

## 目录

### 概述

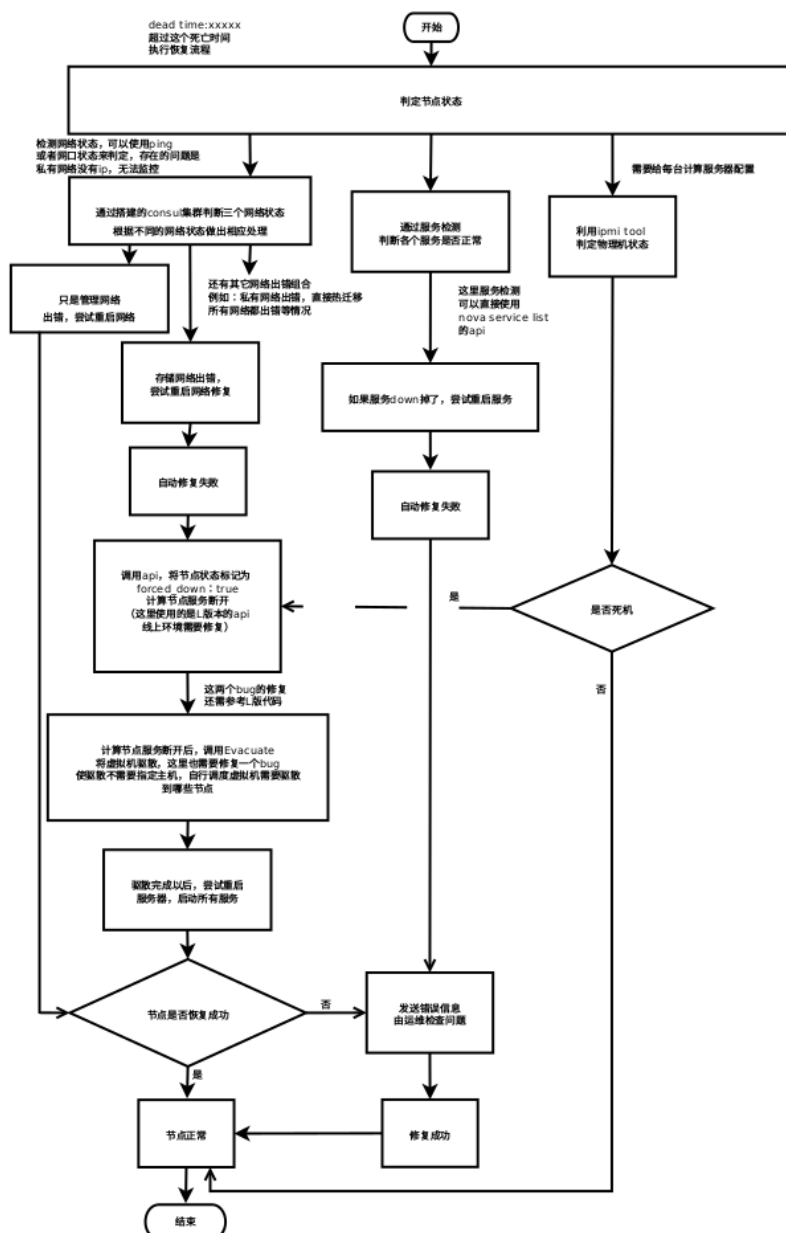
1 总体设计 .....	1
2 详细设计 .....	2
2.1 网络监控 .....	3
2.2 服务监控 .....	4
2.3 物理主机监控 .....	5
2.4 尝试恢复与隔离 .....	6
2.5 驱散虚拟机 .....	7
3 API 说明 .....	8

## 概述

所谓计算节点高可用，是指在计算节点发生硬件故障，如磁盘损坏、CPU 温度过高导致宕机、物理网络故障时，自动将该节点关闭，并让其上的虚拟机在剩下的健康节点上重启，如果可能，最好执行动态迁移

## 1 总体设计

使用consul对openstack集群的网络监控，通过对获取到的数据作相应处理来实现计算节点高可用，管理网络和私有网络出故障，通过给运维发送邮件的方式处理，存储网络出现故障，程序尝试修复，修复失败后隔离该计算节点并将节点上的虚拟机驱散到其它节点上去，总体设计流程图如下：



如上图所示，本程序包括三个主要模块，分别是consul集群的网络检测与处理，各计算节点服务检测与处理和ipmi物理主机的状态检测与处理。

consul集群的网络检测与处理：通过搭建的consul集群判断三个网络状态，根据不同的网络状态做出相应的处理，管理网络和私有网络出错，发送邮件给运维，存储网络出错，尝试重启网络，重启失败，调用api，将节点状态标记为forced down: true，快速的让服务端发现计算节点出错，计算节点断开以后，调用Evacuate将虚拟机驱散，驱散完成以后将该节点标记为fence状态并发送邮件给运维。

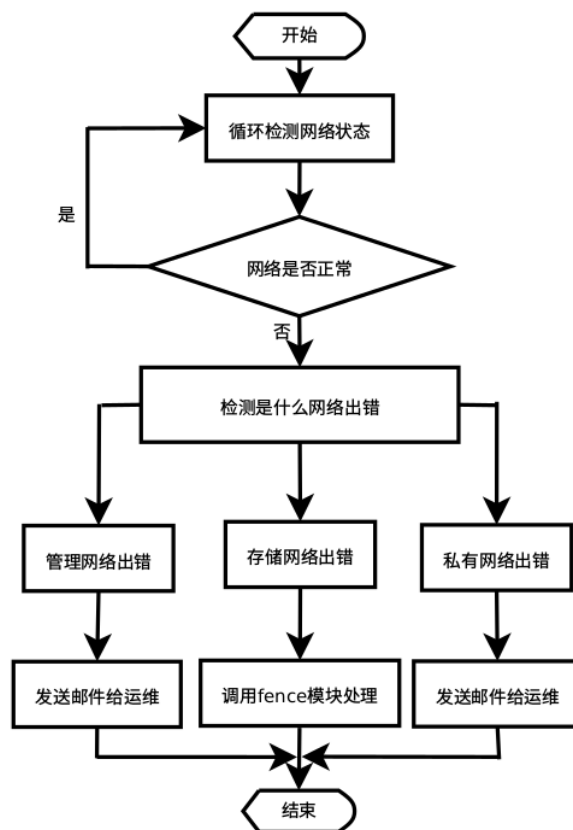
计算节点服务检测与处理：通过自定义程序获取各个计算节点的服务状态，这里同时获取两个状态，分别是服务端获取的状态和节点上实际运行的服务状态，如果只是服务端状态出错，等待一段时间看能否自动修复，两个状态都失败，尝试重启节点上的服务，服务修复失败，进入隔离和驱散流程。

物理主机状态检测与处理：利用ipmi判断物理主机状态，如果检测主机关机，尝试启动节点主机，启动不成功则发送邮件给运维，启动成功，自动进入网络和服务检测，执行相应的处理。

## 2 详细设计

### 2.1 网络监控

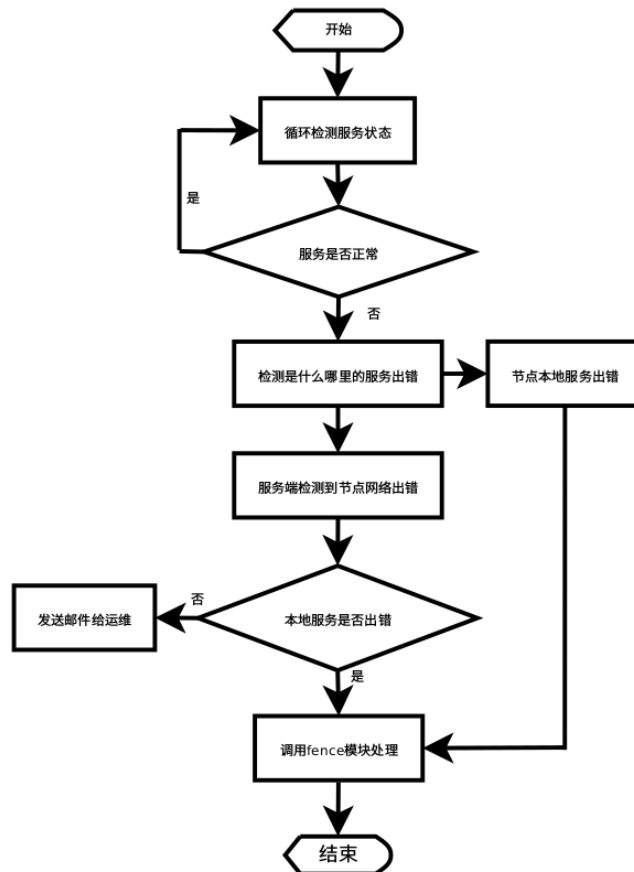
部署三个独立的consul集群，各个集群分别监控管理网络，存储网络和私有网络，程序调用consul api来获取到当前的网络状态，并做出相应的处理，网络监控的流程图如下：



如上图所示，程序循环检测网络状态，网络正常情况下不做任何处理，一旦发现网络故障，判定是哪个网络出错，管理网络和私有网络出错直接发送邮件给运维处理，存储网络出现故障，转交给 **fence** 模块进一步处理。

## 2.2 服务监控

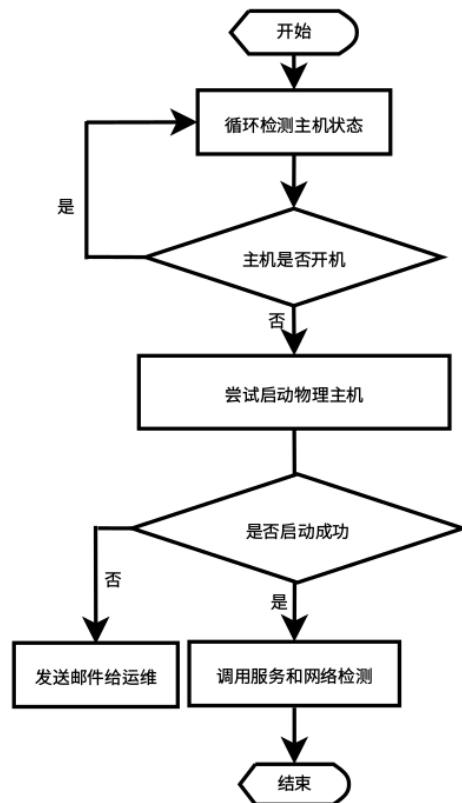
通过程序循环检测计算节点上的计算服务和服务端获取到的服务是否正常，根据两个地方的服务状态做相应的处理，流程图如下：



如上图所示，程序分别检测计算服务是否正常，一方面通过 `api` 获取当前集群中计算服务是否正常，另一方面检测各个计算节点主机上的计算服务是否正常，如果只是集群检测到服务出错而节点本身服务运行正常，等待集群自动恢复并发送一个警告邮件给运维，如果节点主机上的服务出错，调用 `fence` 模块处理。

## 2.3 物理主机监控

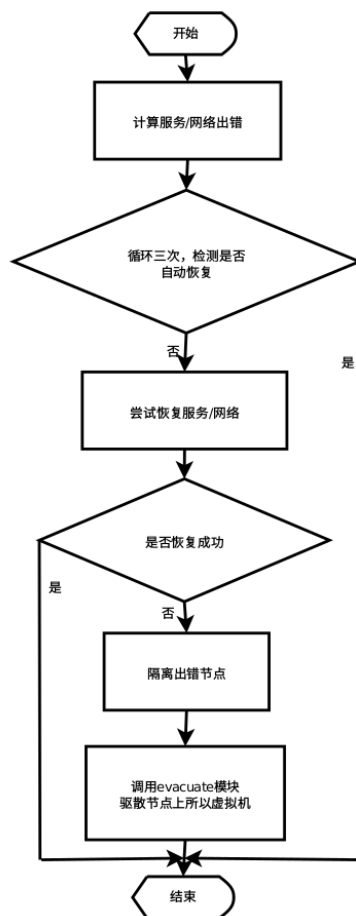
物理主机的监控主要是利用 `ipmitool` 来获取物理主机的状态，如果关机状态，尝试重启物理机，流程图如下：



如上图所示，程序利用工具循环检测物理主机状态，一旦发现主机关机，尝试重启机器，无法启动则发送邮件给运维，启动成功以后，调用服务检测模块和网络检测模块查看是否正常加入集群，如果存在错误则执行相应的处理。

## 2.4 尝试恢复与隔离

程序根据接收到的不同请求，执行不同的操作，其中包括网络服出错的请求和计算服务出错的请求，程序接收到出错消息，尝试恢复错误，如果恢复失败，就将出错的节点隔离出来，流程图如下：



如上图所示，**fence** 模块接收到计算服务和网络出问题，首先多次循环检测问题是否自动恢复，确定确实出现故障时，尝试恢复服务和网络，如果能够恢复成功，直接退出，恢复失败就调用 **api** 将出错的计算节点隔离出去，然后调用 **evacuate** 模块将节点上虚拟机驱散出去。

## 2.5 驱散虚拟机

程序调用 **api** 接口将隔离出来的节点上的虚拟机全部迁移到其它正常运行的计算节点上。

### 3 API 说明

/servers/{server\_id}/action

Evacuate Server (evacuate Action)

- 把一台服务从失败的主机 `evacuate` 到一台新主机上
- 规定在有需求的 `body` 上执行 `evacuate` 功能
  - 主机上，如果是共享存储的，不设置 `adminPass`
  - 目的主机不能和实例主机相同

Normal response codes: 200

Error response codes: `computeFault(400, 500)`, `serviceUnavailable(503)`, `badRequest(400)`, `unauthorized(401)`, `forbidden(403)`, `badMethod(405)`, `itemNotFound(404)`

#### 请求参数

名字	位置	类型	描述
<code>server_id</code>	<code>body</code>	<code>string</code>	The UUID of the server.
<code>evacuate</code>	<code>body</code>	<code>string</code>	Specify the <code>evacuate</code> action in the request body.
<code>host</code> (Optional)	<code>body</code>	<code>string</code>	The name or ID of the host to which the server is evacuated.
<code>adminPass</code> (Optional)	<code>body</code>	<code>string</code>	An administrative password to access the evacuated or rebuilt instance. To set the administrative password, set the <code>enable_instance_password</code> configuration option to <code>True</code> . If you set this option to <code>False</code> and you try to set the administrative password, the API does not set the password and the response shows a <code>null</code> value for the <code>adminPass</code> parameter.
<code>onSharedStorage</code>	<code>body</code>	<code>boolean</code>	Server on shared storage.
Deprecated in version 2.14			
<code>force</code> (Optional)	<code>body</code>	<code>boolean</code>	Force an evacuation by not verifying the provided destination host by the scheduler.
New in version 2.29			

例如:

```
{
  "evacuate": {
    "host": "b419863b7d814906a68fb31703c0dbd6",
    "adminPass": "MySecretPass",
    "onSharedStorage": "False"
  }
}
```

#### 返回

名字	位置	类型	描述
<code>adminPass</code> (Optional)	<code>body</code>	<code>string</code>	An administrative password to access the evacuated or rebuilt instance. To set the administrative password, set the <code>enable_instance_password</code> configuration option to <code>True</code> . If you set this option to <code>False</code> and you try to set the administrative password, the API does not set the password and the response shows a <code>null</code> value for the <code>adminPass</code> parameter.
Deprecated in version 2.14			



例如:

```
{  
  "adminPass": "MySecretPass"  
}
```

## Class Consul.**Agent**

### **members**(wan=False)

Returns all the members that this agent currently sees. This may vary by agent, use the nodes api of Catalog to retrieve a cluster wide consistent view of members.