



(12) 发明专利

(10) 授权公告号 CN 113687929 B

(45) 授权公告日 2024. 08. 20

(21) 申请号 202110975231.4

G06F 9/52 (2006.01)

(22) 申请日 2021.08.24

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 101299677 A, 2008.11.05

申请公布号 CN 113687929 A

审查员 胡振洲

(43) 申请公布日 2021.11.23

(73) 专利权人 济南浪潮数据技术有限公司

地址 250101 山东省济南市自由贸易试验区  
济南片区浪潮路1036号浪潮科技园  
S05楼S311室

(72) 发明人 董文华 张亚奇 李世杰

(74) 专利代理机构 北京集佳知识产权代理有限公司 11227

专利代理师 丁曼曼

(51) Int. Cl.

G06F 9/48 (2006.01)

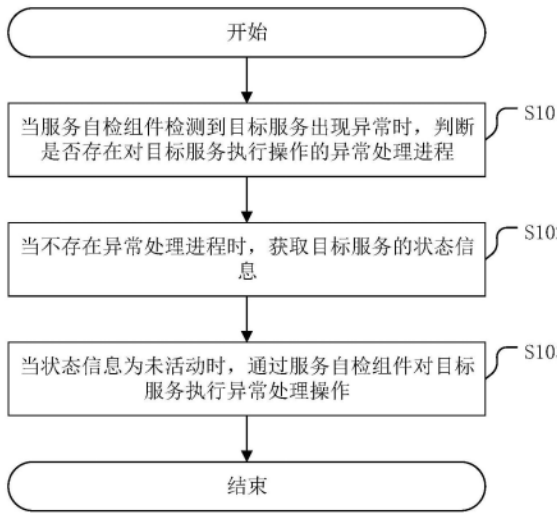
权利要求书2页 说明书7页 附图1页

(54) 发明名称

一种异常处理方法及相关装置

(57) 摘要

本申请公开了一种异常处理方法,包括:当服务自检组件检测到目标服务出现异常时,判断是否存在对目标服务执行操作的异常处理进程;当不存在异常处理进程时,获取目标服务的状态信息;当状态信息为未活动时,通过服务自检组件对目标服务执行异常处理操作。通过当服务自检组件检测到异常时,判断是否存在对目标服务执行操作的异常处理进程,并且在不存在异常处理进程时再确定状态信息是否为未活动,避免同时存在进行操作的异常处理进程,也避免其他的进程刚对目标服务处理完成,不用对目标服务再次进行异常处理,避免出现互斥问题,提高了操作文件系统的体验。本申请还公开了一种异常处理装置、服务器以及计算机可读存储介质,具有以上有益效果。



1. 一种异常处理方法,其特征在于,应用于ctdb对于smb异常进行处理的场景,包括:

当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程;所述服务自检组件为ctdb,所述目标服务为smb服务;

当不存在所述异常处理进程时,获取所述目标服务的状态信息;其中,所述状态信息包括活动和未活动,所述状态信息为活动时用于表征所述目标服务刚刚被异常处理完成,所述状态信息为未活动时用于表征所述目标服务还没有被执行相应的异常处理操作;

当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作;

其中,当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程,包括:

当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在管理软件下发的所述异常处理进程;

其中,当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在管理软件下发的所述异常处理进程,包括:

当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在所述管理软件下发的ism\_pid进程;

其中,还包括:

当存在所述异常处理进程时,对所述服务自检组件的异常处理操作执行终止操作,并通过所述异常处理进程对所述目标服务进行异常处理;

其中,当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作,包括:

当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行重启操作;

其中,还包括:

当所述状态信息为活动时,对所述服务自检组件的异常处理操作执行终止操作。

2. 根据权利要求1所述的异常处理方法,其特征在于,当不存在所述异常处理进程时,获取所述目标服务的状态信息,包括:

当不存在所述异常处理进程时,通过查询命令获取所述目标服务的状态信息。

3. 一种异常处理装置,其特征在于,应用于ctdb对于smb异常进行处理的场景,包括:

进程判断模块,用于当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程;所述服务自检组件为ctdb,所述目标服务为smb服务;

状态获取模块,用于当不存在所述异常处理进程时,获取所述目标服务的状态信息;其中,所述状态信息包括活动和未活动,所述状态信息为活动时用于表征所述目标服务刚刚被异常处理完成,所述状态信息为未活动时用于表征所述目标服务还没有被执行相应的异常处理操作;

异常处理模块,用于当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作;

其中,所述进程判断模块,具体用于当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的异常处理进程;

其中,所述进程判断模块,具体用于当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的ism\_pid进程;

其中,第一终止模块,用于当存在异常处理进程时,对服务自检组件的异常处理操作执行终止操作,并通过异常处理进程对目标服务进行异常处理;

其中,所述异常处理模块,具体用于当状态信息为未活动时,通过服务自检组件对目标服务执行重启操作;

其中,第二终止模块,用于当状态信息为活动时,对服务自检组件的异常处理操作执行终止操作。

4.一种服务器,其特征在于,包括:

存储器,用于存储计算机程序;

处理器,用于执行所述计算机程序时实现如权利要求1至2任一项所述的异常处理方法的步骤。

5.一种计算机可读存储介质,其特征在于,所述计算机可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现如权利要求1至2任一项所述的异常处理方法的步骤。

## 一种异常处理方法及相关装置

### 技术领域

[0001] 本申请涉及计算机技术领域,特别涉及一种异常处理方法、异常处理装置、服务器以及计算机可读存储介质。

### 背景技术

[0002] 随着数据存储技术的不断发展,在文件系统中出现了越来越多的功能,以应对复杂的使用场景。其中,smb(Server Messages Block,信息服务块)为ceph(分布式文件系统)文件系统的组件,提供smb服务将文件系统上的目录、文件共享出去。以windows客户端为例,可以通过映射网络驱动器的方式挂载smb的共享文件夹,映射为windows的网络盘。

[0003] 相关技术中,在管理软件上在配置smb服务时,会下发restart smb命令,此时restart分为stop和start两个流程。当执行到stop时,刚好被ctdb(Cluster Trivial Database,集群数据库)监控到smb处于stop状态,那么ctdb也会执行restart smb将smb服务拉起来。ctdb执行stop时,此时管理软件刚执行到start smb,从而管软的start就会被取消,导致管软上报错重启smb失败。同理,当ctdb正在restart smb服务,管理软件若恰巧也下发了restart smb命令,那么ctdb也会报错重启smb失败。可见,在对应的管理软件中容易出现多个资源访问资源出现的互斥问题,降低了操作文件系统的体验。

[0004] 因此,如何避免出现访问资源的互斥问题为本领域技术人员关注的重点问题。

### 发明内容

[0005] 本申请的目的是提供一种异常处理方法、异常处理装置、服务器以及计算机可读存储介质,以避免出现异常处理时的互斥问题,提高操作文件系统的体验。

[0006] 为解决上述技术问题,本申请提供一种异常处理方法,包括:

[0007] 当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程;

[0008] 当不存在所述异常处理进程时,获取所述目标服务的状态信息;

[0009] 当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作。

[0010] 可选的,当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程,包括:

[0011] 当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在管理软件下发的所述异常处理进程。

[0012] 可选的,当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在管理软件下发的所述异常处理进程,包括:

[0013] 当所述服务自检组件检测到所述目标服务的端口未被占用时,判断是否存在所述管理软件下发的ism\_pid进程。

[0014] 可选的,还包括:

- [0015] 当存在所述异常处理进程时,对所述服务自检组件的异常处理操作执行终止操作,并通过所述异常处理进程对所述目标服务进行异常处理。
- [0016] 可选的,当不存在所述异常处理进程时,获取所述目标服务的状态信息,包括:
- [0017] 当不存在所述异常处理进程时,通过查询命令获取所述目标服务的状态信息。
- [0018] 可选的,当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作,包括:
- [0019] 当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行重启操作。
- [0020] 可选的,还包括:
- [0021] 当所述状态信息为活动时,对所述服务自检组件的异常处理操作执行终止操作。
- [0022] 本申请还提供一种异常处理装置,包括:
- [0023] 进程判断模块,用于当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程;
- [0024] 状态获取模块,用于当不存在所述异常处理进程时,获取所述目标服务的状态信息;
- [0025] 异常处理模块,用于当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作。
- [0026] 本申请还提供一种服务器,包括:
- [0027] 存储器,用于存储计算机程序;
- [0028] 处理器,用于执行所述计算机程序时实现如上所述的异常处理方法的步骤。
- [0029] 本申请还提供一种计算机可读存储介质,所述计算机可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现如上所述的异常处理方法的步骤。
- [0030] 本申请所提供的一种异常处理方法,包括:当服务自检组件检测到目标服务出现异常时,判断是否存在对所述目标服务执行操作的异常处理进程;当不存在所述异常处理进程时,获取所述目标服务的状态信息;当所述状态信息为未活动时,通过所述服务自检组件对所述目标服务执行异常处理操作。
- [0031] 通过当服务自检组件检测到异常时,判断是否存在对所述目标服务执行操作的异常处理进程,即确定此时是否存在其他的进程进行异常处理,并且在不存在异常处理进程时再确定状态信息是否为未活动,最后再进行异常处理操作,避免了同时存在进行操作的异常处理进程,也避免了其他的进程刚对目标服务处理完成,不用对目标服务再次进行异常处理,最终避免出现互斥问题,提高了操作文件系统的体验。
- [0032] 本申请还提供一种异常处理装置、服务器以及计算机可读存储介质,具有以上有益效果,在此不做赘述。

## 附图说明

[0033] 为了更清楚地说明本申请实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的实施例,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,还可以根据提供的附图获得其他的附图。

[0034] 图1为本申请实施例所提供的一种异常处理方法的流程图；

[0035] 图2为本申请实施例所提供的一种异常处理装置的结构示意图。

### 具体实施方式

[0036] 本申请的核心是提供一种异常处理方法、异常处理装置、服务器以及计算机可读存储介质,以避免出现异常处理时的互斥问题,提高操作文件系统的体验。

[0037] 为使本申请实施例的目的、技术方案和优点更加清楚,下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本申请一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本申请保护的范围。

[0038] 相关技术中,在管理软件上在配置smb服务时,会下发restart smb命令,此时restart分为stop和start两个流程。当执行到stop时,刚好被ctdb(Cluster Trivial Database,集群数据库)监控到smb处于stop状态,那么ctdb也会执行restart smb将smb服务拉起来。ctdb执行stop时,此时管理软件刚执行到start smb,从而管软的start就会被取消,导致管软上报错重启smb失败。同理,当ctdb正在restart smb服务,管理软件若恰巧也下发了restart smb命令,那么ctdb也会报错重启smb失败。可见,在对应的管理软件中容易出现多个资源访问资源出现的互斥问题,降低了操作文件系统的体验。

[0039] 因此,本申请提供的一种异常处理方法,通过当服务自检组件检测到异常时,判断是否存在对所述目标服务执行操作的异常处理进程,即确定此时是否存在其他的进程进行异常处理,并且在不存在异常处理进程时再确定状态信息是否为未活动,最后再进行异常处理操作,避免了同时存在进行操作的异常处理进程,也避免了其他的进程刚对目标服务处理完成,不用对目标服务再次进行异常处理,最终避免出现互斥问题,提高了操作文件系统的体验。

[0040] 以下通过一个实施例,对本申请提供的一种异常处理方法进行说明。

[0041] 请参考图1,图1为本申请实施例所提供的一种异常处理方法的流程图。

[0042] 本实施例中,该方法可以包括:

[0043] S101,当服务自检组件检测到目标服务出现异常时,判断是否存在对目标服务执行操作的异常处理进程;

[0044] 可见,本步骤旨在当服务自检组件检测到目标服务出现异常时,判断是否存在对目标服务执行操作的异常处理进程。也就是,该服务自检组件判断是否该目标服务是否出现异常。当判定该目标服务出现异常时,为了避免对出现同时对目标服务进行操作导致的操作失败的问题。

[0045] 进一步的,本步骤可以包括:

[0046] 当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的异常处理进程。

[0047] 可见,本可选方案主要是对如何检测和判断做进一步说明。本可选方案中当服务自检组件检测到目标端口未被占用时,判断是否存在管理软件下发的异常处理进程。因此,可以想到的是本可选方案中需要对该目标目标服务的端口是否被占用。若是,则说明目标服务服务正常,没有异常。若否,则说明该目标服务存在异常。

[0048] 举例来说,在smb服务中,当smb服务启动时,445和139端口处于被占用的状态。因此,ctdb每5s执行一次monitor事件,monitor事件中调用ctdb checktcpport 445的方式来检测smb的445端口是否被占用,具体原理是尝试bind445端口,若可以绑定成功说明改端口未被占用,如果绑定失败,说明改端口被smb服务占用。若445端口被占用说明smb服务正常,反之,445端口未被占用说明smb服务异常。

[0049] 进一步的,上述可选方案也可以包括:

[0050] 当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的ism\_pid进程。

[0051] 可见,本可选方案主要是对如何进行判断作说明。本可选方案中,当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的ism\_pid进程。其中,ism\_pid进程即为管理软件对smb进行重启操作的进程。

[0052] S102,当不存在异常处理进程时,获取目标服务的状态信息;

[0053] 在S101的基础上,本步骤旨在当不存在异常处理进程时,获取目标服务的状态信息。

[0054] 也就是说,当不存在异常处理进程时,也无法确定此时对目标服务进行异常处理是没有问题的。还需要通过该目标服务的状态信息进行判断。即判断该目标服务是否刚刚被异常处理过。因此,本步骤中需要获取到该目标服务的状态信息。

[0055] 其中,状态信息为活动或未活动。当状态信息为活动时,则说明该目标服务刚刚被异常处理完成,不需要再次进行异常处理。当该状态信息为未活动时,则说明该目标服务还没有被执行相应的异常处理操作,需要执行异常处理操作。

[0056] 进一步的,本步骤可以包括:

[0057] 当不存在异常处理进程时,通过查询命令获取目标服务的状态信息。

[0058] 可见,本可选方案主要是对如何获取状态信息做说明。本可选方案中,当不存在异常处理进程时,通过查询命令获取目标服务的状态信息。其中,采用的查询命令可以采用现有技术提供的任意一种查询命令,在此不做具体限定。

[0059] 此外,本实施例还可以包括:

[0060] 当存在异常处理进程时,对服务自检组件的异常处理操作执行终止操作,并通过异常处理进程对目标服务进行异常处理。

[0061] 可见,本可选方案中主要是对当存在异常处理进程时如何处理作说明。本可选方案中,当存在该异常处理进程则说明此时存在对该目标服务进行处理的进程,该服务自检组件不需要再进行异常处理,需要将该服务自检组件的相关操作进行停止处理。

[0062] S103,当状态信息为未活动时,通过服务自检组件对目标服务执行异常处理操作。

[0063] 在S102的基础上,本步骤旨在当状态信息为未活动时,通过服务自检组件对目标服务执行异常处理操作。

[0064] 进一步的,本步骤可以包括:

[0065] 当状态信息为未活动时,通过服务自检组件对目标服务执行重启操作。

[0066] 可见,本可选方案主要是对如何进行异常处理作说明。本可选方案中,当状态信息为未活动时,通过服务自检组件对目标服务执行重启操作。

[0067] 此外,本实施例还可以包括:

[0068] 当状态信息为活动时,对服务自检组件的异常处理操作执行终止操作。

[0069] 可见,本可选方案主要是当状态信息为活动时如何处理作说明。当该状态信息为活动时,则说明异常处理进程刚对该目标服务进行异常处理结束,需要再进行异常处理,则将相关的操作进行终止操作即可。

[0070] 综上,本实施例通过当服务自检组件检测到异常时,判断是否存在对目标服务执行操作的异常处理进程,即确定此时是否存在其他的进程进行异常处理,并且在不存在异常处理进程时再确定状态信息是否为未活动,最后再进行异常处理操作,避免了同时存在进行操作的异常处理进程,也避免了其他的进程刚对目标服务处理完成,不用对目标服务再次进行异常处理,最终避免出现互斥问题,提高了操作文件系统的体验。

[0071] 以下通过一个具体的实施例,对本申请提供的一种异常处理方法做进一步说明。

[0072] 本实施例中,该方法可以包括:

[0073] 步骤1,当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的ism\_pid进程;

[0074] 步骤2,当存在异常处理进程时,对服务自检组件的异常处理操作执行终止操作,并通过异常处理进程对目标服务进行异常处理;

[0075] 步骤3,当不存在异常处理进程时,通过查询命令获取目标服务的状态信息;

[0076] 步骤4,当状态信息为未活动时,通过服务自检组件对目标服务执行重启操作;

[0077] 步骤5,当状态信息为活动时,对服务自检组件的异常处理操作执行终止操作。

[0078] 基于上述步骤,在具体应用场景中,可以将上述的操作步骤应用于ctdb对于smb异常进行处理的技术领域中。

[0079] 该实施例的步骤可以包括:

[0080] 步骤1,ctdb检测到smb服务异常时,查看管软start或restart smb的进程ism\_pid是否存在;

[0081] 步骤2,如果ism\_pid进程存在,则ctdb重启smb的流程自动退出,管软重启smb流程继续进行;

[0082] 步骤3,如果ism\_pid进程不存在,则说明要么ism\_pid进程刚退出,要么没有进程启动过smb.此时需要通过systemctl status smb命令查看smb状态来进一步判断;

[0083] 步骤4,若smb状态为active,则说明ism\_pid进程刚退出,smb已经被启动了,ctdb无需再重启smb,那么ctdb负责重启smb的子进程退出;

[0084] 步骤5,若smb状态为inactive,则说明ism\_pid进程未出现过,smb状态仍为异常,则需要ctdb重启smb服务。

[0085] 可见,本实施例通过当服务自检组件检测到异常时,判断是否存在对目标服务执行操作的异常处理进程,即确定此时是否存在其他的进程进行异常处理,并且在不存在异常处理进程时再确定状态信息是否为未活动,最后再进行异常处理操作,避免了同时存在进行操作的异常处理进程,也避免了其他的进程刚对目标服务处理完成,不用对目标服务再次进行异常处理,最终避免出现互斥问题,提高了操作文件系统的体验。

[0086] 下面对本申请实施例提供的异常处理装置进行介绍,下文描述的异常处理装置与上文描述的异常处理方法可相互对应参照。

[0087] 请参考图2,图2为本申请实施例所提供的一种异常处理装置的结构示意图。

[0088] 本实施例中,该装置可以包括:

[0089] 进程判断模块100,用于当服务自检组件检测到目标服务出现异常时,判断是否存在对目标服务执行操作的异常处理进程;

[0090] 状态获取模块200,用于当不存在异常处理进程时,获取目标服务的状态信息;

[0091] 异常处理模块300,用于当状态信息为未活动时,通过服务自检组件对目标服务执行异常处理操作。

[0092] 可选的,该进程判断模块100,具体用于当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的异常处理进程。

[0093] 可选的,该进程判断模块100,具体用于当服务自检组件检测到目标服务的端口未被占用时,判断是否存在管理软件下发的ism\_pid进程。

[0094] 可选的,该装置,还可以包括:

[0095] 第一终止模块,用于当存在异常处理进程时,对服务自检组件的异常处理操作执行终止操作,并通过异常处理进程对目标服务进行异常处理。

[0096] 可选的,该状态获取模块200,具体用于当不存在异常处理进程时,通过查询命令获取目标服务的状态信息。

[0097] 可选的,该异常处理模块300,具体用于当状态信息为未活动时,通过服务自检组件对目标服务执行重启操作。

[0098] 可选的,该装置,还可以包括:

[0099] 第二终止模块,用于当状态信息为活动时,对服务自检组件的异常处理操作执行终止操作。

[0100] 本申请实施例还提供一种服务器,包括:

[0101] 存储器,用于存储计算机程序;

[0102] 处理器,用于执行所述计算机程序时实现如以上实施例所述的异常处理方法的步骤。

[0103] 本申请实施例还提供一种计算机可读存储介质,所述计算机可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现如以上实施例所述的异常处理方法的步骤。

[0104] 说明书中各个实施例采用递进的方式描述,每个实施例重点说明的都是与其他实施例的不同之处,各个实施例之间相同相似部分互相参见即可。对于实施例公开的装置而言,由于其与实施例公开的方法相对应,所以描述的比较简单,相关之处参见方法部分说明即可。

[0105] 专业人员还可以进一步意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、计算机软件或者二者的结合来实现,为了清楚地说明硬件和软件的可互换性,在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本申请的范围。

[0106] 结合本文中所公开的实施例描述的方法或算法的步骤可以直接用硬件、处理器执行的软件模块,或者二者的结合来实施。软件模块可以置于随机存储器(RAM)、内存、只读存

储器 (ROM)、电可编程ROM、电可擦除可编程ROM、寄存器、硬盘、可移动磁盘、CD-ROM、或技术领域内所公知的任意其它形式的存储介质中。

[0107] 以上对本申请所提供的一种异常处理方法、异常处理装置、服务器以及计算机可读存储介质进行了详细介绍。本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想。应当指出,对于本技术领域的普通技术人员来说,在不脱离本申请原理的前提下,还可以对本申请进行若干改进和修饰,这些改进和修饰也落入本申请权利要求的保护范围内。

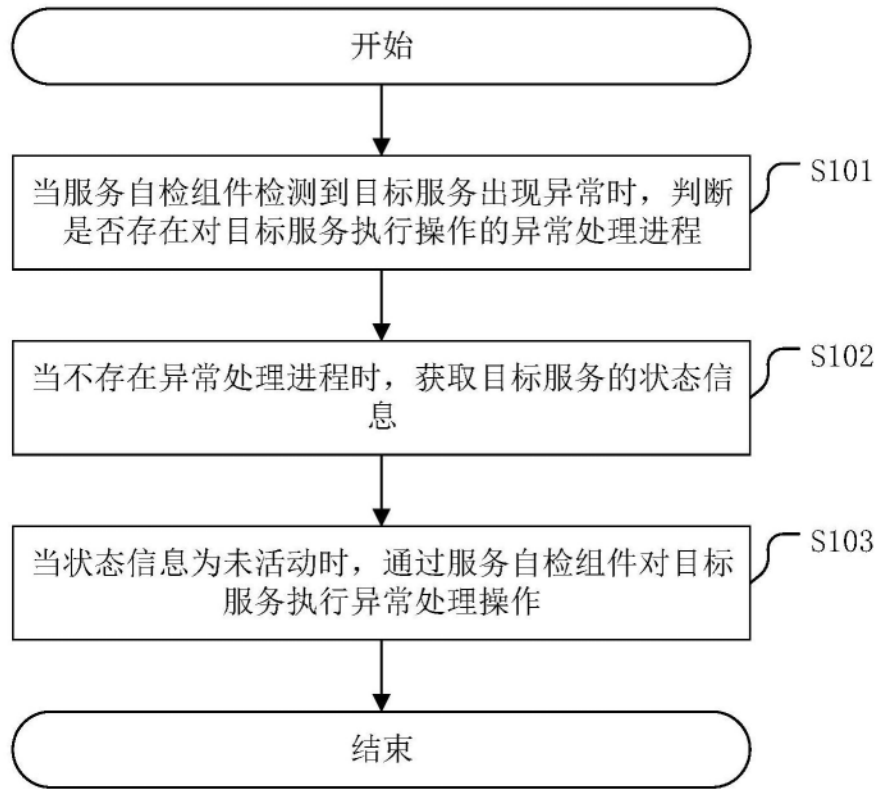


图1



图2