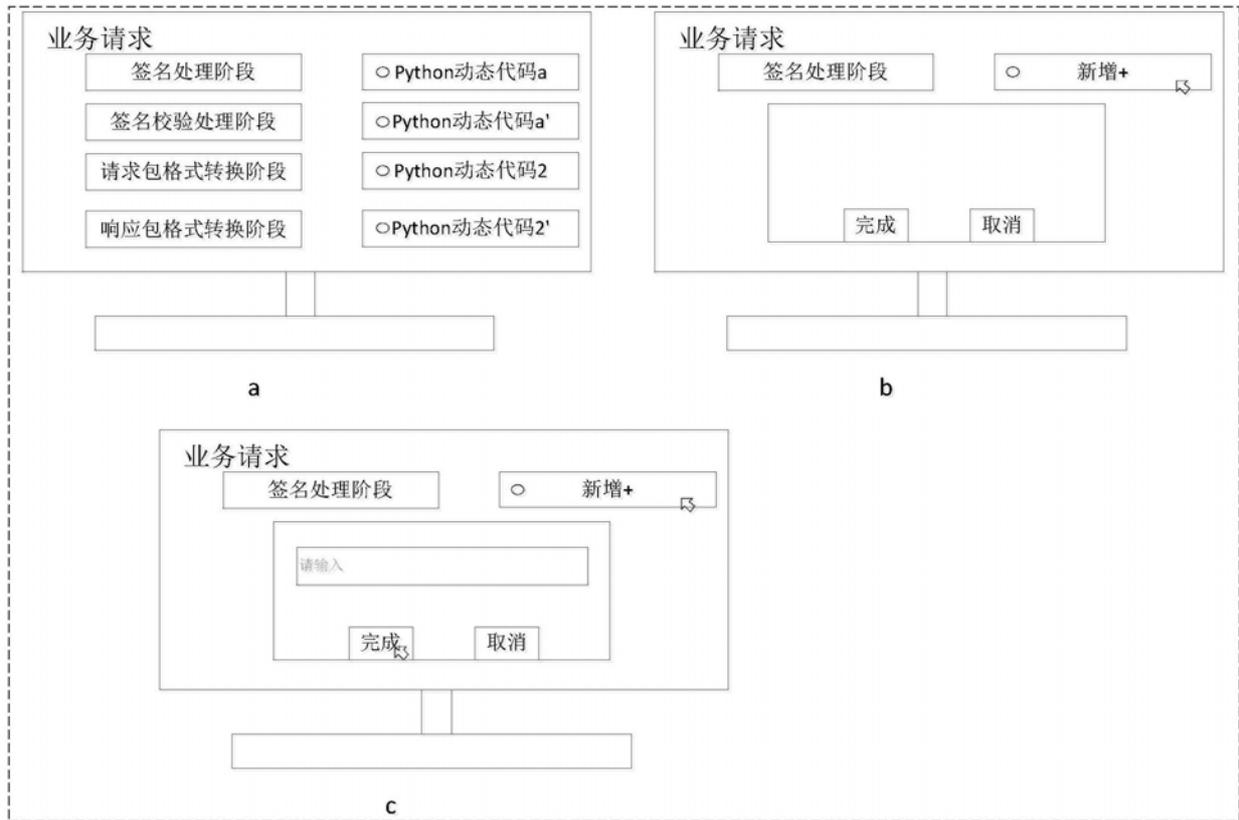


图3

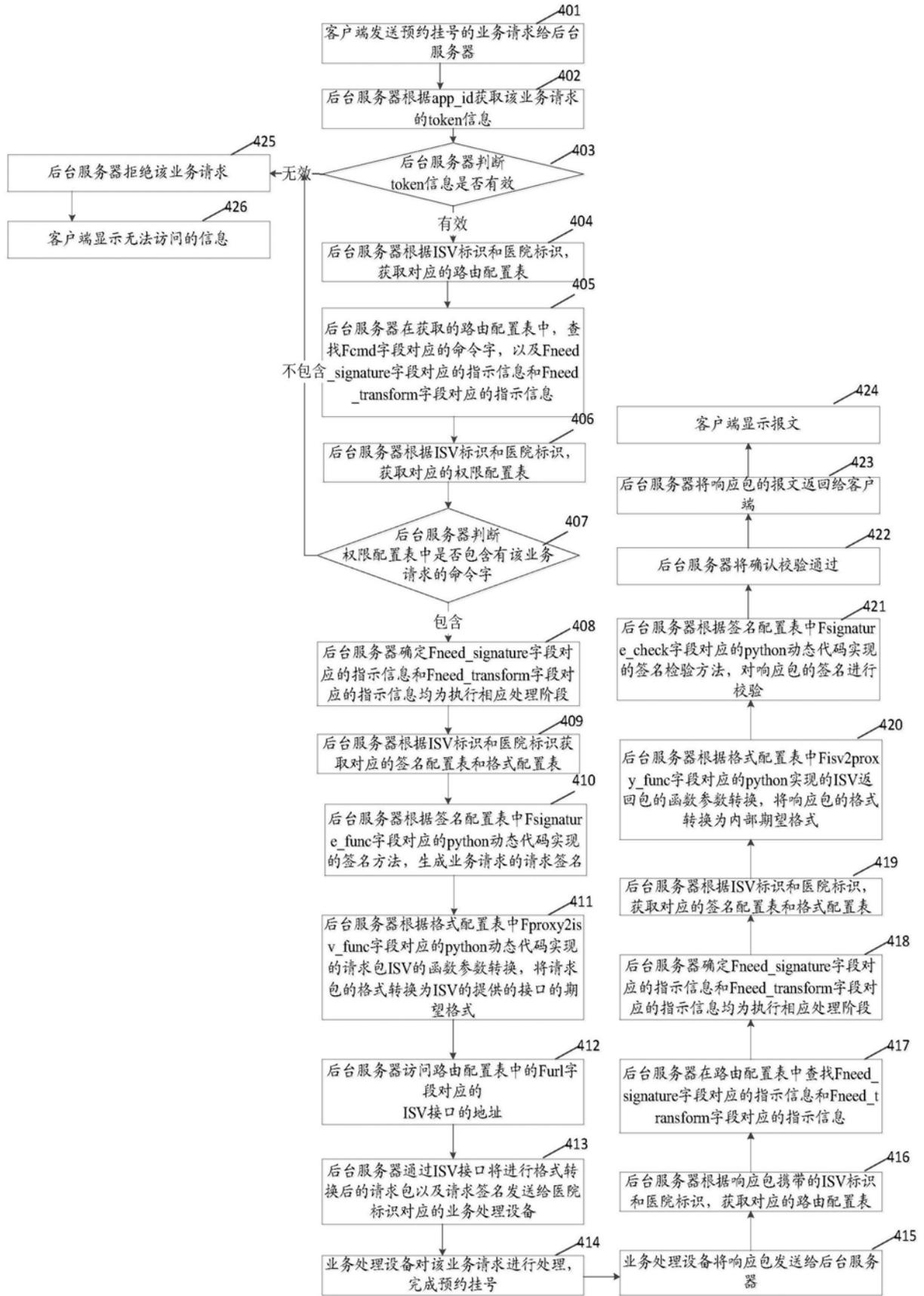


图4

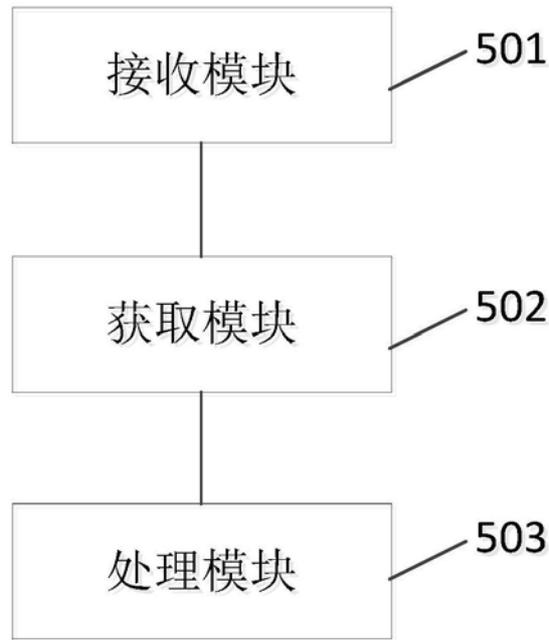


图5

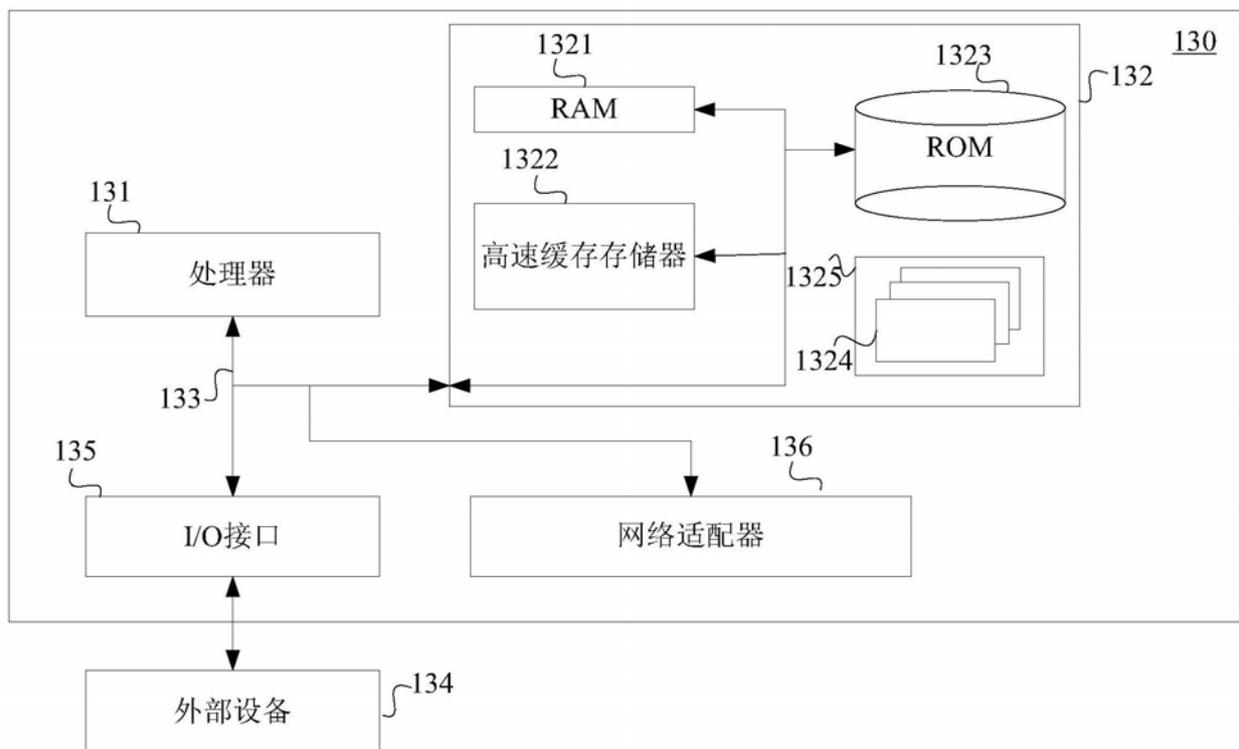


图6