



(12)发明专利申请

(10)申请公布号 CN 108494876 A

(43)申请公布日 2018.09.04

(21)申请号 201810324963.5

(22)申请日 2018.04.12

(71)申请人 北京达佳互联信息技术有限公司

地址 100084 北京市海淀区中关村东路1号
院8号楼20层B2201

(72)发明人 刘硕

(74)专利代理机构 北京市立方律师事务所

11330

代理人 刘延喜

(51)Int.Cl.

H04L 29/08(2006.01)

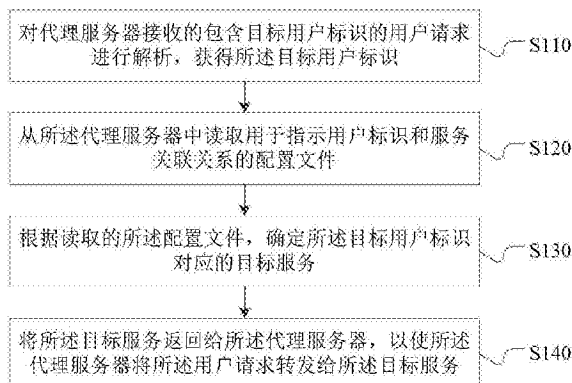
权利要求书2页 说明书14页 附图4页

(54)发明名称

实现服务灰度发布的方法、装置、插件设备和代理服务器

(57)摘要

本发明提供一种实现服务灰度发布的方法、装置、插件设备和代理服务器,以解决现有技术方案可用性差的技术问题。所述的方法包括步骤:对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;根据读取的所述配置文件,确定所述目标用户标识对应的目标服务;将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。本发明实施例可用性强。



1. 一种实现服务灰度发布的方法,其特征在于,包括步骤:

对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;

从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;

根据读取的所述配置文件,确定所述目标用户标识对应的目标服务;

将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。

2. 根据权利要求1所述的实现服务灰度发布的方法,其特征在于,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

判断所述代理服务器与zookeeper是否是第一次建立连接,其中,所述zookeeper存储有所述配置文件;

若是,从所述zookeeper中读取所述配置文件,并将所述配置文件存储在所述代理服务器中;

若否,从所述代理服务器中读取所述配置文件。

3. 根据权利要求1所述的实现服务灰度发布的方法,其特征在于,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

判断所述代理服务器是否接收到zookeeper发送的配置文件变更通知;

若是,从所述zookeeper中读取变更后的配置文件,并根据变更后的配置文件对存储在所述代理服务器中的配置文件进行更新;

若否,从所述代理服务器中读取所述配置文件。

4. 根据权利要求1至3任意一项所述的实现服务灰度发布的方法,其特征在于,所述配置文件包括白名单、对所述目标用户标识取模所采用的设定值以及命中范围;

所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

判断所述目标用户标识是否在所述白名单内;

若是,将新服务作为目标服务;

若否,采用所述设定值对所述目标用户标识取模,若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

5. 一种实现服务灰度发布的方法,其特征在于,包括步骤:

将接收的包含目标用户标识的用户请求发送给插件进行解析,以获得目标用户标识;

将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件,以确定所述目标用户标识对应的目标服务;

接收所述插件返回的所述目标服务;

将所述用户请求转发给所述目标服务。

6. 一种实现服务灰度发布的装置,其特征在于,包括:

用户请求解析模块,用于对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;

配置文件读取模块,用于从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;

目标服务确定模块,用于根据读取的所述配置文件,确定所述目标用户标识对应的目

标服务；

确定结果发送模块，用于将所述目标服务返回给所述代理服务器，以使所述代理服务器将所述用户请求转发给所述目标服务。

7. 一种实现服务灰度发布的装置，其特征在于，包括：

用户请求发送模块，用于将接收的包含目标用户标识的用户请求发送给插件进行解析，以获得目标用户标识；

配置文件发送模块，用于将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件，以确定所述目标用户标识对应的目标服务；

确定结果接收模块，用于接收所述插件返回的所述目标服务；

用户请求转发模块，用于将所述用户请求转发给所述目标服务。

8. 一种计算机可读存储介质，其上存储有计算机程序，其特征在于，该程序被处理器执行时实现如权利要求1至5中任意一项所述的实现服务灰度发布的方法。

9. 一种插件设备，其特征在于，所述插件设备包括：

一个或多个处理器；

存储装置，用于存储一个或多个程序，

当所述一个或多个程序被所述一个或多个处理器执行，使得所述一个或多个处理器实现如权利要求1至4中任意一项所述的实现服务灰度发布的方法。

10. 一种代理服务器，其特征在于，所述代理服务器包括：

一个或多个处理器；

存储装置，用于存储一个或多个程序，

当所述一个或多个程序被所述一个或多个处理器执行，使得所述一个或多个处理器实现如权利要求5所述的实现服务灰度发布的方法。

实现服务灰度发布的方法、装置、插件设备和代理服务器

技术领域

[0001] 本发明涉及计算机技术领域,具体而言,本发明涉及一种实现服务灰度发布的方法、装置、插件设备和代理服务器。

背景技术

[0002] 灰度发布是指在黑与白之间,能够平滑过渡的一种发布方式。AB test就是一种灰度发布方式,让一部分用户继续用A,一部分用户开始用B,如果用户对B没有什么反对意见,那么逐步扩大范围,把所有用户都迁移到B上面来。灰度发布可以保证整体系统的稳定,在初始灰度的时候就可以发现、调整问题,以保证其影响度。

[0003] 传统技术中,实现灰度发布的方案包括Nginx根据权重或ip_hash来进行分配,该方案配置简单,直接在Nginx配置文件中配置,具体如下:

```
[0004] upstream backend{  
[0005]     server 192.168.0.14weight=10;  
[0006]     server 192.168.0.15weight=10;  
[0007] }或  
[0008] upstream backend{  
[0009]     ip_hash;  
[0010]     server 192.168.0.14:88;  
[0011]     server 192.168.0.15:80;  
[0012] }
```

[0013] 然而,本发明的发明人注意到,上述根据权重或ip_hash实现灰度发布的方案存在可用性差的缺陷。

发明内容

[0014] 本发明针对现有方式的缺点,提出一种实现服务灰度发布的方法、装置、插件设备和代理服务器,以实现可用性强的服务灰度发布方案。

[0015] 本发明的实施例根据第一个方面,提供了一种实现服务灰度发布的方法,包括步骤:

[0016] 对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;

[0017] 从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;

[0018] 根据读取的所述配置文件,确定所述目标用户标识对应的目标服务;

[0019] 将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。

[0020] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至

该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0021] 在一个实施例中,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

[0022] 判断所述代理服务器与zookeeper是否是第一次建立连接,其中,所述zookeeper存储有所述配置文件;

[0023] 若是,从所述zookeeper中读取所述配置文件,并将所述配置文件存储在所述代理服务器中;

[0024] 若否,从所述代理服务器中读取所述配置文件。

[0025] 本实施例将配置文件放在zookeeper中,代理服务器中存储备份的配置文件,灵活性更好。而且相较于每次接收到用户请求都去内存数据库中查询数据的方案,本实施例只需在第一次连接时从zookeeper获取配置文件,简化了操作,节省了资源,且无需灌数据。

[0026] 在一个实施例中,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

[0027] 判断所述代理服务器是否接收到zookeeper发送的配置文件变更通知;

[0028] 若是,从所述zookeeper中读取变更后的配置文件,并根据变更后的配置文件对存储在所述代理服务器中的配置文件进行更新;

[0029] 若否,从所述代理服务器中读取所述配置文件。

[0030] 本实施例利用zookeeper机制,在zookeeper存储的配置文件变更时才通知代理服务器,才从zookeeper中获取变更后的配置文件,相较于通过内存数据库每次变更需要清理数据而导致的操作麻烦且笨重的缺陷,本实施例操作简单轻便。

[0031] 在一个实施例中,所述配置文件包括白名单;

[0032] 所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0033] 判断所述目标用户标识是否在所述白名单内;

[0034] 若是,将新服务作为目标服务,否则将旧服务作为目标服务。

[0035] 在一个实施例中,所述配置文件包括对所述目标用户标识取模所采用的设定值以及命中范围;

[0036] 所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0037] 采用所述设定值对所述目标用户标识取模;

[0038] 若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0039] 在一个实施例中,所述配置文件包括白名单、对所述目标用户标识取模所采用的设定值以及命中范围;

[0040] 所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0041] 判断所述目标用户标识是否在所述白名单内;

[0042] 若是,将新服务作为目标服务;

[0043] 若否,采用所述设定值对所述目标用户标识取模,若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0044] 本发明的实施例根据第二个方面,还提供了另一种实现服务灰度发布的方法,包括步骤:

[0045] 将接收的包含目标用户标识的用户请求发送给插件进行解析,以获得目标用户标识;

[0046] 将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件,以确定所述目标用户标识对应的目标服务;

[0047] 接收所述插件返回的所述目标服务;

[0048] 将所述用户请求转发给所述目标服务。

[0049] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0050] 在一个实施例中,所述本地存储的配置文件为,与zookeeper第一次建立连接时,从所述zookeeper中获取的配置文件。

[0051] 本实施例将配置文件放在zookeeper中,代理服务器中存储备份的配置文件,灵活性更好。而且相较于每次接收到用户请求都去内存数据库中查询数据的方案,本实施例只需在第一次连接时从zookeeper获取配置文件,简化了操作,节省了资源,且无需灌数据。

[0052] 在一个实施例中,所述本地存储的配置文件为,在接收到zookeeper发送的配置文件变更通知时,从所述zookeeper中获取的变更后的配置文件。

[0053] 本实施例利用zookeeper机制,在zookeeper存储的配置文件变更时才通知代理服务器,才从zookeeper中获取变更后的配置文件,相较于通过内存数据库每次变更需要清理数据而导致的操作麻烦且笨重的缺陷,本实施例操作简单轻便。

[0054] 本发明的实施例根据第三个方面,还提供了一种实现服务灰度发布的装置,包括:

[0055] 用户请求解析模块,用于对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;

[0056] 配置文件读取模块,用于从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;

[0057] 目标服务确定模块,用于根据读取的所述配置文件,确定所述目标用户标识对应的目标服务;

[0058] 确定结果发送模块,用于将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。

[0059] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0060] 本发明的实施例根据第四个方面,还提供了另一种实现服务灰度发布的装置,包括:

[0061] 用户请求发送模块,用于将接收的包含目标用户标识的用户请求发送给插件进行解析,以获得目标用户标识;

[0062] 配置文件发送模块,用于将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件,以确定所述目标用户标识对应的目标服务;

[0063] 确定结果接收模块,用于接收所述插件返回的所述目标服务;

[0064] 用户请求转发模块,用于将所述用户请求转发给所述目标服务。

[0065] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0066] 本发明的实施例根据第五个方面,还提供了一种计算机可读存储介质,其上存储有计算机程序,该程序被处理器执行时实现上述任意一项所述的实现服务灰度发布的方法。

[0067] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0068] 本发明的实施例根据第六个方面,还提供了一种插件设备,所述插件设备包括:

[0069] 一个或多个处理器;

[0070] 存储装置,用于存储一个或多个程序,

[0071] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现上述任意一项所述的实现服务灰度发布的方法。

[0072] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强。

[0073] 本发明的实施例根据第七个方面,还提供了一种代理服务器,所述代理服务器包括:

[0074] 一个或多个处理器;

[0075] 存储装置,用于存储一个或多个程序,

[0076] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现上述任意一项所述的实现服务灰度发布的方法。

[0077] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务,相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,

因而可用性强。

[0078] 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

[0079] 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:

[0080] 图1为本发明一个实施例的实现服务灰度发布的方法的流程示意图;

[0081] 图2为本发明一个实施例的实现服务灰度发布的装置的结构示意图;

[0082] 图3为本发明另一个实施例的实现服务灰度发布的方法的流程示意图;

[0083] 图4为本发明另一个实施例的实现服务灰度发布的装置的结构示意图;

[0084] 图5为本发明另一个实施例的实现服务灰度发布的方法的流程示意图;

[0085] 图6为本发明另一个实施例的实现服务灰度发布的方法的流程示意图;

[0086] 图7为本发明另一个实施例的实现服务灰度发布的方法的流程示意图;

[0087] 图8为本发明一个实施例的代理服务器的结构示意图。

具体实施方式

[0088] 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。

[0089] 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“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。

[0090] 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。

[0091] 本技术领域技术人员可以理解,这里所使用的服务器,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(3rd Generation Partnership Project,第三代合作伙伴计划)、LTE(Long Term Evolution,长期演进)、WIMAX(Worldwide Interoperability for Microwave Access,全球微波互联互通)的移动通信、基于TCP/IP(Transmission Control Protocol/Internet Protocol,传输控制协议/因特网互联协议)、UDP(User Datagram Protocol,用户数据报协议)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。

[0092] 有必要先对本发明的研究背景进行如下的先导性说明。

[0093] 在实现服务灰度发布时,根据权重或ip_hash选择的发布对象是随机的,发布对象不可控,不能指定特定的用户来进行发布,因此可用性差。另外,根据权重或ip_hash实现服务灰度发布的方案,发布对象受服务器故障影响,在某一台服务器出现故障时,权重及ip_hash会受到影响,影响最终发布的效果,因此可靠性差。

[0094] 为了解决上述缺陷,在实现服务恢复发布时,还有另外一种方案,该方案通过redis或者memcache等数据库指定特定的用户来进行发布,可用性强,可靠性好。

[0095] 但是,本发明的发明人经研究发现,上述通过redis或者memcache等数据库实现服务灰度发布的方案,每次接收到用户请求时都会去redis或者memcache等数据库里查询,浪费资源,需要部署redis或者memcache等数据库,需要灌数据,每次变更需要清理数据,操作麻烦且笨重。

[0096] 因此,本发明实施例提供了一种实现服务灰度发布的方案,配置文件可以放在zookeeper中,这样灵活性更好,本地有配置文件的备份,在接收到用户请求时可以直接从本地存储的备份中查询,节省了资源,无需部署redis或者memcache等数据库,也无需灌数据,每次变更不需要清理数据,操作简单轻便。

[0097] 首先从代理服务器内置的插件的角度出发,对本发明的具体实施方式进行详细介绍。

[0098] 如图1所示,在一个实施例中,一种实现服务灰度发布的方法,包括步骤:

[0099] S110、对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识。

[0100] 代理服务器可以为Nginx,也可以为其它类型的服务器,本发明并不对此作出限定。其中,Nginx是一款轻量级的Web服务器/反向代理服务器及电子邮件(IMAP/POP3)代理服务器,并在一个BSD-like协议下发行,其特点是占有内存少,并发能力强。

[0101] 代理服务器的插件可以通过lua或者其他语言实现。其中,lua是一种可以嵌入到Nginx配置文件中的动态脚本语言,使用lua来实现Nginx插件,不会影响Nginx性能,能迅速实现请求分发。

[0102] 目标用户标识用于标识用户的身份信息,可以为uid(user identifier用户标识符),也可以为其它能够标识用户身份信息的标识。用户请求除包含目标用户标识之外,还可以包含其他信息,例如对应灰度发布的版本号等等,本发明并不对用户请求的具体内容进行限定。

[0103] 用户在客户端输入灰度发布的相关请求,客户端将该用户请求发送给前端的代理服务器。当该用户请求到达前端的代理服务器后,其内嵌的插件会去解析用户请求,以获得发送请求的用户的用户标识。

[0104] S120、从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件。

[0105] 配置文件用于描述各个用户标识和各个服务之间的关联关系,可以根据自定义策略设置,只要能实现指定特定的用户进行发布即可。配置文件存储在代理服务器中,例如代理服务器的内存中。那么插件即可以直接从代理服务器中读取该配置文件,相较于传统技术中每次都需要去数据库中读取数据进行查询的方案,该方式简化了操作,节省了资源。

[0106] S130、根据读取的所述配置文件,确定所述目标用户标识对应的目标服务。

[0107] 由于配置文件指示有用户标识对应的服务,因此根据该配置文件,就可以直接确定出目标用户标识所对应的目标服务。

[0108] S140、将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。

[0109] 插件确定目标用户标识对应的目标服务后,将确定的该目标服务的信息返回给代理服务器。随后代理服务器的Location会根据插件的返回结果将其用户请求转发给对应的服务器进行处理,实现灰度发布。

[0110] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务。相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强;发布对象不受服务器故障影响,因而可靠性强。

[0111] 在一个实施例中,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

[0112] S1201、判断所述代理服务器与zookeeper是否是第一次建立连接,其中,所述zookeeper存储有所述配置文件。

[0113] zookeeper本质上是一个用于协调分布式集群的管理服务,可以作为配置文件订阅的服务。配置文件可以放在zookeeper中,这样灵活性更好。

[0114] S1202、若是,从所述zookeeper中读取所述配置文件,并将所述配置文件存储在所述代理服务器中。

[0115] 由于在zookeeper中生成配置文件,因此如果代理服务器和zookeeper是第一次建立连接,则代理服务器中没有该配置文件,插件无法直接从代理服务器中读取该配置文件,因此插件需要从zookeeper中读取配置文件,根据读取的该配置文件确定目标用户标识对应的目标服务。为了确保后续可以直接从代理服务器中读取配置文件,在代理服务器中对该配置文件进行备份。

[0116] S1203、若否,从所述代理服务器中读取所述配置文件。

[0117] 如果代理服务器和zookeeper不是第一次建立连接,则代理服务器中已经有该配置文件的备份,因此插件可以直接从代理服务器中读取配置文件,根据该读取的配置文件确定目标用户标识对应的目标服务,不需要每次都去zookeeper查询配置文件。

[0118] 在一个实施例中,所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

[0119] S120a、判断所述代理服务器是否接收到zookeeper发送的配置文件变更通知。

[0120] zookeeper生成配置文件,当对zookeeper中的配置文件进行变更后,zookeeper就会主动向代理服务器发送配置文件变更通知。

[0121] S120b、若是,从所述zookeeper中读取变更后的配置文件,并根据变更后的配置文件对存储在所述代理服务器中的配置文件进行更新。

[0122] 如果接收到zookeeper向代理服务器发送的配置文件变更通知,说明zookeeper中的配置文件已经进行变更,代理服务器当前存储的配置文件为老版本的配置文件,因此插件需要从zookeeper中直接读取变更后的配置文件,根据读取的该配置文件确定目标用户

标识对应的目标服务。另外,还需要对代理服务器中存储的当前配置文件进行更新,以保证代理服务器存储的配置文件最新,后续可以直接从代理服务器中读取配置文件。

[0123] S120c、若否,从所述代理服务器中读取所述配置文件。

[0124] 如果没有接收到zookeeper向代理服务器发送的配置文件变更通知,说明zookeeper中的配置文件没有进行变更,代理服务器当前存储的配置文件为最新版本的配置文件,因此插件可以直接从代理服务器中读取配置文件,根据读取的该配置文件确定目标用户标识对应的目标服务。

[0125] 配置文件可以根据自定义策略进行设置,配置文件不同,确定目标用户标识对应的目标服务的方式不同。下面结合三个实施例进行介绍。

[0126] 在一个实施例中,所述配置文件包括白名单;所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0127] S1301、判断所述目标用户标识是否在所述白名单内。

[0128] 白名单(white list)中设置有能够收到发布版本的各个用户标识。配置文件中的白名单可以以“用户标识1,用户标识2……”的形式直接罗列,例如,123333,123123123,这种方式比较简洁。配置文件中的白名单也可以以“{“whiteList”:[用户标识1,用户标识2,……]}”Key-Value的形式进行设置,例如,{“whiteList”:[876572,234242]}。本发明并不对此作出限定。

[0129] S1302、若是,将新服务作为目标服务,否则将旧服务作为目标服务。

[0130] 如果白名单中有该目标用户标识,说明该用户为能够接收发布版本的用户,将代理服务器跳转的服务确定为新服务,即新上线的服务。插件就可以向代理服务器返回新服务的信息,例如返回@service_new。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service_new转发给新上线的服务。

[0131] 如果白名单中没有该目标用户标识,说明该用户为不能接收发布版本的用户,将代理服务器跳转的服务确定为旧服务,即原来的服务。插件就可以向代理服务器返回旧服务的信息,例如返回@service。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service转发给旧服务。

[0132] 在另一个实施例中,所述配置文件包括对所述目标用户标识取模所采用的设定值以及命中范围;所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0133] S130-1、采用所述设定值对所述目标用户标识取模。

[0134] 设定值表示mod函数中的modNum,具体数值可以根据用户需要进行设置。命中范围是从mod的结果中选取的子集,该子集表示命中的范围,即能够收到发布版本的数值范围,具体可以根据用户需要进行设置。

[0135] 配置文件中的设定值和命中范围可以以“modNum;tailNum range;”形式直接罗列,例如,1000;100-200。配置文件中的设定值和命中范围也可以以“{“modNum”:设定值,“range”:命中范围}”Key-Value的形式进行设置,例如,{“modNum”:1000,“range”:[100,200]}。本发明并不对此作出限定。

[0136] S130-2、若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0137] 如果取模后的尾数在所述命中范围内,说明该用户为能够接收发布版本的用户,将代理服务器跳转的服务确定为新服务,即新上线的服务。插件就可以向代理服务器返回新服务的信息,例如返回@service_new。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service_new转发给新上线的服务。

[0138] 如果取模后的尾数不在所述命中范围内,说明该用户为不能接收发布版本的用户,将代理服务器跳转的服务确定为旧服务,即原来的服务。插件就可以向代理服务器返回旧服务的信息,例如返回@service。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service转发给旧服务。

[0139] 在另一个实施例中,所述配置文件包括白名单、对所述目标用户标识取模所采用的设定值以及命中范围;所述根据读取的所述配置文件,确定所述目标用户标识对应的目标服务,包括:

[0140] S130a、判断所述目标用户标识是否在所述白名单内。

[0141] 白名单(white list)中设置有能够收到发布版本的各个用户标识。设定值表示mod函数中的modNum,具体数值可以根据用户需要进行设置。命中范围是从mod的结果中选取的子集,该子集表示命中的范围,即能够收到发布版本的数值范围,具体可以根据用户需要进行设置。白名单的优先级最高。

[0142] 配置文件中的设定值、命中范围和白名单可以以“modNum;tailNum range;white list”形式直接罗列,例如,1000;100-200;123333,123123123。配置文件中的设定值、命中范围和白名单也可以以“{“modNum”:设定值,“range”:命中范围,“whiteList”:用户标识1,用户标识2……}”Key-Value的形式进行设置,例如,{“modNum”:1000,“range”:[100,200],“whiteList”:[876572,234242]}。本发明并不对此作出限定。

[0143] S130b、若是,将新服务作为目标服务。

[0144] 如果白名单中有该目标用户标识,说明该用户为能够接收发布版本的用户,将代理服务器跳转的服务确定为新服务,即新上线的服务。插件就可以向代理服务器返回新服务的信息,例如返回@service_new。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service_new转发给新上线的服务。

[0145] S130c、若否,采用所述设定值对所述目标用户标识取模,若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0146] 如果白名单中没有该目标用户标识,对目标用户标识进行取模。如果取模后的尾数在所述命中范围内,说明该用户为能够接收发布版本的用户,将代理服务器跳转的服务确定为新服务,即新上线的服务。插件就可以向代理服务器返回新服务的信息,例如返回@service_new。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service_new转发给新上线的服务。如果取模后的尾数不在所述命中范围内,说明该用户为不能接收发布版本的用户,将代理服务器跳转的服务确定为旧服务,即原来的服务。插件就可以向代理服务器返回旧服务的信息,例如返回@service。随后代理服务器的Location会根据插件的返回转发给对应的服务器,如@service转发给旧服务。

[0147] 应当理解的是,本发明并不对配置文件的形式进行限定,用户还可以根据需要设置其他字段。例如,添加如HTTP(HyperText Transfer Protocol,超文本传输协议)的serverName,这样,如果一个代理服务器给多个服务提供前端调度,则需要根据bizName来

确定该服务的配置。

[0148] 基于同一发明构思,本发明还提供一种实现服务灰度发布的装置,下面结合附图对本发明装置的具体实施方式进行详细描述。

[0149] 如图2所示,在一个实施例中,一种实现服务灰度发布的装置,包括:

[0150] 用户请求解析模块110,用于对代理服务器接收的包含目标用户标识的用户请求进行解析,获得所述目标用户标识;

[0151] 配置文件读取模块120,用于从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件;

[0152] 目标服务确定模块130,用于根据读取的所述配置文件,确定所述目标用户标识对应的目标服务;

[0153] 确定结果发送模块140,用于将所述目标服务返回给所述代理服务器,以使所述代理服务器将所述用户请求转发给所述目标服务。

[0154] 在一个实施例中,配置文件读取模块120用于执行以下操作:

[0155] 判断所述代理服务器与zookeeper是否是第一次建立连接,其中,所述zookeeper存储有所述配置文件;

[0156] 若是,从所述zookeeper中读取所述配置文件,并将所述配置文件存储在所述代理服务器中;

[0157] 若否,从所述代理服务器中读取所述配置文件。

[0158] 在一个实施例中,配置文件读取模块120还用于执行以下操作:

[0159] 所述从所述代理服务器中读取用于指示用户标识和服务关联关系的配置文件,包括:

[0160] 判断所述代理服务器是否接收到zookeeper发送的配置文件变更通知;

[0161] 若是,从所述zookeeper中读取变更后的配置文件,并根据变更后的配置文件对存储在所述代理服务器中的配置文件进行更新;

[0162] 若否,从所述代理服务器中读取所述配置文件。

[0163] 在一个实施例中,所述配置文件包括白名单;目标服务确定模块130用于执行以下操作:

[0164] 判断所述目标用户标识是否在所述白名单内;

[0165] 若是,将新服务作为目标服务,否则将旧服务作为目标服务。

[0166] 在一个实施例中,所述配置文件包括对所述目标用户标识取模所采用的设定值以及命中范围;目标服务确定模块130还用于执行以下操作:

[0167] 采用所述设定值对所述目标用户标识取模;

[0168] 若取模后的尾数在所述命中范围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0169] 在一个实施例中,所述配置文件包括白名单、对所述目标用户标识取模所采用的设定值以及命中范围;目标服务确定模块130还用于执行以下操作:

[0170] 判断所述目标用户标识是否在所述白名单内;

[0171] 若是,将新服务作为目标服务;

[0172] 若否,采用所述设定值对所述目标用户标识取模,若取模后的尾数在所述命中范

围内,将新服务作为目标服务,否则将旧服务作为目标服务。

[0173] 上述实现服务灰度发布的装置的其它技术特征与上述实现服务灰度发布的方法的技术特征相同,在此不予赘述。

[0174] 然后从代理服务器的角度出发,对本发明的具体实施方式进行详细描述。

[0175] 如图3所示,在一个实施例中,一种实现服务灰度发布的方法,包括步骤:

[0176] S210、将接收的包含目标用户标识的用户请求发送给插件进行解析,以获得目标用户标识。

[0177] 代理服务器可以为Nginx,也可以为其它类型的服务器,本发明并不对此作出限定。其中,Nginx是一款轻量级的Web服务器/反向代理服务器及电子邮件(IMAP/POP3)代理服务器,并在一个BSD-like协议下发行,其特点是占有内存少,并发能力强。

[0178] 代理服务器的插件可以通过lua或者其他语言实现。其中,lua是一种可以嵌入到Nginx配置文件中的动态脚本语言,使用lua来实现Nginx插件,不会影响Nginx性能,能迅速实现请求分发。

[0179] 目标用户标识用于标识用户的身份信息,可以为uid,也可以为其它能够标识用户身份信息的标识。用户请求除包含目标用户标识之外,还可以包含其他信息,例如对应灰度发布的版本号等等,本发明并不对用户请求的具体内容进行限定。

[0180] 用户在客户端输入灰度发布的相关请求,客户端将该用户请求发送给前端的代理服务器。当该用户请求到达前端的代理服务器后,其内嵌的插件会去解析用户请求,以获得发送请求的用户的用户标识。

[0181] S220、将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件,以确定所述目标用户标识对应的目标服务。

[0182] 配置文件用于描述各个用户标识和各个服务之间的关联关系,可以根据自定义策略设置,只要能实现指定特定的用户进行发布即可。配置文件存储在代理服务器中,例如代理服务器的内存中。那么插件即可以直接从代理服务器中读取该配置文件,根据该配置文件,就可以直接确定出目标用户标识所对应的目标服务。相较于传统技术中每次都需要去数据库中读取数据进行查询的方案,该方式简化了操作,节省了资源。

[0183] S230、接收所述插件返回的所述目标服务。

[0184] 插件确定目标用户标识对应的目标服务后,将确定的该目标服务的信息返回给代理服务器。

[0185] S240、将所述用户请求转发给所述目标服务。

[0186] 代理服务器的Location会根据插件的返回结果将其用户请求转发给对应的服务器进行处理,实现灰度发布。

[0187] 本实施例通过配置文件指定服务所对应的用户标识,在接收到用户请求时,就可以根据该用户请求和配置文件确定目标用户标识对应的目标服务,从而将用户请求转发至该目标服务。相较于传统技术中由于根据权重或ip_hash选择的发布对象是随机的,不能指定特定的用户来进行发布而导致的可用性差的缺陷,本实施例选择的发布对象是指定的,因而可用性强;发布对象不受服务器故障影响,因而可靠性强。

[0188] 在一个实施例中,所述本地存储的配置文件为,与zookeeper第一次建立连接时,从所述zookeeper中获取的配置文件。

[0189] zookeeper本质上是一个用于协调分布式集群的管理服务,可以作为配置文件订阅的服务。配置文件可以放在zookeeper中,这样灵活性更好。由于在zookeeper中生成配置文件,因此如果代理服务器和zookeeper是第一次建立连接,则代理服务器中没有该配置文件,插件无法直接从代理服务器中读取该配置文件,因此插件需要从zookeeper中读取配置文件,根据读取的该配置文件确定目标用户标识对应的目标服务。为了确保后续可以直接从代理服务器中读取配置文件,在代理服务器中对该配置文件进行备份,因此代理服务器此时存储的文件即是第一次连接时从zookeeper中获取的配置文件。

[0190] 在一个实施例中,所述本地存储的配置文件为,在接收到zookeeper发送的配置文件变更通知时,从所述zookeeper中获取的变更后的配置文件。

[0191] zookeeper生成配置文件,当对zookeeper中的配置文件进行变更后,zookeeper就会主动向代理服务器发送配置文件变更通知。如果接收到zookeeper向代理服务器发送的配置文件变更通知,说明zookeeper中的配置文件已经进行变更,代理服务器当前存储的配置文件为老版本的配置文件,因此插件需要从zookeeper中直接读取变更后的配置文件,根据读取的该配置文件确定目标用户标识对应的目标服务。另外,还需要对代理服务器中存储的当前配置文件进行更新,以保证代理服务器存储的配置文件最新,后续可以直接从代理服务器中读取配置文件,因此代理服务器此时存储的文件即是从所述zookeeper中获取的变更后的配置文件。

[0192] 上述从代理服务器角度描述的实现服务灰度发布的方法的其它技术特征与上述从插件角度描述的实现服务灰度发布的方法的技术特征相同,在此不予赘述。

[0193] 基于同一发明构思,本发明还提供一种实现服务灰度发布的装置,下面结合附图对本发明装置的具体实施方式进行详细介绍。

[0194] 如图4所示,在一个实施例中,一种实现服务灰度发布的装置,包括:

[0195] 用户请求发送模块210,用于将接收的包含目标用户标识的用户请求发送给插件进行解析,以获得目标用户标识;

[0196] 配置文件发送模块220,用于将本地存储的用于指示用户标识和服务关联关系的配置文件发送给所述插件,以确定所述目标用户标识对应的目标服务;

[0197] 确定结果接收模块230,用于接收所述插件返回的所述目标服务;

[0198] 用户请求转发模块240,用于将所述用户请求转发给所述目标服务。

[0199] 在一个实施例中,所述本地存储的配置文件为,与zookeeper第一次建立连接时,从所述zookeeper中获取的配置文件。

[0200] 在一个实施例中,所述本地存储的配置文件为,在接收到zookeeper发送的配置文件变更通知时,从所述zookeeper中获取的变更后的配置文件。

[0201] 上述实现服务灰度发布的装置的其它技术特征与上述实现服务灰度发布的方法的技术特征相同,在此不予赘述。

[0202] 为了更好的理解本发明,下面结合三个具体实施例进行详细介绍。

[0203] 如图5所示,为一实施例的实现服务灰度发布的方法的流程示意图,该方法应用在代理服务器和zookeeper第一次建立连接时,包括步骤:

[0204] S1、Nginx接收包含用户标识的用户请求;

[0205] S2、Nginx内嵌的lua插件对用户请求进行解析,获得用户标识。从zookeeper中读

取包含白名单的配置文件,并存储在Nginx的内存中。根据该配置文件决定Nginx的跳转。比如,该用户标识在白名单内,返回@service_new,否则返回@service;

[0206] S3、Nginx的Location会根据lua插件的返回转发给对应的服务器,例如,@service_new则将用户请求转发给新上线的服务,@service则将用户请求转发给旧的服务。

[0207] 如图6所示,为一实施例的实现服务灰度发布的方法的流程示意图,该方法应用在代理服务器和zookeeper第N次建立连接且zookeeper中的配置文件无更新时,其中,N大于1,该方法包括步骤:

[0208] S1、Nginx接收包含用户标识的用户请求;

[0209] S2、Nginx内嵌的lua插件对用户请求进行解析,获得用户标识。从Nginx的内存中读取包含白名单的配置文件。根据该配置文件决定Nginx的跳转。比如,该用户标识在白名单内,返回@service_new,否则返回@service;

[0210] S3、Nginx的Location会根据lua插件的返回转发给对应的服务器,例如,@service_new则将用户请求转发给新上线的服务,@service则将用户请求转发给旧的服务。

[0211] 如图7所示,为一实施例的实现服务灰度发布的方法的流程示意图,该方法应用在zookeeper中的配置文件有更新时,该方法包括步骤:

[0212] S1、Nginx接收包含用户标识的用户请求;

[0213] S2、Nginx内嵌的lua插件对用户请求进行解析,获得用户标识。Nginx接收到zookeeper发送的配置文件更新通知,lua插件从zookeeper中读取更新后的包含白名单的配置文件,并对Nginx的内存中存储的当前配置文件进行更新。根据更新后的配置文件决定Nginx的跳转。比如,该用户标识在白名单内,返回@service_new,否则返回@service;

[0214] S3、Nginx的Location会根据lua插件的返回转发给对应的服务器,例如,@service_new则将用户请求转发给新上线的服务,@service则将用户请求转发给旧的服务。

[0215] 本发明实施例还提供一种计算机可读存储介质,其上存储有计算机程序,该程序被处理器执行时实现上述任意一项所述的实现服务灰度发布的方法。其中,所述存储介质包括但不限于任何类型的盘(包括软盘、硬盘、光盘、CD-ROM、和磁光盘)、ROM(Read-Only Memory,只读存储器)、RAM(Random Access Memory,随机存储器)、EPROM(Erasable Programmable Read-Only Memory,可擦写可编程只读存储器)、EEPROM(Electrically Erasable Programmable Read-Only Memory,电可擦可编程只读存储器)、闪存、磁性卡片或光线卡片。也就是,存储介质包括由设备(例如,计算机)以能够读的形式存储或传输信息的任何介质。可以是只读存储器,磁盘或光盘等。

[0216] 本发明实施例还提供一种插件设备,所述插件设备包括:

[0217] 一个或多个处理器;

[0218] 存储装置,用于存储一个或多个程序,

[0219] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现上述任意一项所述的实现服务灰度发布的方法。

[0220] 本发明实施例还提供一种代理服务器,所述代理服务器包括:

[0221] 一个或多个处理器;

[0222] 存储装置,用于存储一个或多个程序,

[0223] 当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理

器实现上述任意一项所述的实现服务灰度发布的方法。

[0224] 图8为本发明代理服务器的结构示意图,包括处理器320、存储装置330、输入单元340以及显示单元350等器件。本领域技术人员可以理解,图8示出的结构器件并不构成对所有代理服务器的限定,可以包括比图示更多或更少的部件,或者组合某些部件。存储装置330可用于存储应用程序310以及各功能模块,处理器320运行存储在存储装置330的应用程序310,从而执行设备的各种功能应用以及数据处理。存储装置330可以是内存储器或外存储器,或者包括内存储器和外存储器两者。内存储器可以包括只读存储器、可编程ROM (PROM)、电可编程ROM (EPROM)、电可擦写可编程ROM (EEPROM)、快闪存储器、或者随机存储器。外存储器可以包括硬盘、软盘、ZIP盘、U盘、磁带等。本发明所公开的存储装置包括但不限于这些类型的存储装置。本发明所公开的存储装置330只作为例子而非作为限定。

[0225] 输入单元340用于接收信号的输入,以及接收用户请求以及插件返回的结果。输入单元340可包括触控面板以及其它输入设备。触控面板可收集用户在其上或附近的触摸操作(比如用户使用手指、触笔等任何适合的物体或附件在触控面板上或在触控面板附近的操作),并根据预先设定的程序驱动相应的连接装置;其它输入设备可以包括但不限于物理键盘、功能键(比如播放控制按键、开关按键等)、轨迹球、鼠标、操作杆中的一种或多种。显示单元350可用于显示用户输入的信息或提供给用户的信息以及计算机设备的各种菜单。显示单元350可采用液晶显示器、有机发光二极管等形式。处理器320是计算机设备的控制中心,利用各种接口和线路连接整个电脑的各个部分,通过运行或执行存储在存储装置330内的软件程序和/或模块,以及调用存储在存储装置内的数据,执行各种功能和处理数据。

[0226] 在一实施方式中,服务器包括一个或多个处理器320,以及一个或多个存储装置330,一个或多个应用程序310,其中所述一个或多个应用程序310被存储在存储装置330中并被配置为由所述一个或多个处理器320执行,所述一个或多个应用程序310配置用于执行以上实施例所述的实现服务灰度发布的方法。

[0227] 应该理解的是,虽然附图的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,其可以以其他的顺序执行。而且,附图的流程图中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,其执行顺序也不必然是依次进行,而是可以与其他步骤或者其他步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

[0228] 应该理解的是,在本发明各实施例中的各功能单元可集成在一个处理模块中,也可以各个单元单独物理存在,也可以两个或两个以上单元集成于一个模块中。上述集成的模块既可以采用硬件的形式实现,也可以采用软件功能模块的形式实现。

[0229] 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。

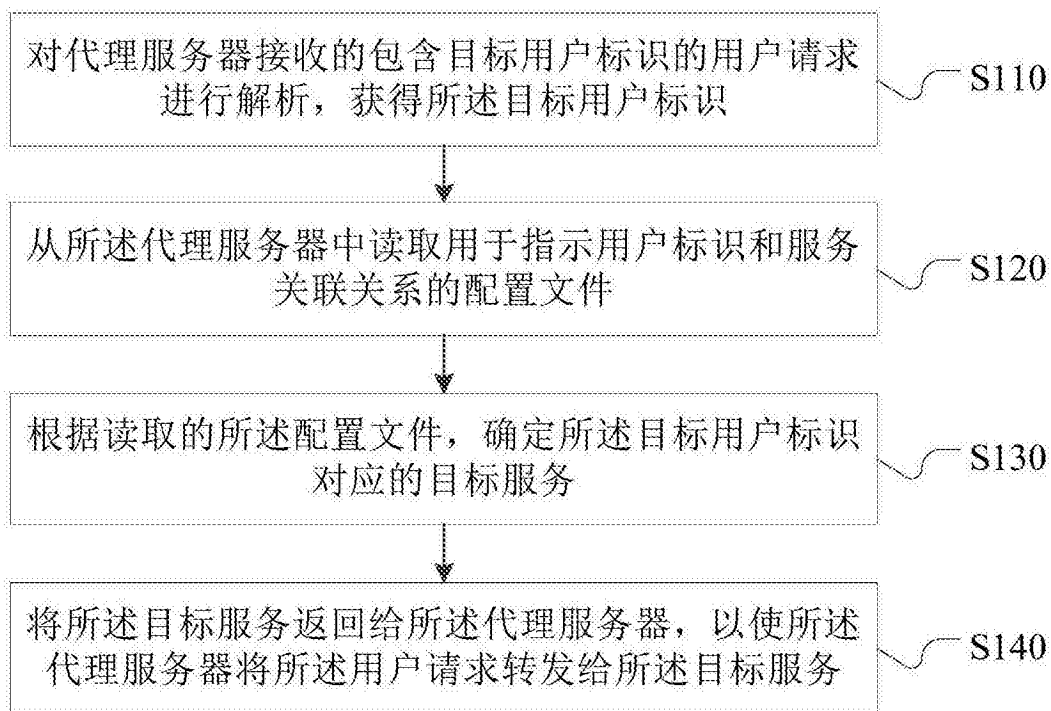


图1

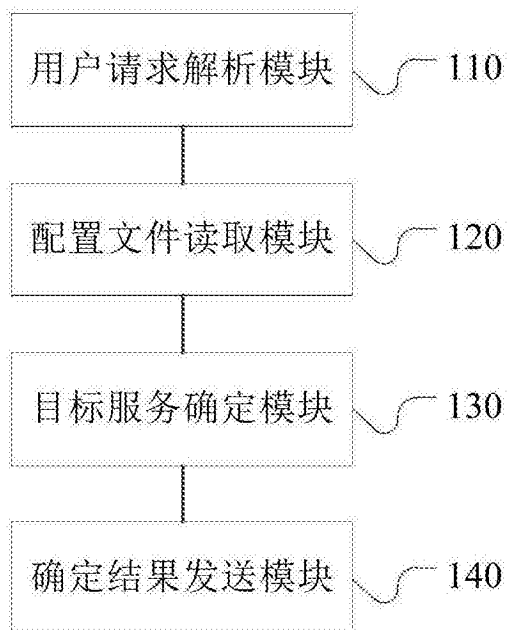


图2

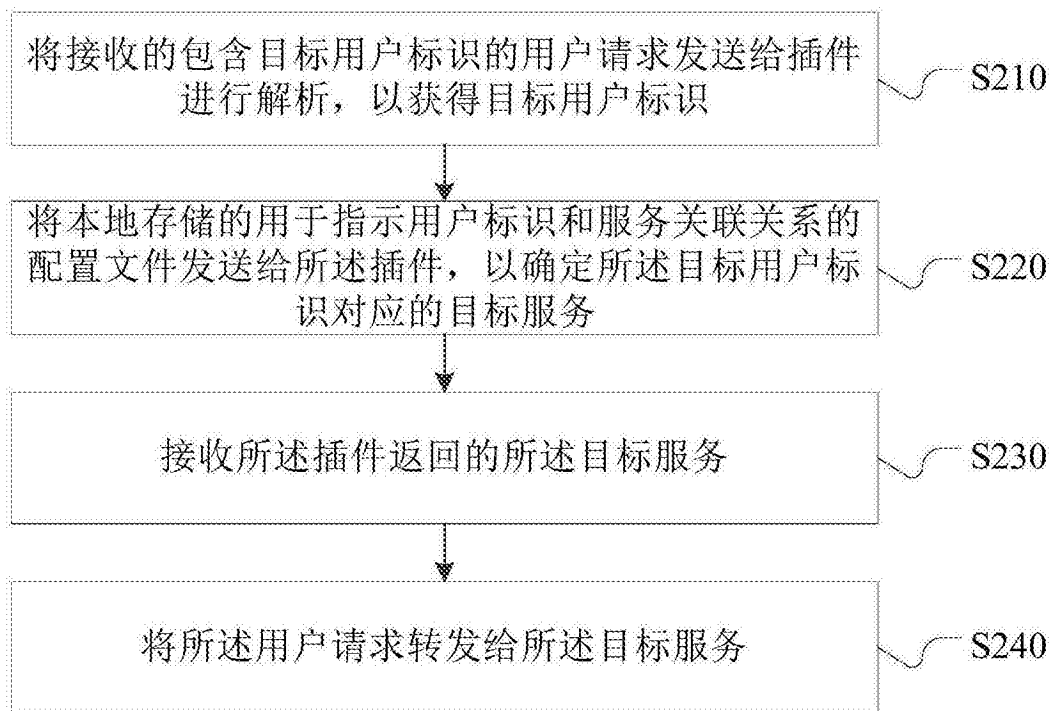


图3

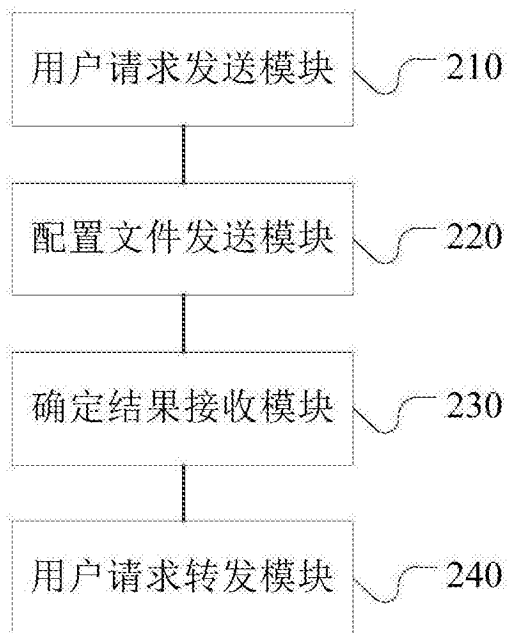


图4

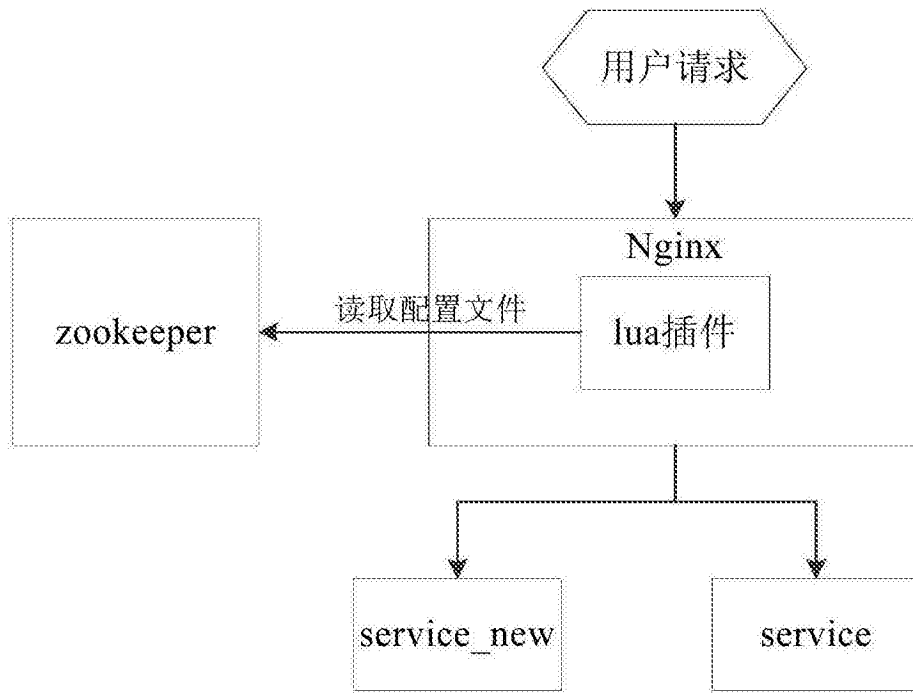


图5

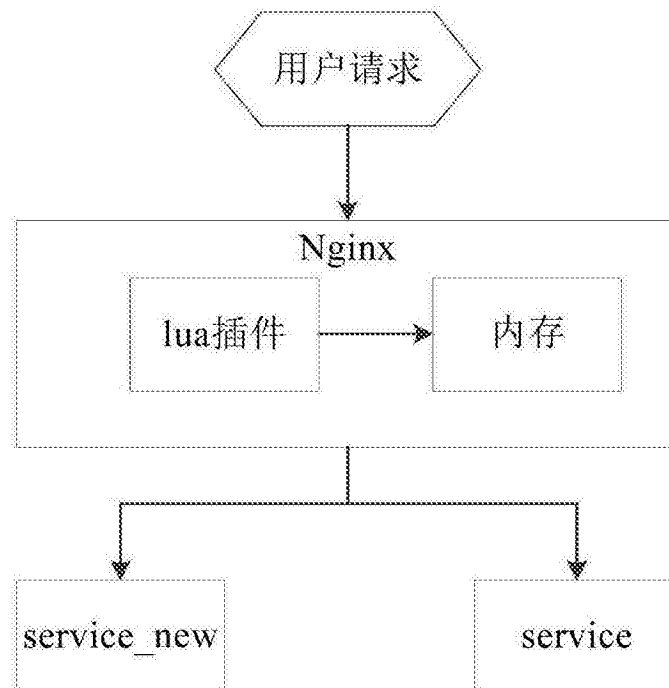


图6

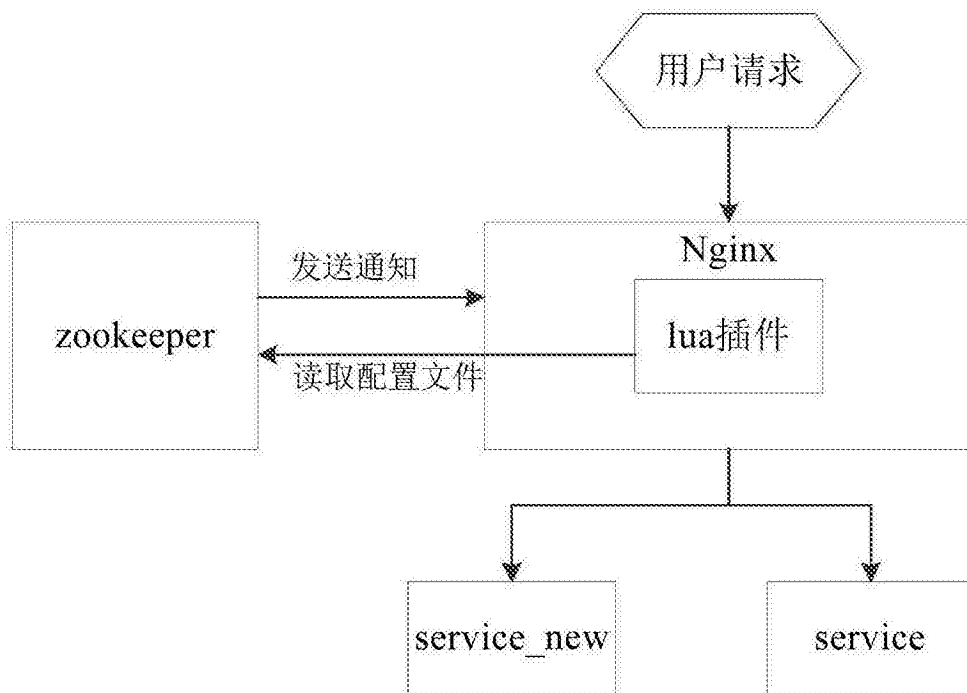


图7

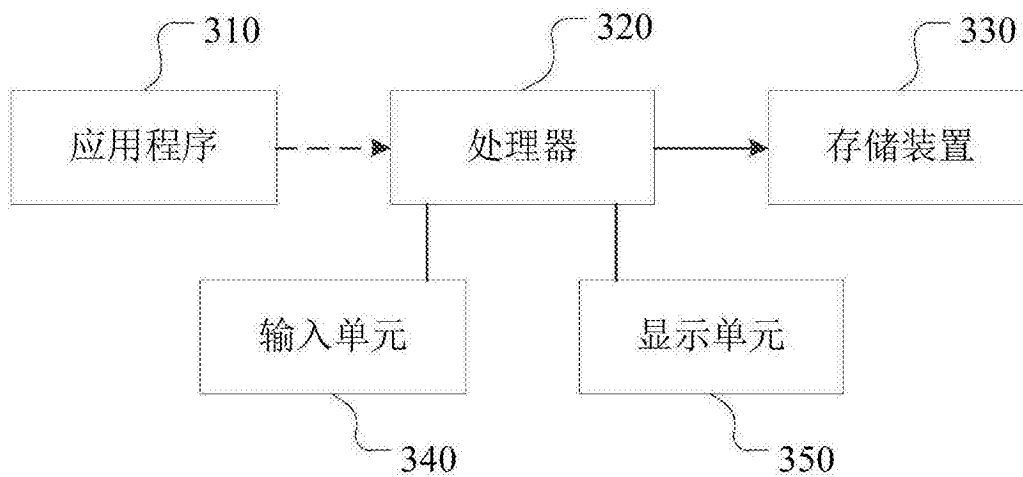


图8