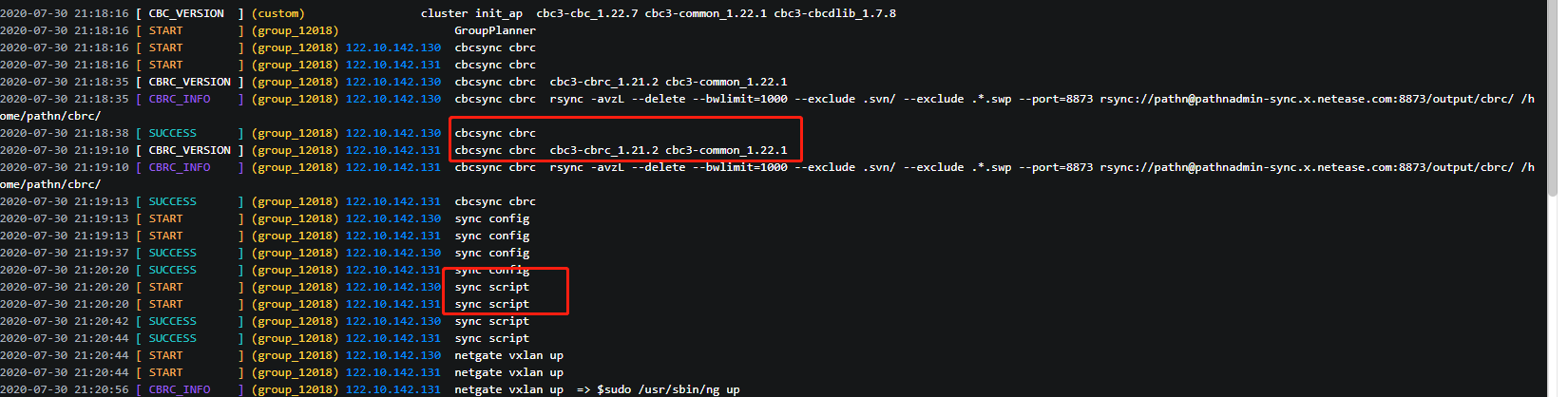
运维规范（内部 SRE 学习）：https://webase.nie.netease.com/

对外文档（给 DEV 看的流程指引）：http://wtdoc.nie.netease.com/

cbc认知：cbc的gitlab配置hooks，修改gitlab代码，会触发hooks同步rsync到管理机；config.py里面CBCSYNC配置，是对应受控机使用的，当在管理机进行cbcsync cbrc ，受控机就会通过这个配置去拉取管理机数据进行rsync操作。



##### CBCSYNC CONFIG ######

CBCSYNC = {

"RSYNC\_BASE\_OPTIONS" : "-avzL --delete --bwlimit=1000 --exclude .svn/ --exclude .\*.swp --port=8873",

"RSYNC\_CBRC\_SRC" : "%s:%s/output/cbrc/" % (RSYNC\_BASE, RSYNC\_PORT),

"RSYNC\_CBRC\_DEST" : "/home/pathn/cbrc/",

}

sync则是cbcmod里面对应的模块

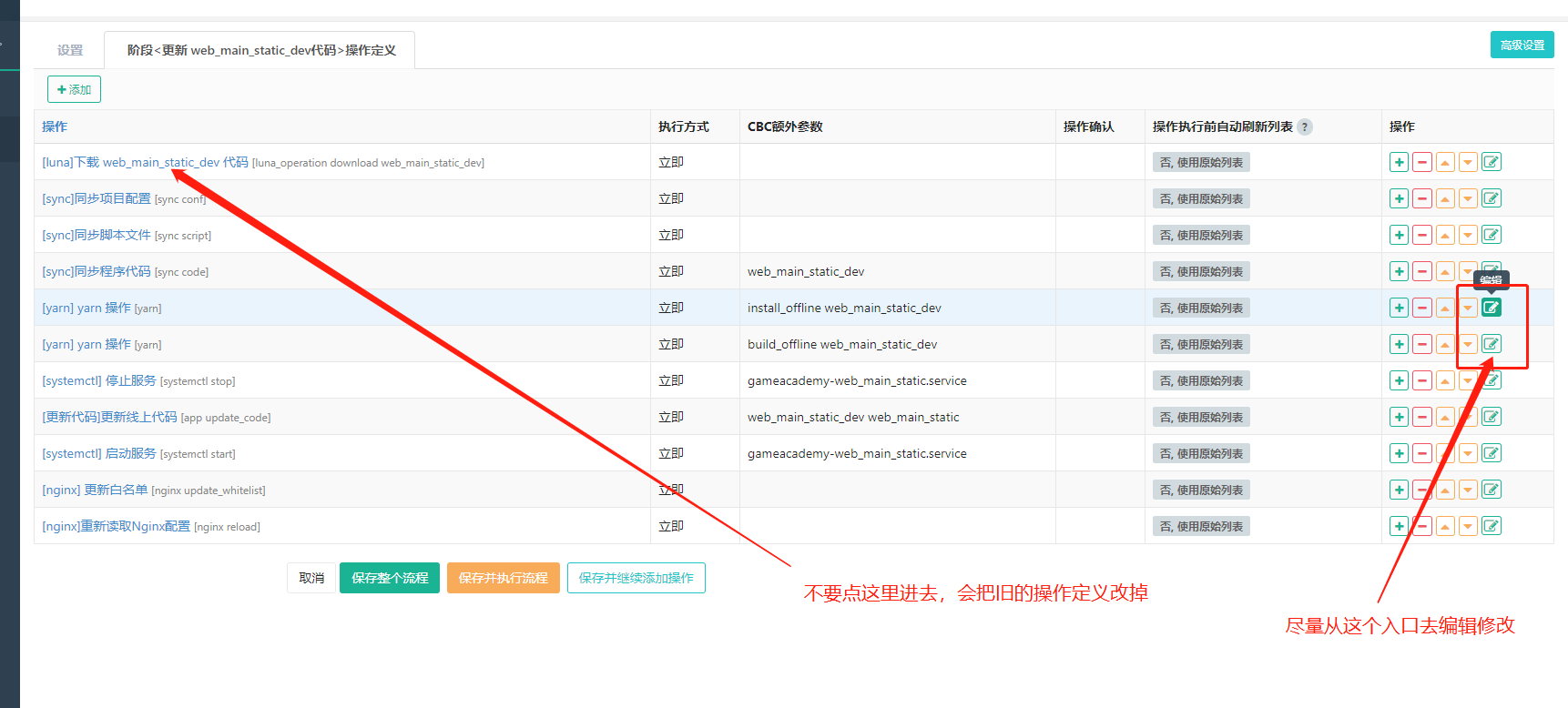
**学院cbrc代码，了解模块功能功能**

<https://git-sa.nie.netease.com/ambitus/gameacademy_cbc3/blob/master/cbc_mods>

**[测试服][web\_static][代码更新]更新代码**

拷贝复制一份，可以看到各阶段的命令和参数，对照luna上传的目录结构，增删部分步骤，例如这个流程里，程序上传的目录里去除了next这一层目录，不再有ssi这个目录，需要修改把添加软连的步骤删掉，next关系到进程启动命令的路径，需要跟程序确认。

复制出来的流量，**不要直接点击步骤里的操作去直接修改**，会把旧的操作定义改掉，影响旧的流程。最后提前编辑好新的操作，再加到新流程里，尽量使用有边框的编辑按钮去操作。



 新建web\_live\_static\_dev流程，对应web\_live\_static\_dev模块需要先在luna添加

<https://aladdin.nie.netease.com/_gameacademy/aladdin/processes/25907/stage0>

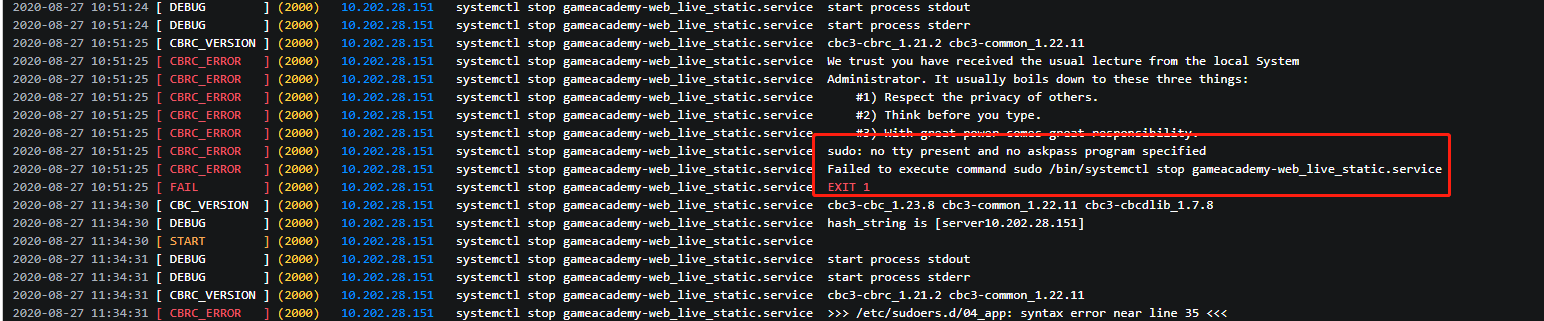
**Puppet添加[gameacademy-web\_live\_static.service](https://git-sa.nie.netease.com/puppet/gameacademy/blob/master/files/conf/systemd/gameacademy-web_live_static.service)**

源文件路径file/conf/systemd/[gameacademy-web\_live\_static.service](https://git-sa.nie.netease.com/puppet/gameacademy/blob/master/files/conf/systemd/gameacademy-web_live_static.service)(参照gameacademy-web\_static.service)

之后在 manifests/roles/app.pp引用

gameacademy::classes::systemctl { 'gameacademy-web\_live\_static.service': unit\_name => 'gameacademy-web\_live\_static.service', unit\_source\_file => 'puppet:///modules/gameacademy/conf/systemd/gameacademy-web\_live\_static.service', }

**阿拉丁流程报错，systemctl stop gameacademy-web\_live\_static.service**



需要puppet修改files/sudoers.d/04\_app，添加gameacademy-web\_live\_static.service的部分内容

gameacademy ALL=(ALL:ALL) NOPASSWD: /bin/systemctl stop gameacademy-web\_live\_static.service

gameacademy ALL=(ALL:ALL) NOPASSWD: /bin/systemctl start gameacademy-web\_live\_static.service

gameacademy ALL=(ALL:ALL) NOPASSWD: /bin/systemctl status gameacademy-web\_live\_static.service

gameacademy ALL=(ALL:ALL) NOPASSWD: /bin/systemctl restart gameacademy-web\_live\_static.service

**[symlink]创建软链 [symlink]----cbc模块配置----这一步在这个流程里不再需要，因为不用ssi这个模块**

先在symlink\_ops.py里定义好def,之后在symlink.py调用

<https://git-sa.nie.netease.com/ambitus/gameacademy_cbc3/blob/master/cbc_mods/symlink_ops.py>

def symlink\_web\_live\_static\_ssi\_assets(self):

"""

Target: /home/gameacademy/running/ssi\_assets

Link\_Name: /home/gameacademy/running/web\_live\_static/ssi/assets

"""

\_target = "/home/gameacademy/running/ssi\_assets"

\_lname = "/home/gameacademy/running/web\_live\_static/ssi/assets"

return self.\_symlink\_dir(\_target, \_lname)

<https://git-sa.nie.netease.com/ambitus/gameacademy_cbc3/blob/master/cbc_mods/symlink.py>

def \_symlink\_web\_live\_static\_ssi\_assets(self):

"""

symlink

"""

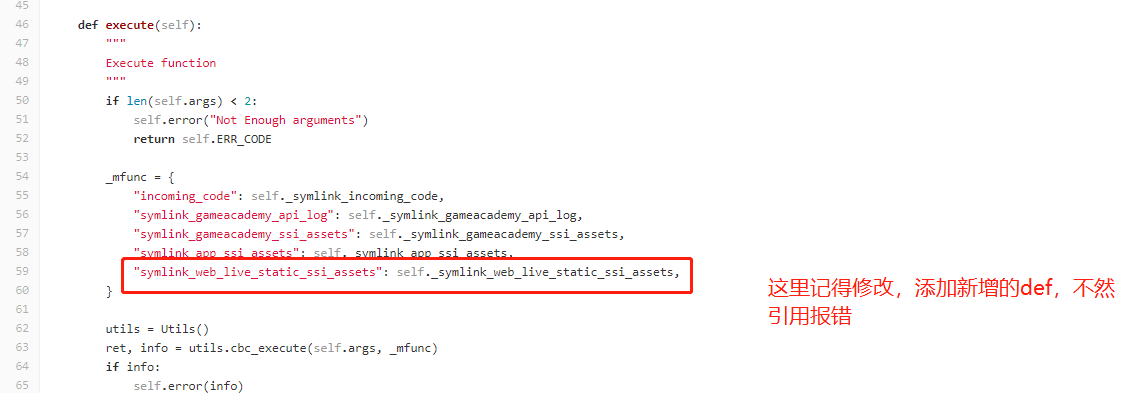
symlinkOPS = SymlinkOPS()

ret, info = symlinkOPS.symlink\_web\_live\_static\_ssi\_assets()

if info is not None:

self.error(info)

return ret



这样就可以通过symlink\_web\_live\_static\_ssi\_assets这个参数在阿拉丁操作建立软连

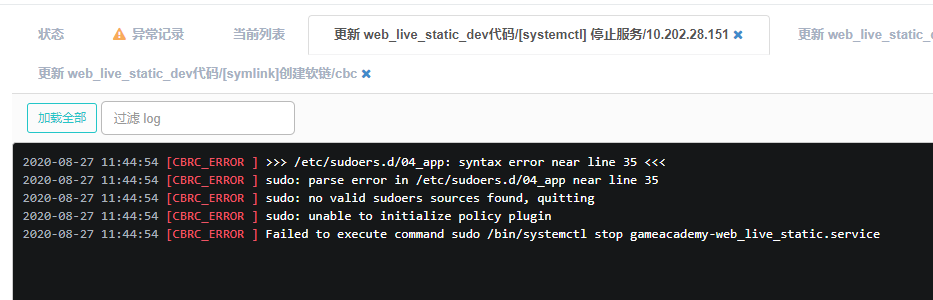
**改了cbrc之后需要上管理机sync cbrc**

**切换到项目用户，cd cbc3/cbc**

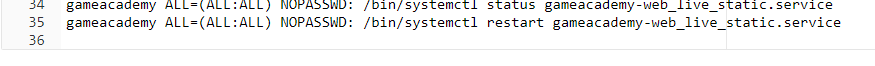
cbc hostlist/list-testing-app sync cbrc

**阿拉丁操作gameacademy-web\_live\_static.service报错**

**是因为Files/sudoers.d/04\_app编辑最后没有回车换行，需要末尾留一行**



SRE|冼镇东 11:59:28

SRE|冼镇东 11:59:58

有部分程序最后一行是要这样换一下行的

SRE|冼镇东 12:00:15

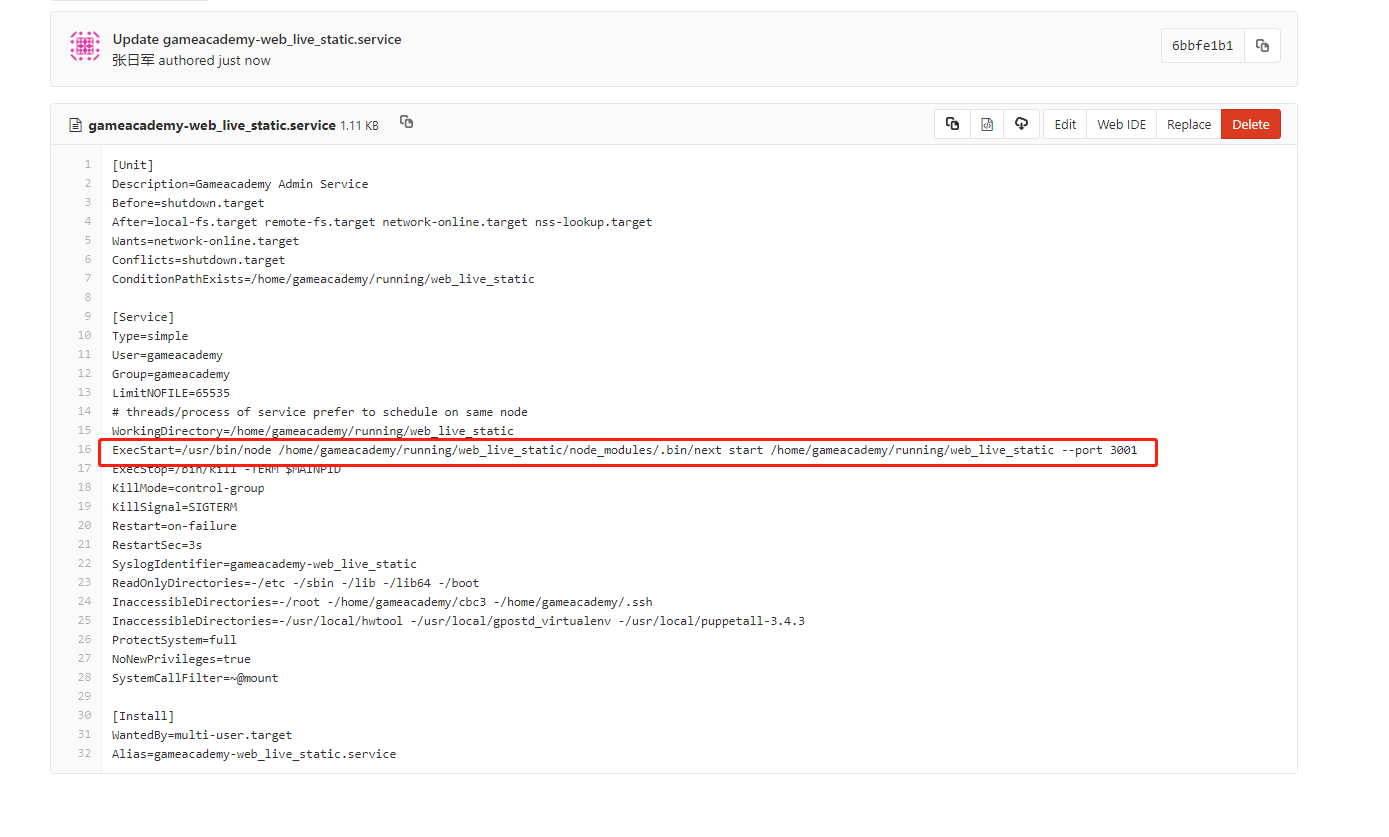
不然就会有No newline at end of file的提示

SRE|冼镇东 12:00:44

应该编辑器里会有结尾自动加空行的功能，建议打开

iptables也是有这个问题的

**gameacademy-web\_live\_static.service启动失败**



按照程序给的命令去启动，之前旧的web\_static里使用的是Environment=PORT=3000定义端口，无法启动。修改为：

ExecStart=/usr/bin/node /home/gameacademy/running/web\_live\_static/node\_modules/.bin/next start /home/gameacademy/running/web\_live\_static --port 3001

不同的应用启动的方式不一样，需要跟程序确认，最好让程序测试过正常再去修改puppet代码

**web\_main\_static新建阿拉丁流程**

叶知(吕泽义) 2020-11-02 15:02:21

node服务端口需要加1，为：3002

nginx上的配置也如之前直播一样：

资源链接代理需要是：game\_academy\_main（如之前直播的game\_academy\_live）

叶知(吕泽义) 2020-11-02 15:06:34

Nginx同时指向的一级路径需要是：/training，（对应之前直播的/live）

**正式环境流程需要配置日志rsyslog**



上次正式环境没有加这三个，配正式环境的时候可以看看



**限定群组范围，限定机器角色**，正式环境和测试环境流程有所不同，需要区分。

测试环境：限定群组范围是testing，机器角色是app；

正式环境：限定群组为production，机器为gameacademy

<https://git-sa.nie.netease.com/puppet/gameacademy/blob/master/files/conf/rsyslog.d/gameacademy-web_main_static.conf>

Puppet报错找不到source文件，没有添加

/home/gameacademy/var/log/gameacademy/web\_main\_static/web\_main\_static.log这个配置，已配上，观察明天邮件汇总----不再报警

**popo添加权限配置**

1,正式环境通过管理机来跳转，测试环境和uat环境就不弄这个了，你们还是可以直接连

2,登陆管理机的用户用workflow的ssh分组管理，这个弄一个独立的分组来做就行

3,业务之间的隔离应该不需要太担心，dumbo地址，库名，账号的密码和权限都是不一样的，误连几率很低

第二步，popo管理添加分组，新建admin01分组

[http://workflow2.x.netease.com/#/popo/start\_process/server\_ssh\_zh](http://workflow2.x.netease.com/" \l "/popo/start_process/server_ssh_zh)

之后修改管理机的admin.pp, puppet的admin模块用user模块引入一下对应的分组

user模块说明https://g.126.fm/01Z9lOE

应该可以简单些

# Add user

class { users :

project => $project\_user,

role => 'admin01', ##设置其它角色会自动包含SA账号

usercheck\_ensure => present,

}

或者在后面添加include ::users::popo::admin01（分组）

# Add user

class { users :

project => $project\_user,

usercheck\_ensure => present,

role => 'gs', ##会添加gs这个分组，同时添加sa分组

devgroup => ['root']

}

include ::users::popo::vdev ###添加vdev这个分组

**Popo接入pathn海外加速数据项**

<https://galaxy.nie.netease.com/_popo/galaxy/dataentrys/11717/content>

{% $LBC\_IP = [

"106.2.95.77" => ["port" => ["80","8080"]],

"42.186.122.63" => ["port" => ["80","8080"]],

"106.2.69.242" => ["port" => ["80","8180","443"]],

"42.186.122.57" => ["port" => ["80","8180","443"]],

"42.186.241.16" => ["port" => ["80","443"]],

"59.111.137.135" => ["port" => ["80","443"]],

"42.186.175.67" => ["port" => ["80","443"]],

"42.186.235.145" => ["port" => ["443"]]

]

%}

{% foreach $LBC\_IP as $ip => $portlist %}

{% foreach $portlist.port as $port %}

{% $item = [] %}

{% $item['ip'] = $ip %}

{% $item['external\_ip'] = $ip %}

{% $item['port'] = $port %}{\* 此处为您的端口 \*}

{% $item['external\_port'] = $port %}

{% $item['enable\_public'] = 1 %}

{% $item['label'] = 'hangzhou' %}{\* 此处为服务器所在区域，如california, tokyo \*}

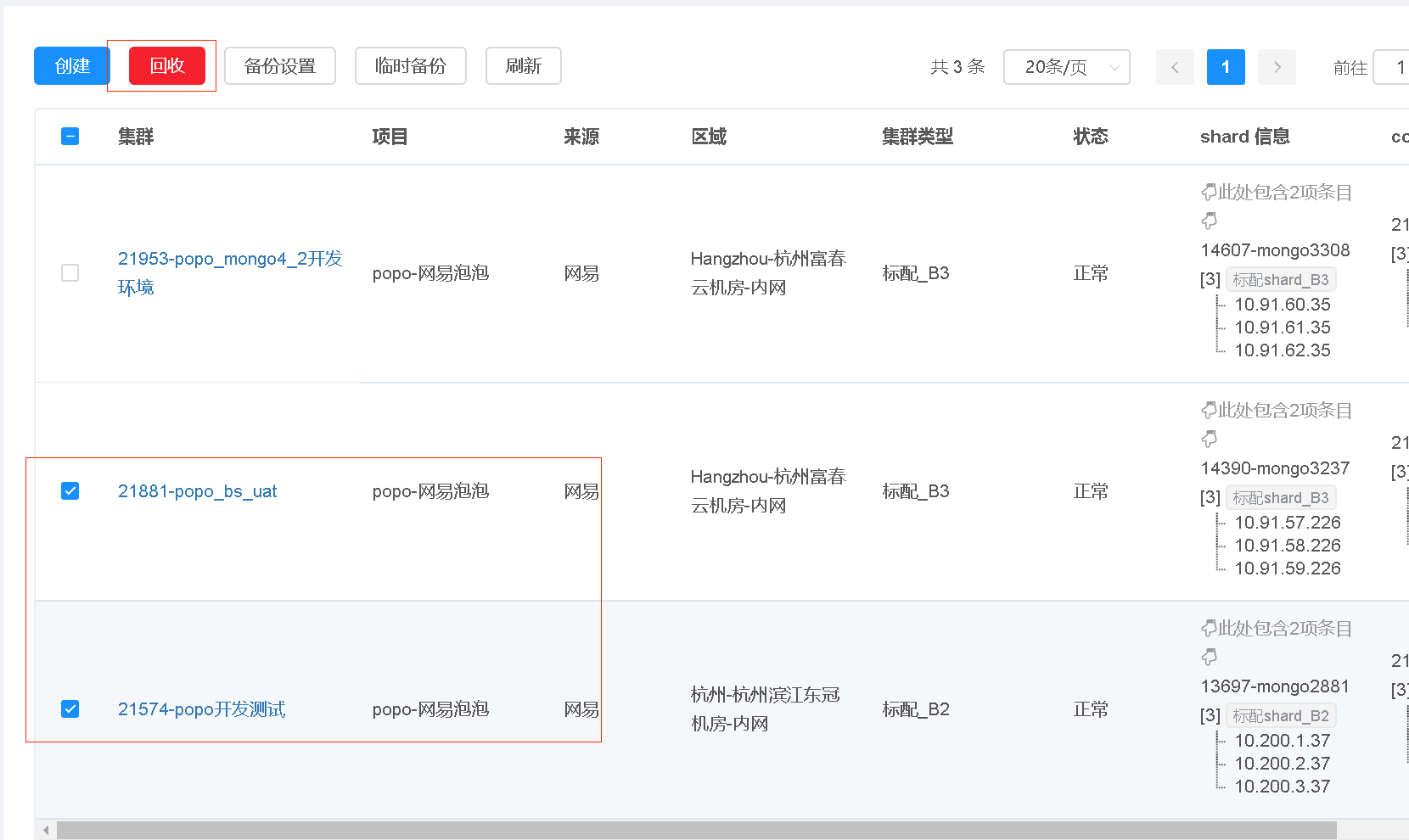
{% $EXT\_PATHN\_PROJECT\_MAP[] = $item %}

{%/foreach%}

{%/foreach%}

**Popo回收旧MongoDB实例**

<http://jupiter.nie.netease.com/popo/ticket_list>



跟业务确认后，在ocean系统申请回收即可

**Popo正式环境MongoDB申请配置**

[Ocean管理系统][正式][通知]

[创建] 21980-popo\_im消息状态生产环境集群 创建集群正在进行中, 详细信息如下：

集群ID：21980

区域：杭州富春云机房

实例类型：高配shard\_V2

引擎配比：3 WiredTiger

mongo版本：4.2.8

操作者：冼镇东

Shard数量：2

**GCP cld小小云实例升级内存（@duty cld找值班处理）**

SRE|冼镇东 14:52:49

@SRE|张日军 那大佬跟一下这台机升级下内存？10.212.9.67

SRE|冼镇东 14:53:02

现在是8c16g升级到8c32g

gcp的测试环境一般都是程序直接管环境的，重启之后关一下puppet，其他应该不用管什么

----机器应该是容器，无法systemctl status puppet（没有启动init进程），应该也没有安装puppet

[http://workflow2.x.netease.com/#/gcp/start\_process/xin\_liu\_cheng\_test](http://workflow2.x.netease.com/" \l "/gcp/start_process/xin_liu_cheng_test)

一般没有8核32G的配置，需要选择其他需求，补充说明，升级过程需要大概十分钟，关机后提前通知小小云值班操作。

**ACP 10.214.8.8小小云升级实例**

测试机程序反映cpu消耗过多，要求升级cpu，原本4core\_8GB升到8core\_16GB

机器上没有加key，直接在skyline前端操作关机，值班那边需要再次系统关机

**小小云不同机房网段情况**

Cloud值班1-欧阳俊斌 15:27:35

可自己操作重启网段：

开发机：

- 10.212.xx.xx

https://skyline-office.nie.netease.com/cld/instances

IDC ：

- 10.202.xx.xx 麒麟座

- 10.214.xx.xx 鹿豹座

https://skyline.nie.netease.com/cld/instances

无法自助重启网段：

- 192.168.xx.xx

- 10.246.xx.xx

- 10.82.xx.xx

Cloud值班1-欧阳俊斌 15:27:39

开发机：

- 10.212.xx.xx

<https://skyline-office.nie.netease.com/cld/instances>

Popo小小云

10.82是测试服 ##idc机房，无法在skyline前端查看

10.212是开发服 ##skyline-office

**通过小蜜查询实例信息**

10.82这些段属于旧集群，skyline前端看不了

@cld 10.82.8.20

Symphony值班 2020-11-13 10:52:49

开发机只能访问开发集群的虚拟机或者容器， 10.212. 或者10.246 还有楼下 192.168.40.xx的网段

Symphony值班 2020-11-13 10:53:10

其他都不通， 如果 容器和容器， 或者容器和虚拟机不通，可能就需要添加peering

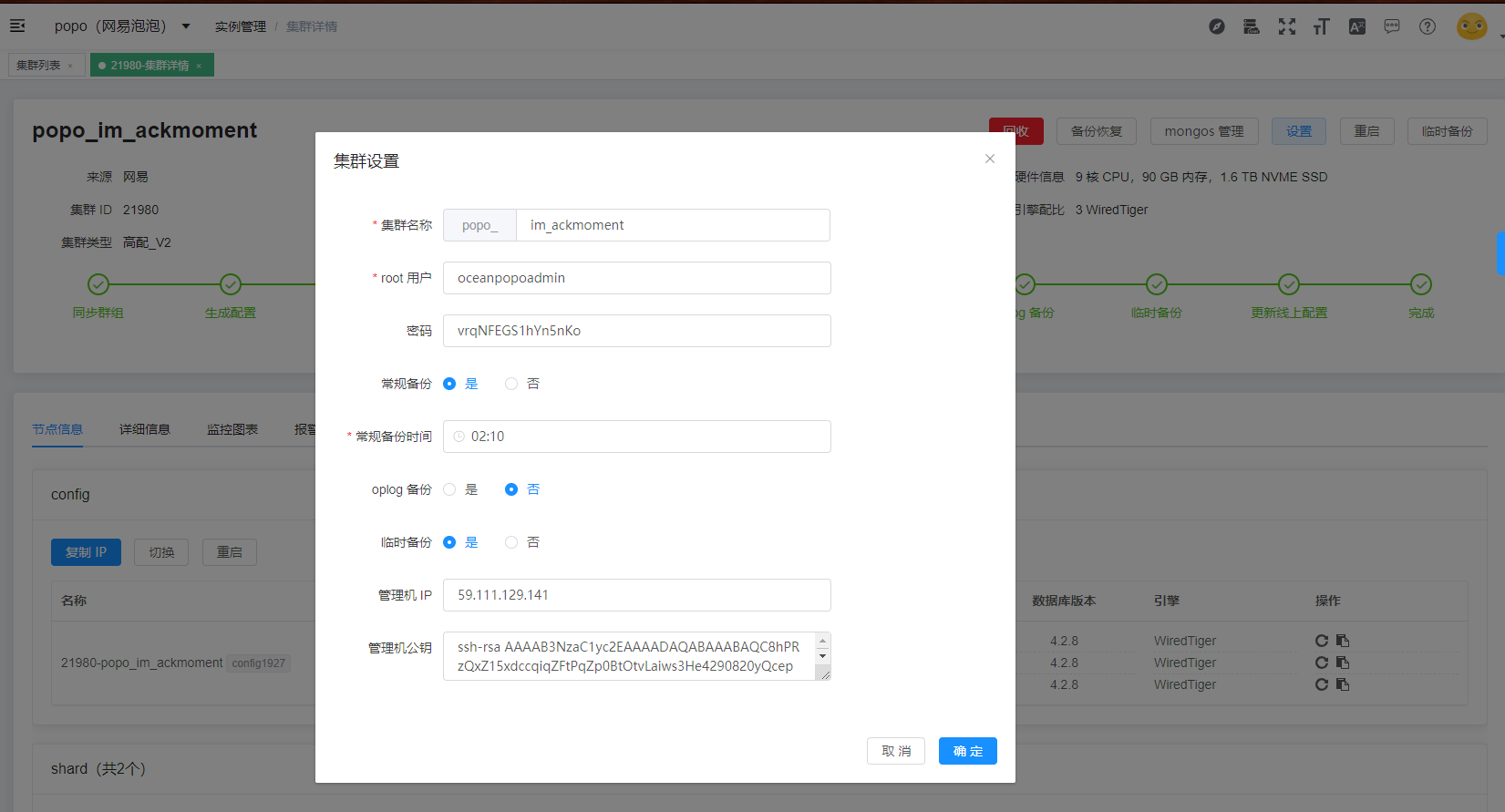
**容器内存占用高-men\_used\_not\_include\_cache**

不含 cache 的内存都一直很高的话，建议调高内存

Symphony值班 16:28:54-----OOM 或者容器内部服务异常会重建

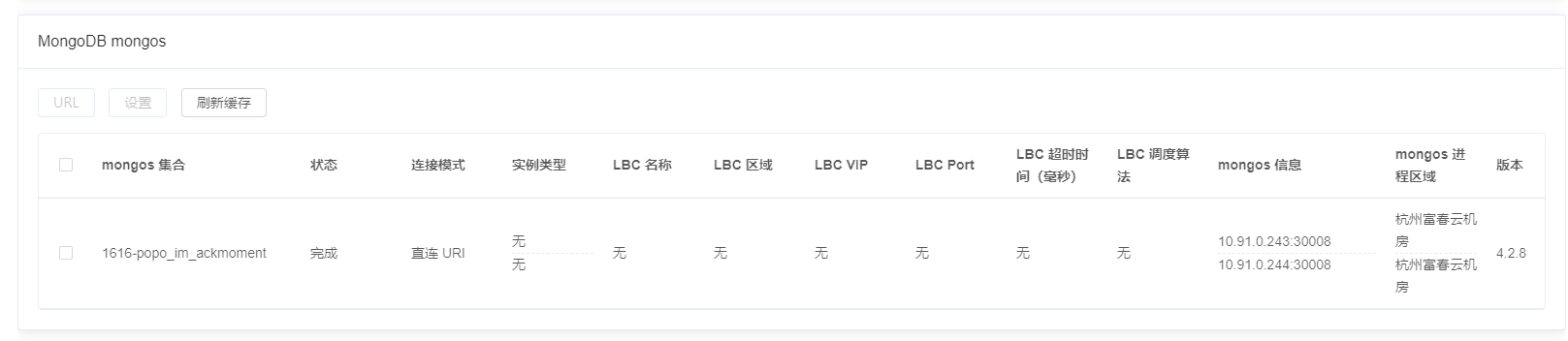
不然一旦没有 cache 可以释放，容器使用的内存到达 limitmem，就会导致容器 OOM

**Popo mongo修改管理用户密码**



popo\_im\_ackmoment实例在ocean系统修改了密码后，需要到登录到数据库同步修改密码；

连接方式，一般使用**管理机10.82.112.24**去连接**mongos的ip和端口**（一开始在办公电脑的虚拟机去连，一直连不上）



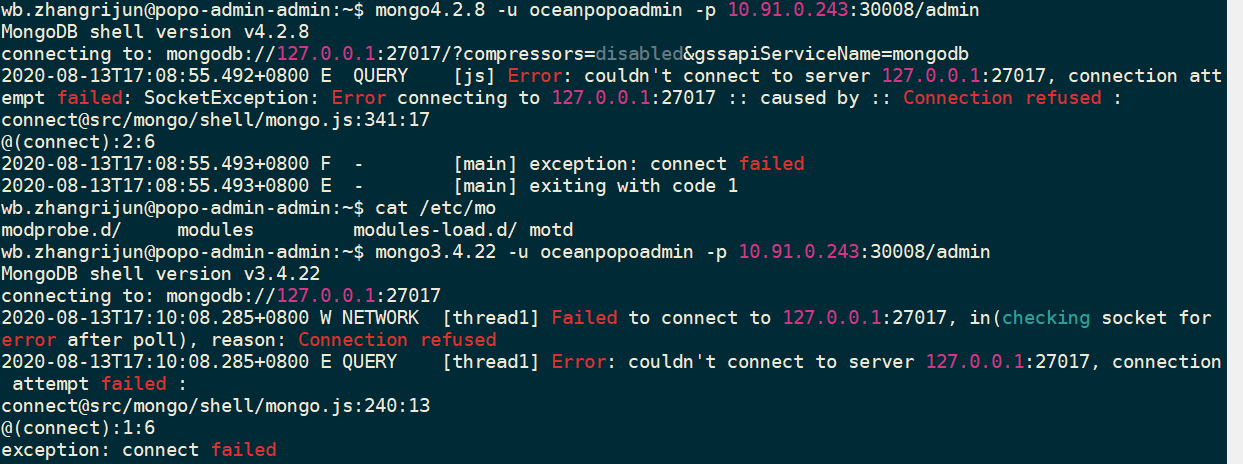
mongo4.2.8 10.91.0.243:30008/admin -u oceanpopoadmin 使用旧的密码登录，db.system.users.find()可以查看用户表里的内容。

mongos> db.changeUserPassword("oceanpopoadmin","vrqNFEGS1hYn5nKo");

mongos> db.auth("oceanpopoadmin","vrqNFEGS1hYn5nKo")

1 ###正常返回1验证修改成功

如果是使用 mongo4.2.8 -u oceanpopoadmin -p 10.91.0.243:30008/admin去登录会报错，会提示登录到本机的127.0.0.1，需要修改远程mongos的主机ip端口的写的位置，紧跟mongo命令后面。



**mongos对应用程序是什么角色？**

对应用程序来说，mongos可以看做是代理，跟直接访问mongo集群一样。

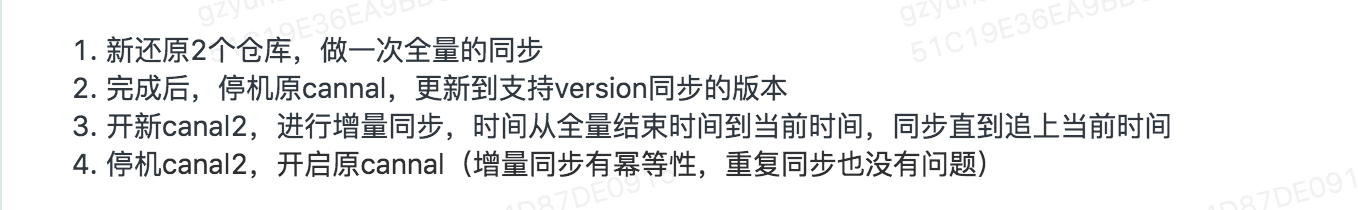
**数据操作在哪个节点决策？ 决策类都在mongos上进行，包括数据整合。**

普通操作：例如新增一条数据，mongos会根据shard key计算出这条数据该存哪个shard，然后直接访问这个shard将数据写入。

条件式操作：例如根据条件查询并对结果排序，处理过程是 mongos根据查询条件广播到个shard节点获取结果并排序再返回给应用程序。

**gcp回收旧mysql实例(先关闭，到第二天根据情况再回收，防止cron等问题)**





SRE|张日军 17:56:55

明天新的临时实例需要怎么搞，直接申请新实例？

SRE|冼镇东 17:58:26

SRE|冼镇东 17:58:48

gcp的主库是这两个，要分开的原因是一个实例装不下了

SRE|冼镇东 17:59:51

申请一个硬盘容量能装下源实例数据的基础实例->用源实例的备份恢复到临时实例

SRE|冼镇东 18:00:27

SRE|冼镇东 18:00:53



SRE|冼镇东 18:01:06

有时也会要恢复到指定的时间点

SRE|冼镇东 18:01:11

反正你看看就行

SRE|冼镇东 18:01:40

不过一定要小心不要恢复到其他的实例上去了

SRE|冼镇东 18:01:58

数据库一乱整个服务没了

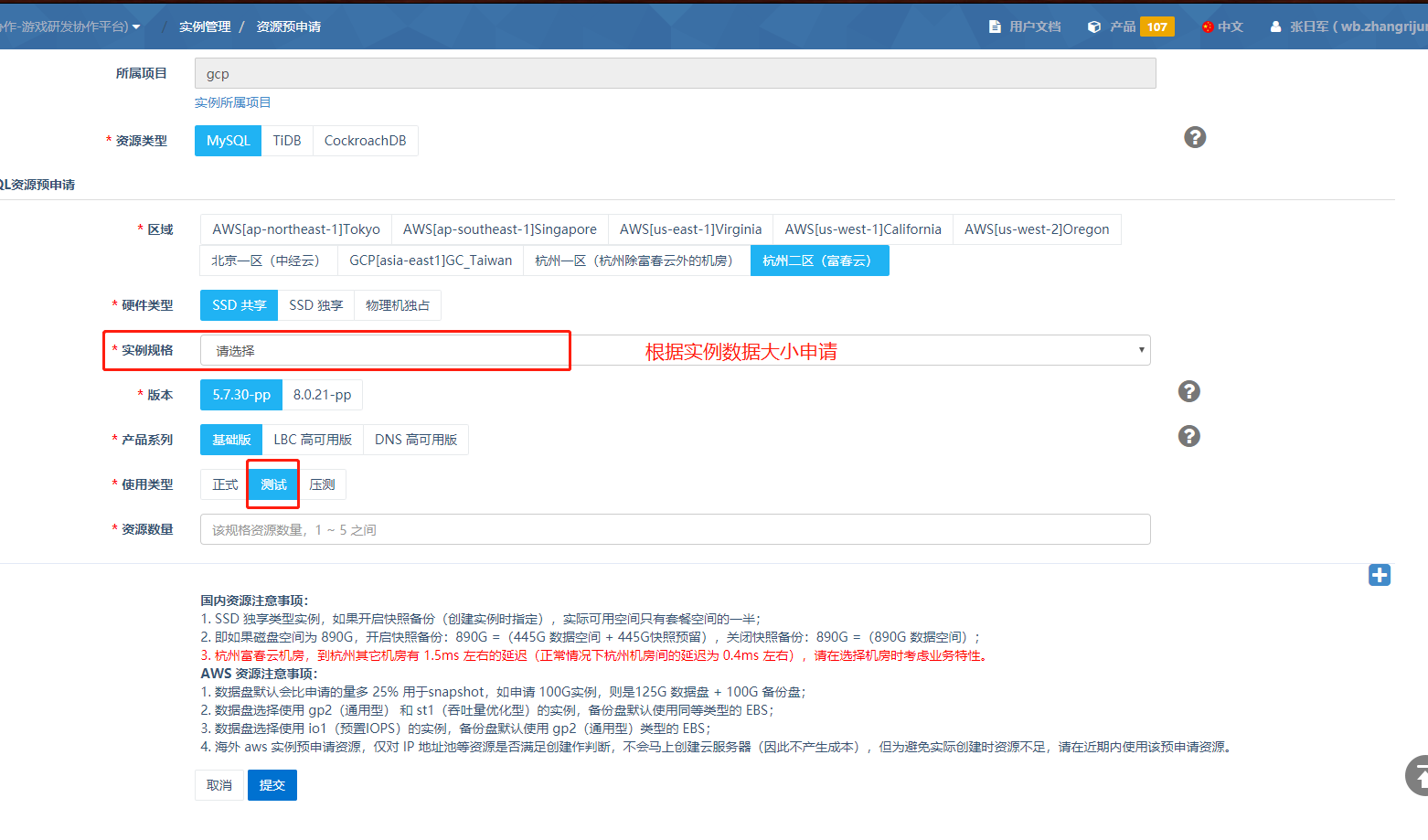
SRE|冼镇东 18:02:30

就用这个恢复到新申请的临时实例就行

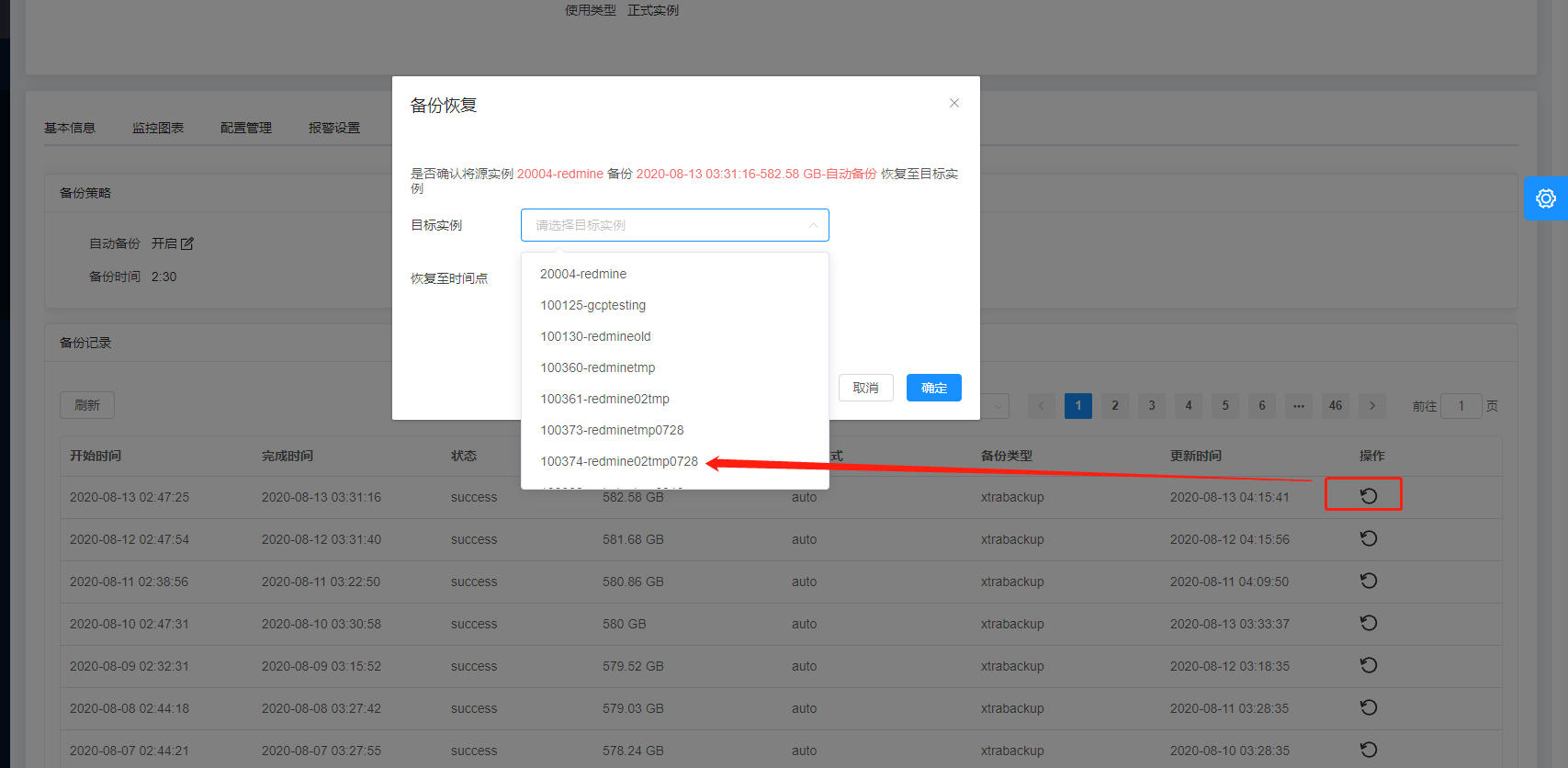
<https://dumbo.nie.netease.com/_gcp/instances/list>旧的平台

<https://db.nie.netease.com/mysql/_gcp/resources/create>新的统一数据库平台----两个平台操作界面略有不同。

先查看数据库大小，对应的申请不少于该数据大小的新实例，资源预申请--->创建实例（测试类型，采用实例名+tmp+日期命名如redminetmp0813，机房一般都可以）



数据库实例----备份管理---备份记录，选择新建的实例，时间留空则恢复到备份文件对应的时间点。90G左右需要恢复一个小时



可以看看企业平台关于gcp和gameacademy的文档做下了解

<https://dumbo-office.nie.netease.com/_gcp/instances/tidb>数据库的管理平台，对应一个office的内网访问的环境，分布在高唐机房，只允许办公区访问，savpn无法访问。

**游戏学院sym vpc出口ip**

企业平台|服务端|史梦杰 15:25:02

@SRE|冼镇东 东哥，我们游戏学院现在改成docker部署了，导致docker的IP不在 corp 那边的白名单，有办法把镜像改成固定的出口IP吗？能改的话，后续扩展受影响吗？

SRE|冼镇东 15:25:21

corp是外网入口么？

学院的sym vpc在豺狼座----https://sym2.nie.netease.com/#/symphony/networking/eip ##sym2系统需要先切换到对应的集群才能看到项目对应的实例。

**外网ip在网络管理---网络配置---EIP里查看，对应不同子网vpc和不同环境**

**只要在这个网段内，eip都是不会变的，两个vpc**



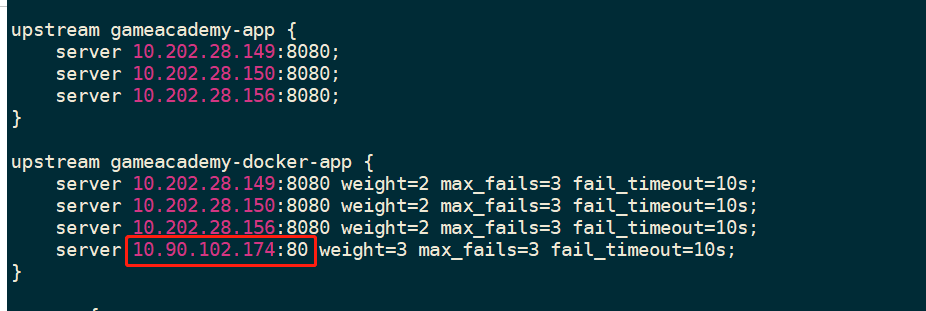
**学院gameacademy，目前docker化，nginx下配置了docker入口**https://g.126.fm/007EmJG

一个lbc，两个前端nginx，三个app

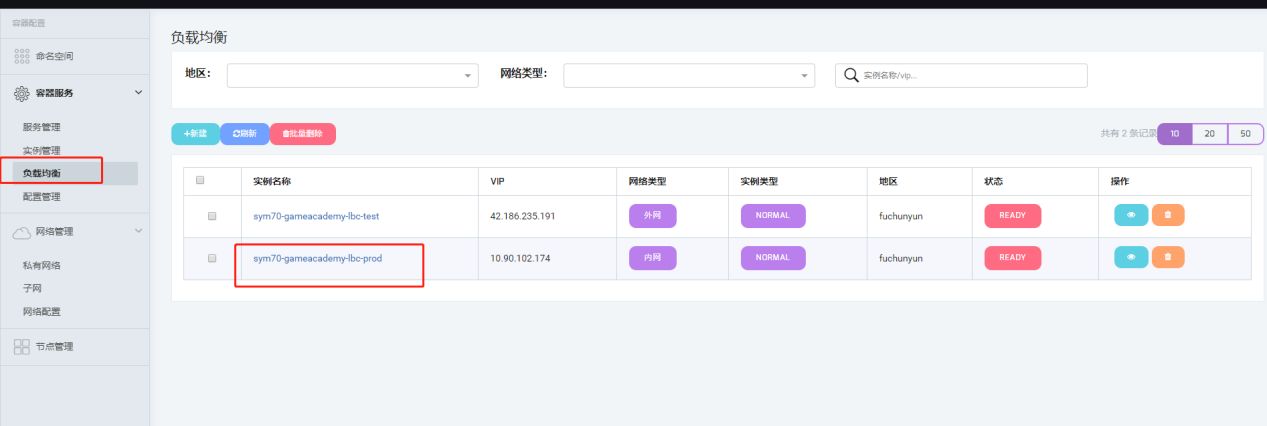
用户---lbc--10.82.8.13/10.82.8.14两台nginx前端----10.202.28.149/150/156三台app

ps -ef |grep java 查看机器上进程，app机器上通过java进程承载了主要的业务

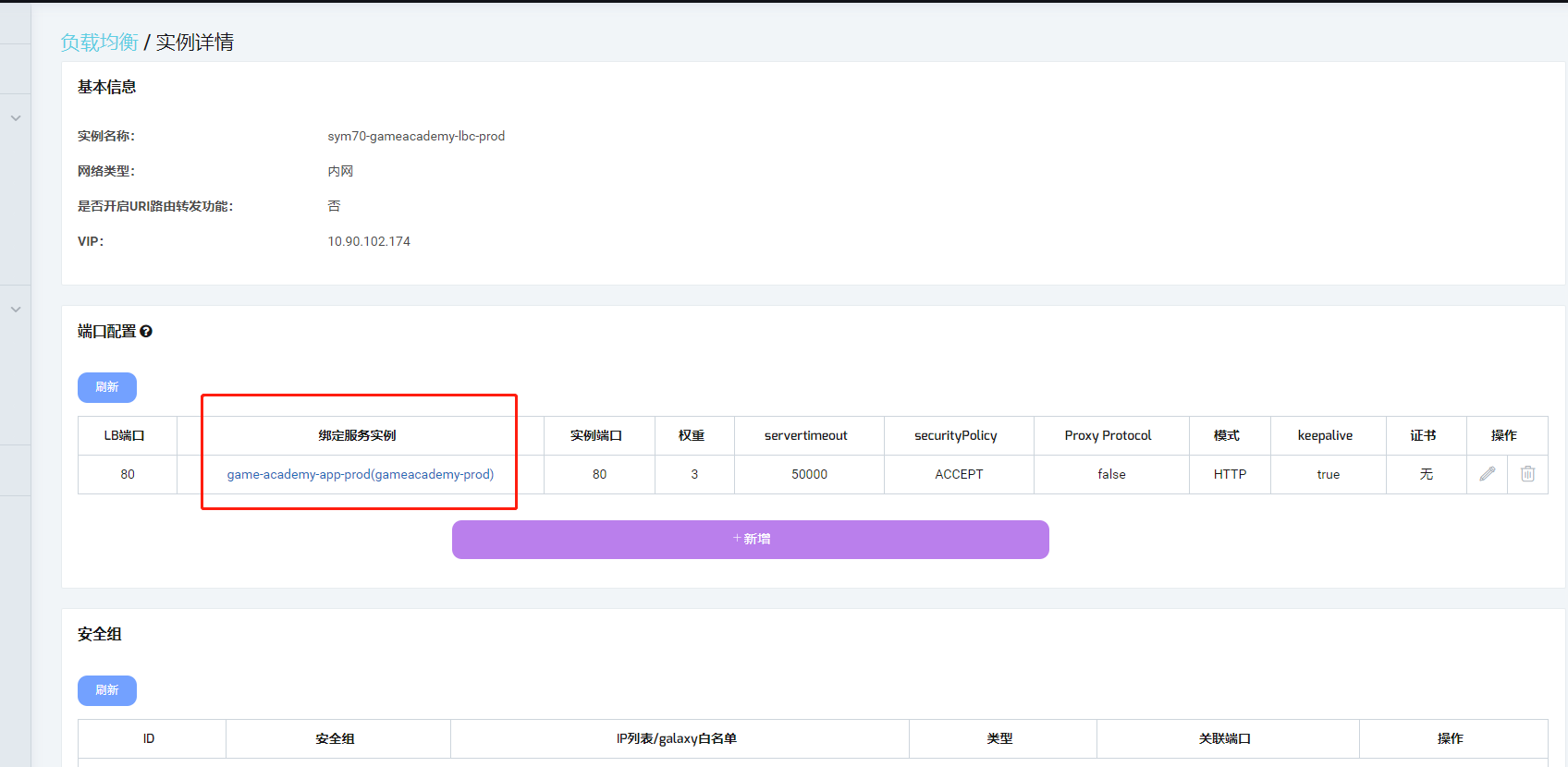
10.82.8.13例如这台，配置在/etc/nginx/sites-enabled# cat gameacademy.conf（galaxy上配置管理- gameacademy.conf）



10.90.102.174这个是docker的负载均衡入口ip（lbc实例sym70-gameacademy-lbc-prod）



通过绑定的服务可以查看是rs为10.213.20.3/4/7这三个容器



**学院直播业务，使用了popo项目的中台服务，由nginx代理到open.popo.netease.com实现**

配置部分：

location ~ ^/live\_api/live\_app/(.\*)$ {

set $x\_openpopo\_domain open.popo.netease.com;

proxy\_pass https://$x\_openpopo\_domain/open-api/live/app/$1$is\_args$args;

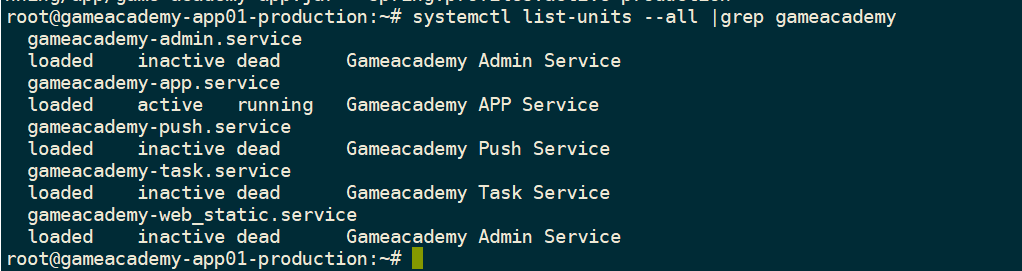
proxy\_set\_header Host $x\_openpopo\_domain;

proxy\_set\_header X-Real-IP $remote\_addr;

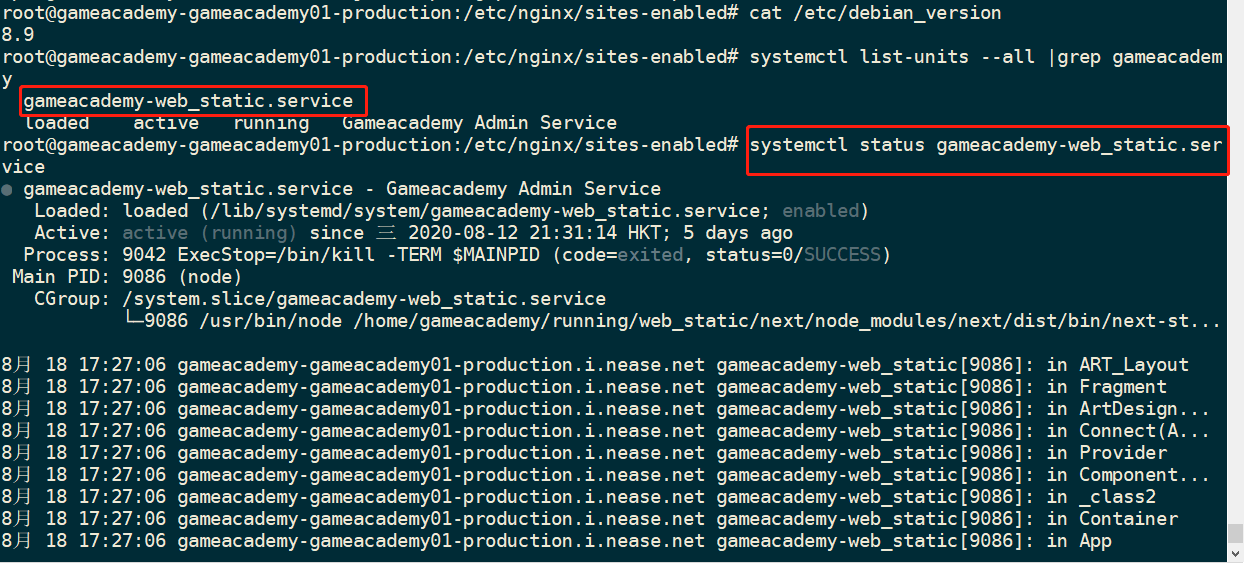
}

**服务主要使用oracle jdk8的spring boot，使用systemd包装java启动命令来控制服务，systemctl start/stop/restart gameacademy-xxxx**。

**app机器**上管理的服务通过systemctl list-units --all |grep gameacademy来查询，主要有app，admin，push，task，web\_static几类服务。



**前端机器**管理的服务gameacademy-web\_static.service



**app接入pathn是使用了popo项目的资源，所以增加pathn入口要在popo项目下做配置。**

popo项目下数据项EXT\_PATHN\_PROJECT\_MAP里面，这个ip为学院lbc入口实例 gameacademy的ip，转发到前端。

"59.111.137.135" => ["port" => ["80","443"]],

`Lbc有些旧的实例，目前很少流量，用nginx配置转发，比较少用到，影响很小`

前端两台nginx配置/etc/nginx/sites-enabled/gameacademy.conf白名单设置：

server {

listen 80;

listen 8011 proxy\_protocol;

server\_name game.academy.163.com ios-game.academy.163.com api.academy.163.com;

# LBC IP List

include /etc/nginx/whitelist/enable\_lbc\_proxy\_protocol.conf;

**10.202.28.149app01-production机器nginx里proxy\_pass配置**

upstream gameacademy-app {

server 127.0.0.1:8080;

server 10.202.28.150:8080 backup;

server 10.202.28.156:8080 backup;

}

server {

listen 80;

listen 8011 proxy\_protocol;

server\_name app.game.academy.netease.com app.academy.163.com api.academy.163.com academy-api.netease.com academy.netease.com academy.163.com;

keepalive\_timeout 0s;

# LBC IP List

include /etc/nginx/whitelist/enable\_lbc\_proxy\_protocol.conf;

client\_max\_body\_size 1024m;

location / { ##通用匹配任何未匹配到其它location的请求都会匹配到，相当于switch中的default

proxy\_pass <http://gameacademy-app;> ##访问http://server\_name里的域名，就会转发到upstream gameacademy-app里面的几个server里。

proxy\_set\_header Host $http\_host;

proxy\_set\_header X-Real-IP $remote\_addr;

}

}

**gameacademy-gameacademy01-production-10.82.8.13机器的nginx配置**

<https://galaxy.nie.netease.com/_gameacademy/galaxy/configtpls/41775/preview>

proxy\_pass + upstream（显式） + resolver（变量设置域名）

<https://www.jianshu.com/p/5caa48664da5>说明文档

访问[http://localhost/foo](http://localhost/foo" \t "_blank)时，upstream模块会优先查找是否有定义upstream后端服务器，如果有定义则直接利用，不再走DNS解析。所以proxy模块会将请求转发到127.0.0.1的8001端口上。

server {

listen 80;

server\_name www.xxx.com;

location /aming/

{

proxy\_pass http://192.168.1.10:8080/linux/;

proxy\_set\_header host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

}

}

语法：proxy\_set\_header field value;

field ：为要更改的项目，也可以理解为变量的名字，比如host

value ：为变量的值

如果不设置proxy\_set\_header，则默认host的值为proxy\_pass后面跟的那个域名或者IP（一般写IP）

proxy\_set\_header X-Real-IP $remote\_addr;

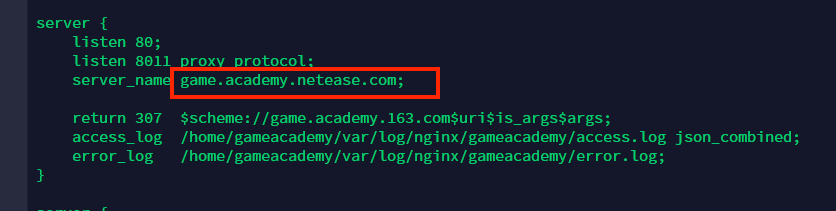
和

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

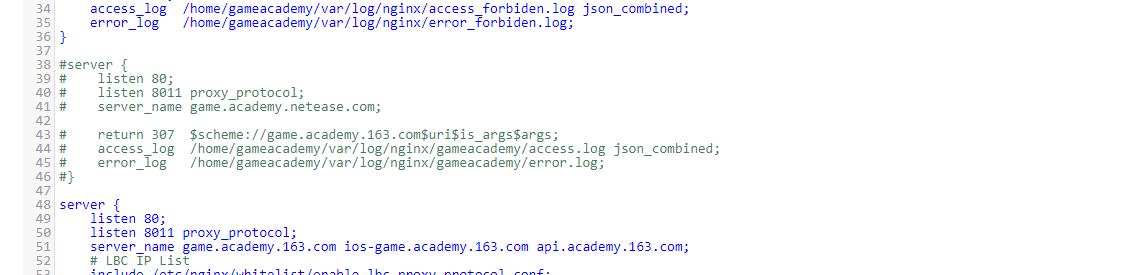
用来设置被代理端接收到的远程客户端IP，如果不设置，则header信息中并不会透传远程真实客户端的IP地址。

**游戏学院有一个 CRLF 漏洞： [https://soc.nie.netease.com/#/vuln/user-hole?id=6795](https://soc.nie.netease.com/#/vuln/user-hole?id=6795，)**[，](https://soc.nie.netease.com/#/vuln/user-hole?id=6795，)

我们今天排查了一下，是因为 nginx 里的这个配置导致的：

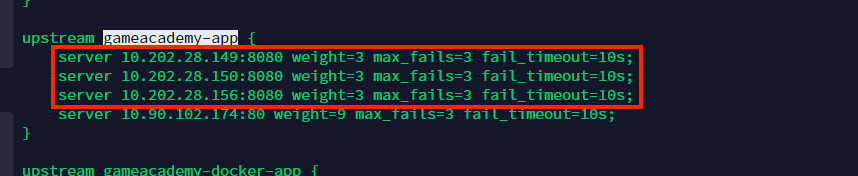


需要修改两台nginx机器的配置https://galaxy.nie.netease.com/\_gameacademy/galaxy/configtpls/41775/preview



暂时注销掉，galaxy同步之后阿拉丁同步nginx配置流程即可。

**游戏学院 web服务器（10.82.8.13、10.82.8.14）的 nginx 配置，把里面这一部分注释掉**



<https://galaxy.nie.netease.com/_gameacademy/galaxy/configtpls/41775/params>

学院有个更新流程，其实就是改了galaxy上跑这个流程就行了



<https://aladdin.nie.netease.com/_gameacademy/aladdin/processes/23905/execute>

# **location 匹配规则**

location [=|~|~\*|^~] /uri/ { … }

| **模式** | **含义** |
| --- | --- |
| location = /uri | = 表示精确匹配，只有完全匹配上才能生效 |
| location ^~ /uri | ^~ 开头对URL路径进行前缀匹配，并且在正则之前。 |
| location ~ pattern | 开头表示区分大小写的正则匹配 |
| location ~\* pattern | 开头表示不区分大小写的正则匹配 |
| location /uri | 不带任何修饰符，也表示前缀匹配，但是在正则匹配之后 |
| location / | 通用匹配，任何未匹配到其它location的请求都会匹配到，相当于switch中的default |

前缀匹配时，Nginx 不对 url 做编码，因此请求为 /static/20%/aa，可以被规则 ^~ /static/ /aa 匹配到（注意是空格）

<https://moonbingbing.gitbooks.io/openresty-best-practices/content/ngx/nginx_local_pcre.html>

**易协作相关**

SRE|冼镇东 18:08:25

https://g.126.fm/01lJVXa

SRE|冼镇东 18:08:32

易协作的架构基本还是这样

SRE|冼镇东 18:08:48

一个gcp，一个acp

SRE|冼镇东 18:09:00

acp是美术专用的协作平台

SRE|冼镇东 18:09:10

都是nginx+passenger的

SRE|张日军 18:09:35

1

SRE|冼镇东 18:09:46

你可以需要了解下passenger，这个东西你跑nginx reload业务是会中断的

https://g.126.fm/00cOylO这个是部署文档

**gcp 重要配置haproxy.conf 全局代理机器gcp-haproxy03-3000&gcp-haproxy04-3000**

/MachineConfig/{%$host.ip%}/template/haproxy.template

/MachineConfig/{%$host.ip%}/template/nginx.template

重要的配置模板参考

**Haproxy.conf配置说明**

haproxy 的配置文件由两部分组成：全局设定和对代理的设定，共分为五段：global，defaults，frontend，backend，listen

1.global: (全局配置主要用于设定义全局参数，属于进程级的配置，通常和操作系统配置有关)

2.default : (配置默认参数，这些参数可以被用到frontend，backend，Listen组件)

在此部分中设置的参数值，默认会自动引用到下面的frontend、backend、listen部分中，因引，某些参数属于公用的配置，只需要在defaults部分添加一次即可。而如果frontend、backend、listen部分也配置了与defaults部分一样的参数，Defaults部分参数对应的值自动被覆盖。

3.frontend：( 接收请求的前端虚拟节点，Frontend可以更加规则直接指定具体使用后端的backend)

frontend是在haproxy 1.3版本以后才引入的一个组件，同时引入的还有backend组件。通过引入这些组件，在很大程度上简化了haproxy配置文件的复杂性。forntend可以根据ACL规则直接指定要使用的后端backend

4.backend : (后端服务集群的配置，真实服务器，一个Backend对应一个或者多个实体服务器)

在HAProxy1.3版本之前，HAProxy的所有配置选项都在这个部分中设置。为了保持兼容性，haproxy新的版本依然保留了listen组件配置试。两种配置方式任选一中

5.Listen : (Fronted和backend的组合体) 比如haproxy实例状态监控部分配置

**frontend部分配置说明**

frontend是在haproxy 1.3版本以后才引入的一个组件，同时引入的还有backend组件。通过引入这些组件，在很大程度上简化了haproxy配置文件的复杂性。frontend根据任意 HTTP请求头内容做ACL规则匹配,然后把请求定向到相关的backend

frontend http\_80\_in #定义一个名为http\_80\_in的frontend

bind 0.0.0.0:80 #设置监听端口，即haproxy提供的web服务端口，和lvs的vip 类似

mode http #http 的7层模式

log global #应用全局的日志设置

option httplog #启用http的log

option httpclose #每次请求完毕后主动关闭http通道，HAproxy不支持keep-alive模式

option forwardfor #如果后端服务器需要获得客户端的真实IP需要配置此参数，将可以从HttpHeader中获得客户端IP

default\_backend wwwpool #设置请求默认转发的后端服务池

**backend部分配置说明**  
用来定义后端服务集群的配置，真实服务器，一个Backend对应一个或者多个实体服务器

backend wwwpool #定义wwwpool服务器组。

mode http #http的7层模式

option redispatch

option abortonclose

balance source #负载均衡的方式，源哈希算法

cookie SERVERID #允许插入serverid到cookie中，serverid后面可以定义

option httpchk GET /test.html #心跳检测

server web1 10.1.1.2:80 cookie 2 weight 3 check inter 2000 rise 2 fall 3 maxconn 8

**server:用来定义多台后端真实服务器,不能用于defaults和frontend部分,格式为:server name address:port param\***

**acl规则**

<https://www.cnblogs.com/linuxzkq/p/4927285.html>

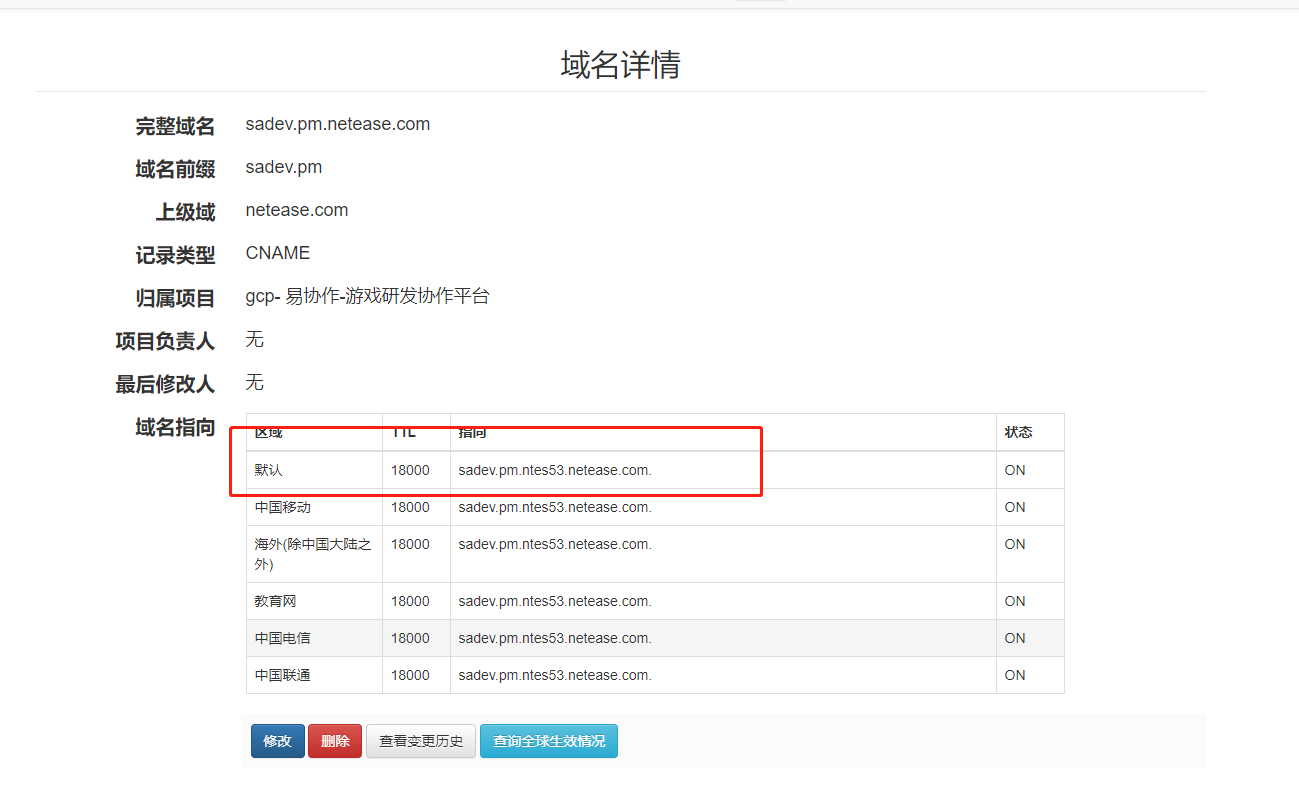
 haproxy的ACL用于实现基于请求报文的首部、响应报文的内容或其它的环境状态信息来做出转发决策，这大大增强了其配置弹性。其配置法则通常分为两 步，首先去定义ACL，即定义一个测试条件，而后在条件得到满足时执行某特定的动作，如阻止请求或转发至某特定的后端。定义ACL的语法格式如下。  
  
  acl <aclname> <criterion> [flags] [operator] <value> ...  
  
  <aclname>：ACL名称，区分字符大小写，且其只能包含大小写字母、数字、-(连接线)、\_(下划线)、.(点号)和:(冒号)；haproxy中，acl可以重名，这可以把多个测试条件定义为一个共同的acl；  
  <criterion>：测试标准，即对什么信息发起测试；测试方式可以由[flags]指定的标志进行调整；而有些测试标准也可以需要为其在<value>之前指定一个操作符[operator]；  
  [flags]：目前haproxy的acl支持的标志位有3个：  
    -i：不区分<value>中模式字符的大小写；  
    -f：从指定的文件中加载模式；  
    --：标志符的强制结束标记，在模式中的字符串像标记符时使用；

一个域名对应一个实例https://g.126.fm/01lJVXa

例如实例<http://sadev.pm.netease.com/>

在域名系统里查找sadev.pm.netease.com这个域名指向sadev.pm.ntes53.netease.com

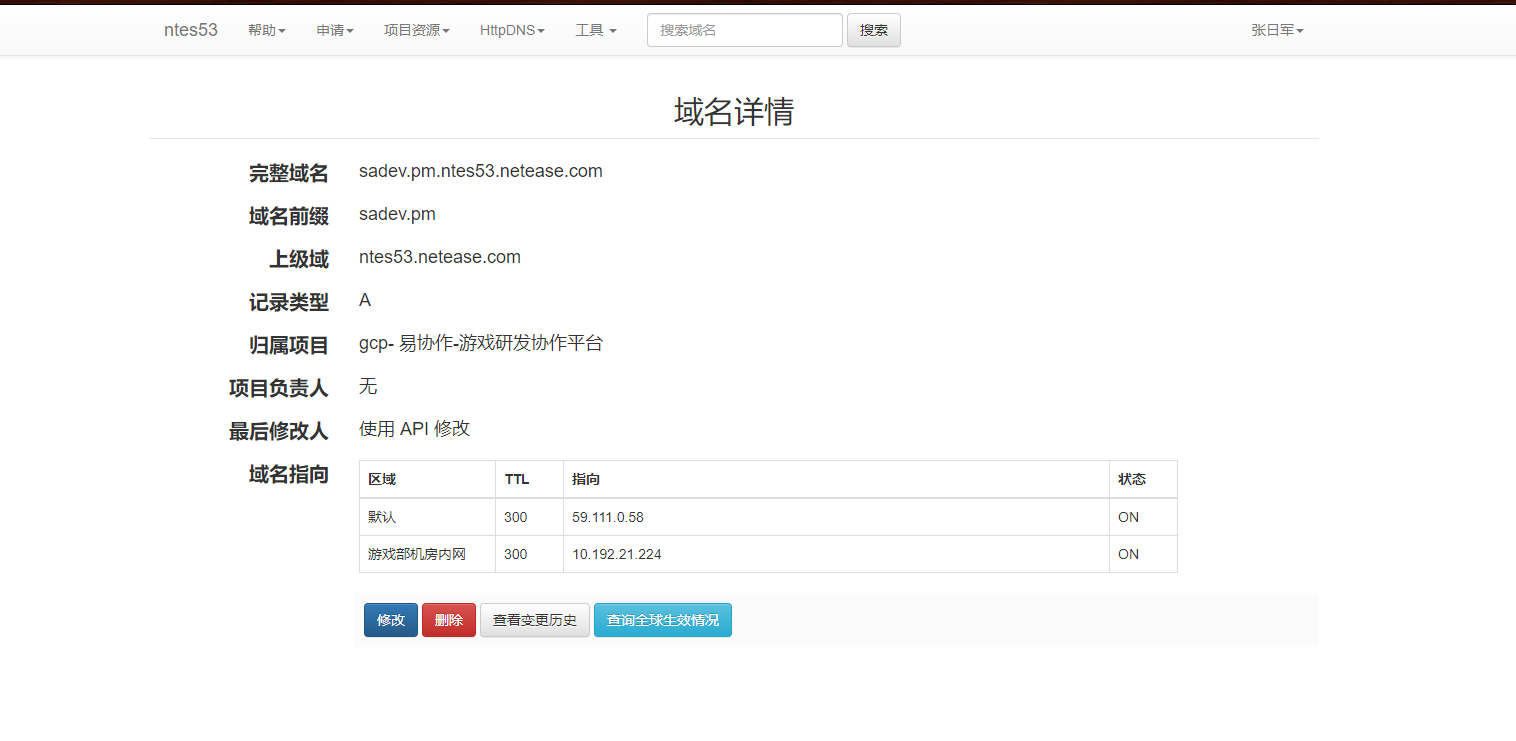




再查sadev.pm.ntes53.netease.com，可以看到其lbc入口，即可确认实例是落在哪个机器上面。

Lbc分内外网两个入口gcp和internalgcp两个实例

<https://hz.lbc.nie.netease.com/_gcp/lbc/instances?_num=20&_page=1&virtualip.ip=>



实例管理由程序完成创建，项目有需求时通过程序完成。

在haproxy机器上的配置里找到<http://sadev.pm.netease.com/>相关配置项,每个实例对应一行配置，对应数据库的的信息，nginx配置passager管理；

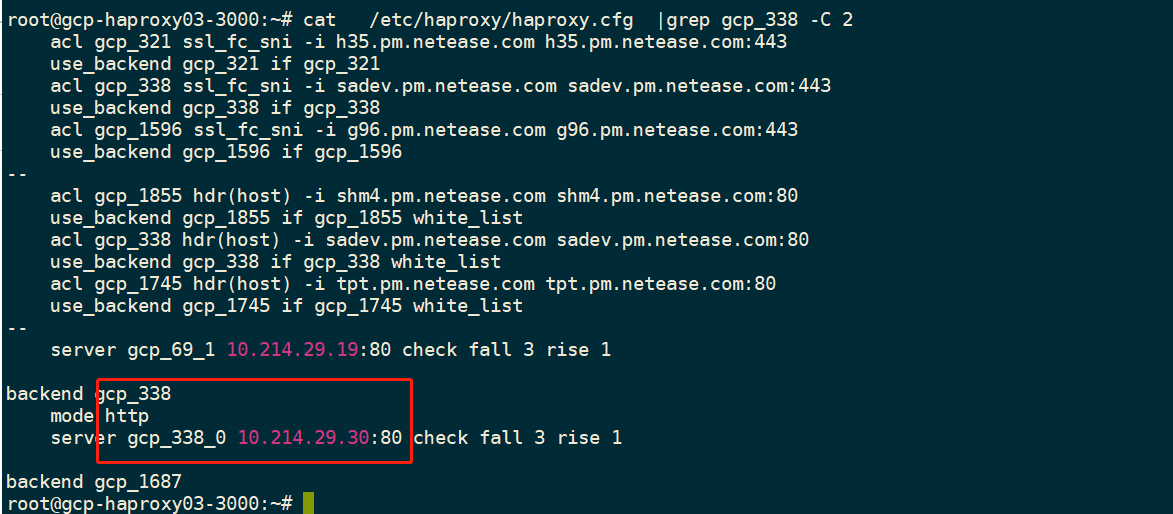
root@gcp-haproxy03-3000:~# cat /etc/haproxy/haproxy.cfg |grep sadev

acl gcp\_338 ssl\_fc\_sni -i sadev.pm.netease.com sadev.pm.netease.com:443

acl gcp\_338 hdr(host) -i sadev.pm.netease.com sadev.pm.netease.com:80

其中**gcp\_338**记录了域名后端信息

root@gcp-haproxy03-3000:~# cat /etc/haproxy/haproxy.cfg |grep gcp\_338 -C 2



10.214.29.30对应机器gcp-hyzn04-4000，群组4000-GCP外网正式服务器---互娱职能

正式服务器部署了passenger，通过passenger管理业务进程。

netstat -ltnp

cat /etc/nginx/sites-enabled/sadev\_338.conf

dpkg -l|grep passenger

dpkg -l|grep nginx

ps -ef|grep ruby

ps -ef|grep sadev

passenger-status

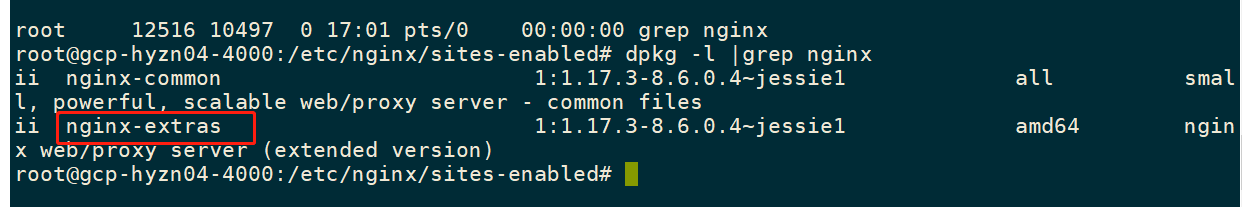
cd /etc/nginx/sites-enabled/

**群组4000-GCP外网正式服务器**

passenger通过nginx-extra包安装拓展，

**查看监听端口netstat -ltnp**

**dpkg -l |grep nginx，机器上会有**nginx-extras拓展版nginx



passenger-status查看运行情况

passenger管理，可以单独杀掉，不要通过重启或者reload nginx来重启passager，这样会导致机器上所有的实例受影响，

通过 touch tmp/restart.txt 来标记passenger需要重启实例 (此操作是实例级别的)

例如sadev\_338这个实例，需要进入到/home/gcp/running/sadev\_338实例目录，创建tmp/restart.txt空文件。

或者passenger-config restart-app

root@gcp-hyzn04-4000:/home/gcp/running/sadev\_338# passenger-config restart-app --help

Usage 1: passenger-config restart-app <APP PATH PREFIX> [OPTIONS] ##指定实例目录

Usage 2: passenger-config restart-app . [OPTIONS] ##重启当前目录下所有实例

Usage 3: passenger-config restart-app --name <APP GROUP NAME> [OPTIONS] ##通过实例名筛选重启实例

**passenger-status查看sadev\_338，有多个进程，挂掉其中一两个不影响**

/home/gcp/running/sadev\_338 (production):

App root: /home/gcp/running/sadev\_338

Requests in queue: 0

\* PID: 29132 Sessions: 0 Processed: 902 Uptime: 2d 0h 32m 39s

CPU: 0% Memory : 317M Last used: 34s a

\* PID: 4335 Sessions: 0 Processed: 62 Uptime: 1d 0h 12m 37s

CPU: 0% Memory : 232M Last used: 3m 12

\* PID: 1586 Sessions: 0 Processed: 12 Uptime: 21h 19m 42s

CPU: 0% Memory : 208M Last used: 9m 58s



cat /etc/nginx/sites-enabled/sadev\_338.conf

passenger\_min\_instances 3; 最小进程数

client\_max\_body\_size 500m; 用户可上传文件大小，

cat /etc/nginx/passenger.conf

passenger\_max\_instances\_per\_app 10;

passenger\_max\_pool\_size 120; 最大进程数，过大可能会导致内存占用过多

passenger\_pool\_idle\_time 90;

SRE|冼镇东 18:09:46

你可以需要了解下passenger，这个东西你跑nginx reload业务是会中断的

**<https://g.126.fm/00cOylO这个是部署文档>**

整个访问路径：用户----lbc----两台haproxy----后端群组4000的具体机器上nginx--passenger（也就是实例服务器）

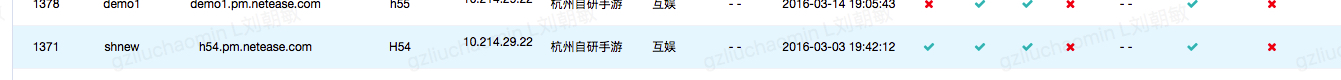
**gcp实例异常排查思路**

/home/gcp/var/log/nginx里面查对应实例nginx错误日志；

/home/gcp/var/log/gcp 里面查实例业务日志，

例如/home/gcp/var/log/gcp/gcp\_338/local/production.log

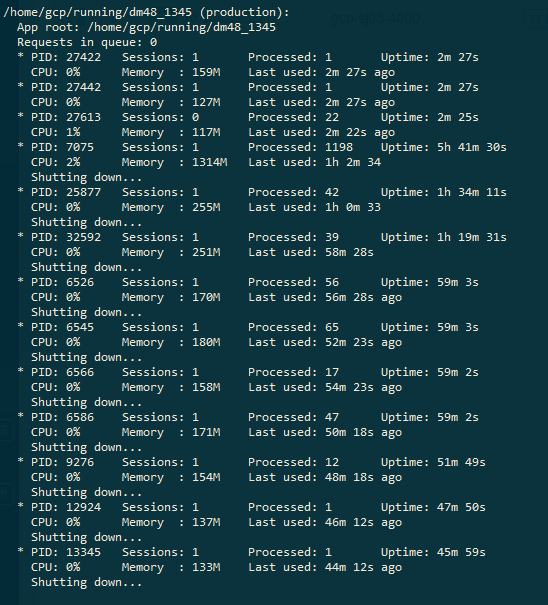
**h48.pm.netease.com页面显示this website is under heavy load**



炒面(刘朝敏) 17:08:01

同机器的实例，是可以访问的

冼镇东 | SRE -- 2020-11-06 17:11:53

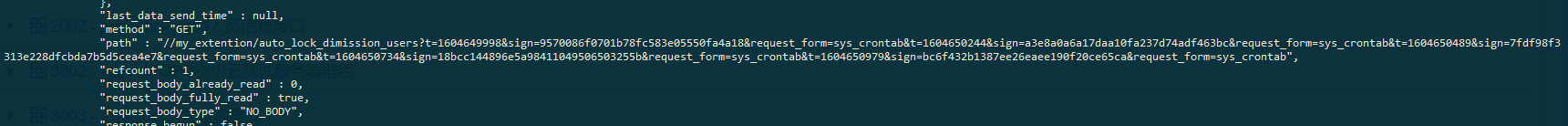


现在是他自动起的

冼镇东 | SRE 2020-11-06 17:12:04

我touch一下restart.txt吧

冼镇东 | SRE 2020-11-06 17:14:51



冼镇东 | SRE 2020-11-06 17:15:06

//my\_extention/auto\_lock\_dimission\_users基本都是这个接口的

炒面(刘朝敏) 2020-11-06 17:15:19

我看看这个是什么接口

冼镇东 | SRE 2020-11-06 17:18:23

留了一份输出，我kill掉那些shutting down的进程吧

/home/xianzhendong/passenger-status.txt

炒面(刘朝敏) 2020-11-06 17:20:56

kill吧，定时刷新离职用户的请求

冼镇东 | SRE 2020-11-06 17:21:08

kill完了

冼镇东 | SRE -- 2020-11-06 17:22:04

应该是这些shutting down的进程一直关不掉，然后新请求进来不能创建新的instance导致的

##**对应异常进程，项目确认后可以kill pid关闭；如果需要保留现场，保存一份输出到文件。**

**sudo passenger-status --show=requests 查看请求**

tail -f /home/gcp/var/log/nginx/u3\_1683/access.log ##查看连接日志

<https://www.phusionpassenger.com/library/config/nginx/reference/>

Passenger + Nginx配置

**gcp开发环境dumbo-office和jetis-office实例**

**mysql数据库4核16g 100g**

gcp-gcptest-130124-m.of1.dumbo.nie.netease.com



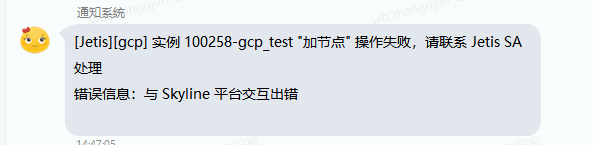
<https://dumbo-office.nie.netease.com/_gcp/instances/list?_num=20&_page=1&use=test>

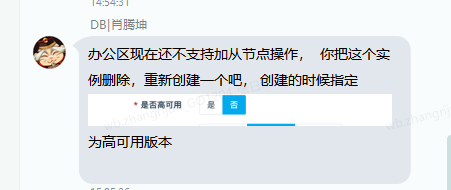
**Jetis配置申请参考线上实例，使用集成 Sentinel类型，最小内存版**

<https://jetis-office.nie.netease.com/_gcp/jetis/instances?_num=20&_page=1>



创建完成后发现没有从节点，手动添加从节点报错





**需要删除重建，指定高可用**

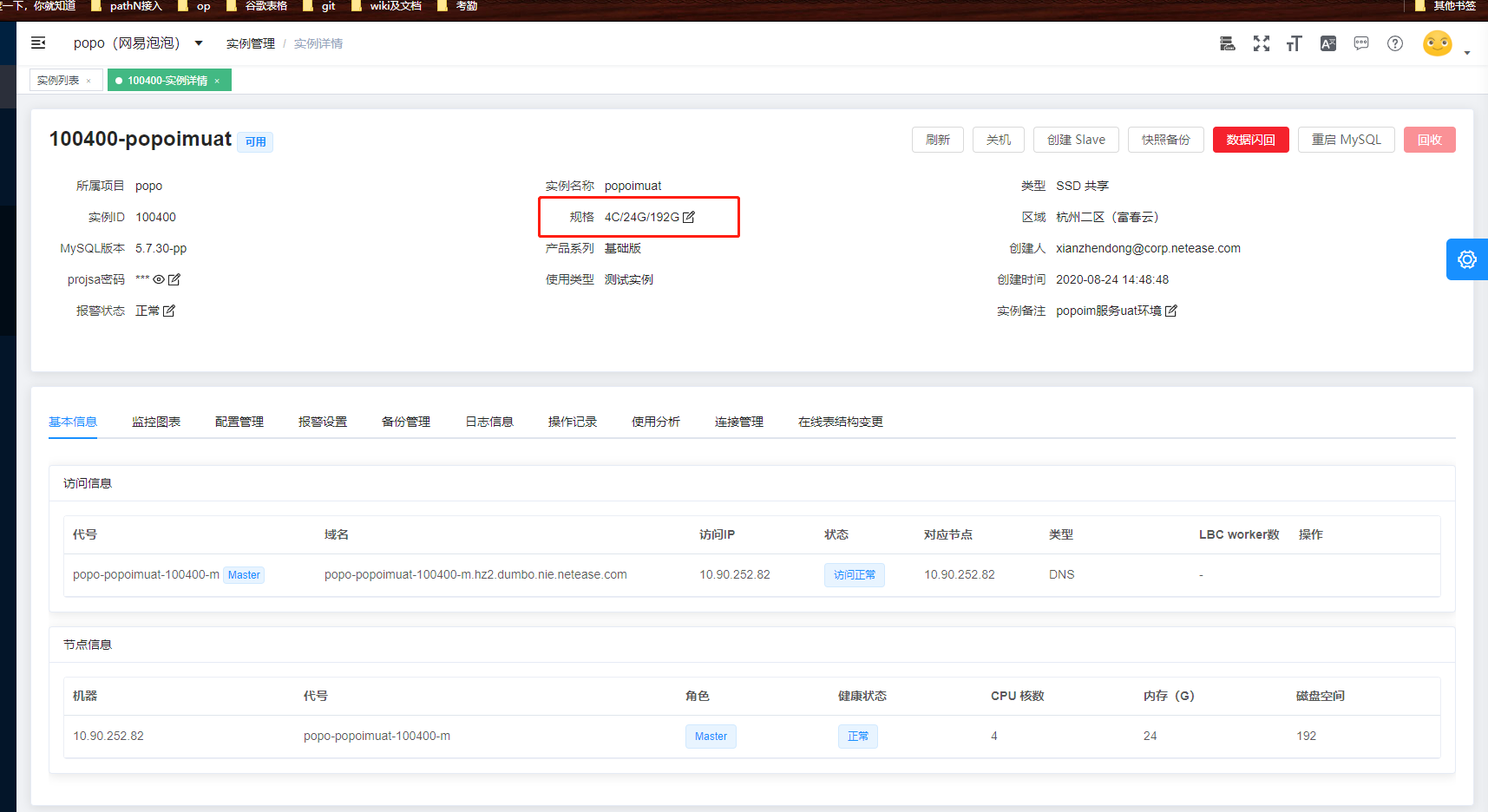


**popo的杜鹃座申请加一台节点**

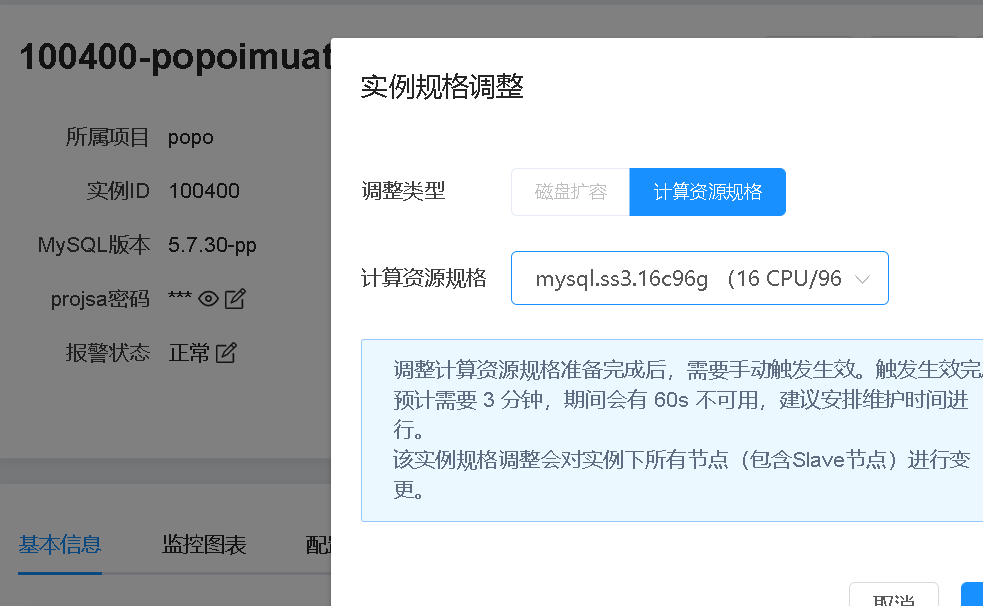
Workflow申请资源，不是直接sym2操作

**popo dombo实例调整磁盘**

<https://db.nie.netease.com/mysql/_popo/instances/detail/100400>(旧的dumbo入口没有这个规格调整按钮)



SRE|冼镇东 11:25:01



uat的dumbo数据库磁盘容量感觉不够

SRE|冼镇东 11:25:36

ssd共享实例的磁盘容量是和cpu内存绑定的

SRE|冼镇东 11:25:54

要扩磁盘就要cpu内存也一起扩

SRE|冼镇东 11:26:15

不然的话考虑下小小云？

**gcp宝瓶座的日志接入elk，找下loghub帮忙搞下**

<https://sym2.nie.netease.com/#/symphony/services/detail/gcp/gcp-api-gateway/proxy@hz-dg-sym_v2pre>



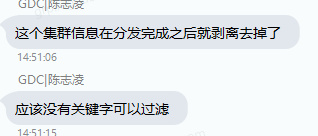
这个麻烦接入下elk

SRE|冼镇东 14:27:21

应该是不影响的，只需要跟loghub那边说一下就行

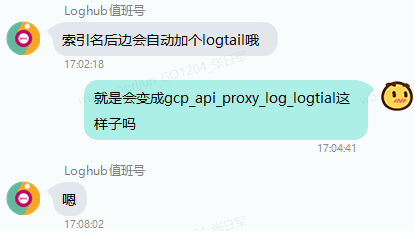
SRE|冼镇东 14:27:29

我们不用做什么操作的

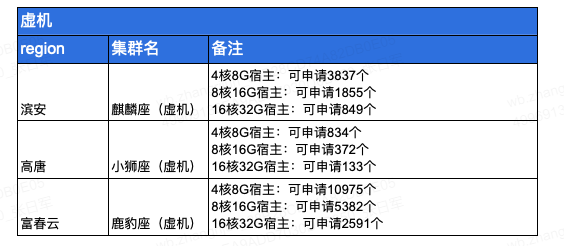


##需要找条日志去查询验证，无法查到相关宝瓶座等信息

gcp\_logtail更改名称gcp\_api\_proxy\_log







**Popo的sym2资源一部分在杜鹃，一部分在豺狼**

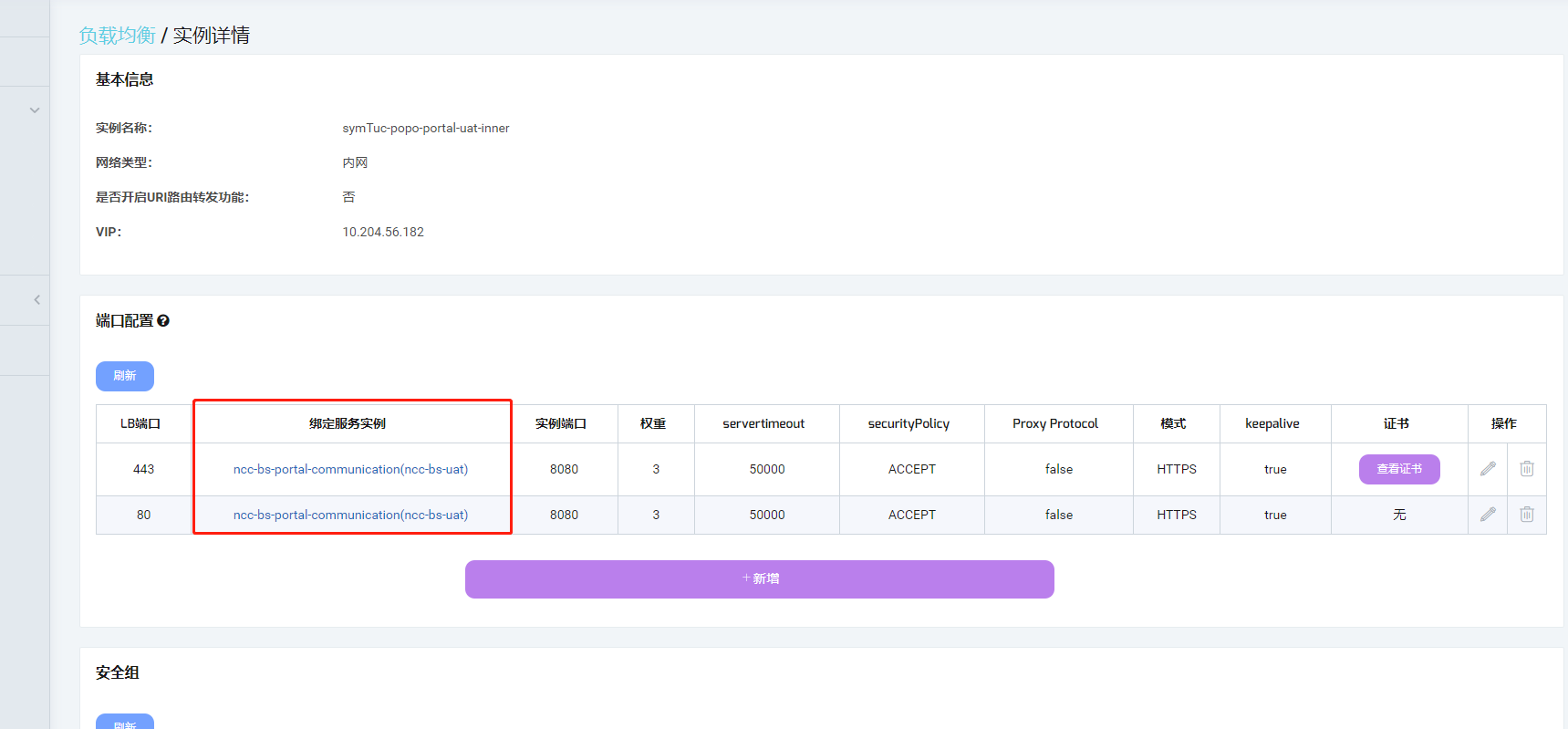
communication.uat-portal.netease.com

uat-portal.netease.com

麻烦@SRE|张日军 给这两个域名对应的服务搞一个新的内网lbc，并且创建对应的内网域名，都是sym上的lbc

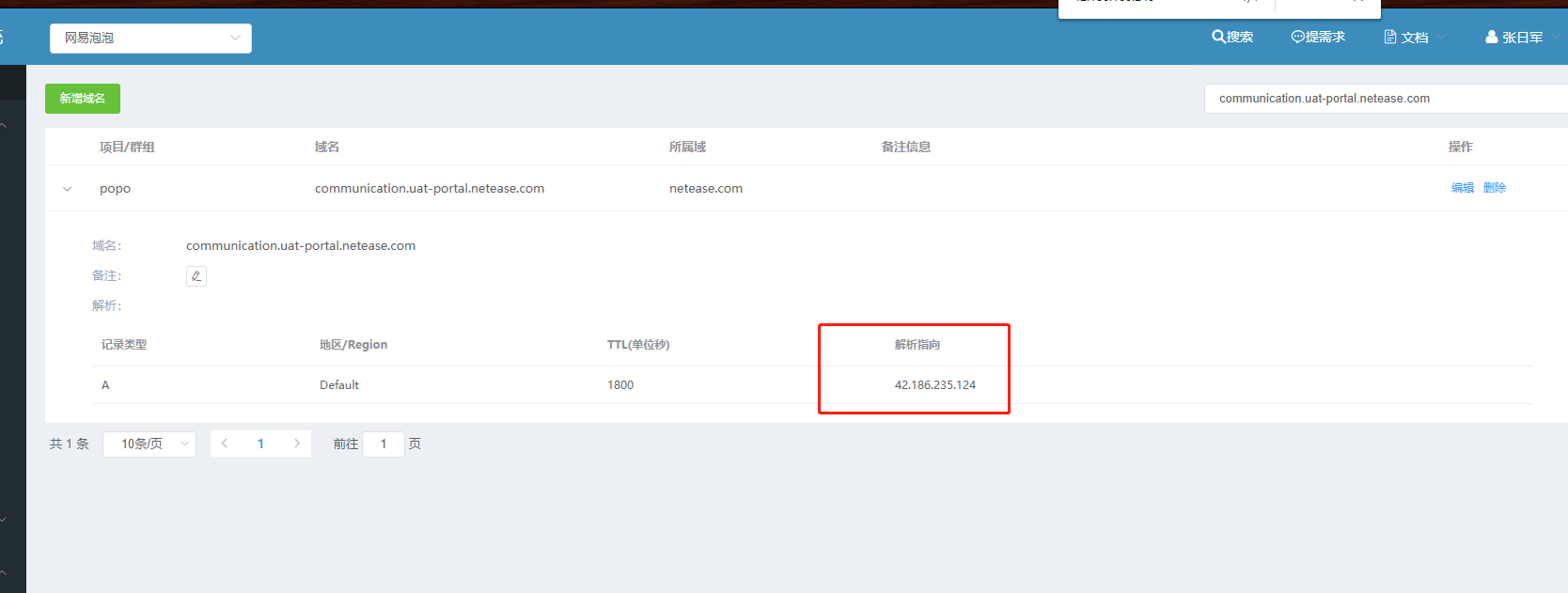
新建实例symTuc-popo-portal-uat-inner

[80端口跳转443配置需要找sym值班grp.dockersa@corp.netease.com](mailto:80端口跳转443配置需要找sym值班grp.dockersa@corp.netease.com)

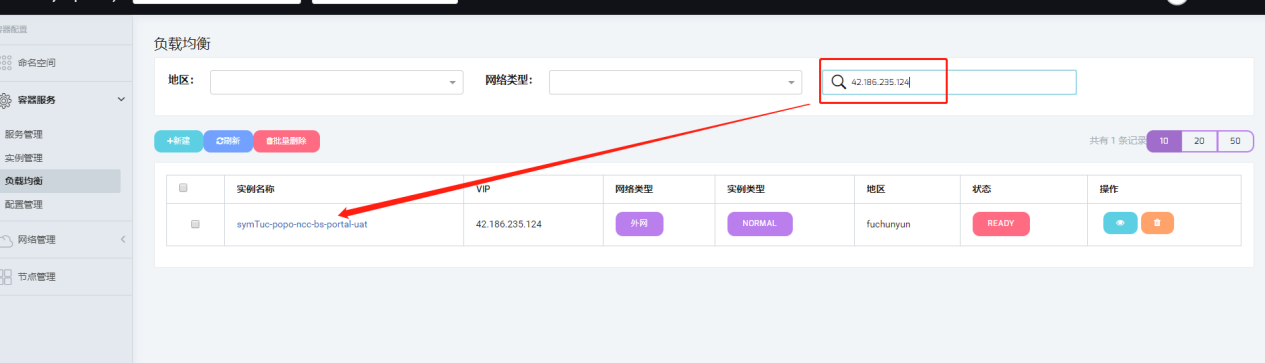


新建负载均衡实例----绑定的服务参照外网的实例---根据域名在Jupiter上查指向的lbc vip，就可以确定对应哪个负载均衡实例。

比如communication.uat-portal.netease.com指向42.186.235.124



在sym2上负载均衡里可以通过ip搜索对应实例



之后在<https://ntes53.netease.com/domain_name/addrecord/>申请新域名，指向新建负载均衡对应的vip



internal-communication.uat-portal.netease.com

internal-uat-portal.netease.com

两个内网域名之前指向杜鹃座lbc，已下线杜鹃座实例，修改新增指向财狼座

sym70-popo-popo-portal-uat-inner--->10.204.56.211

sym70-popo-popo-home-uat-inner--->10.204.56.213

需要在财狼座新增lbc，参考之前绑定的服务，需要配置80跳转443

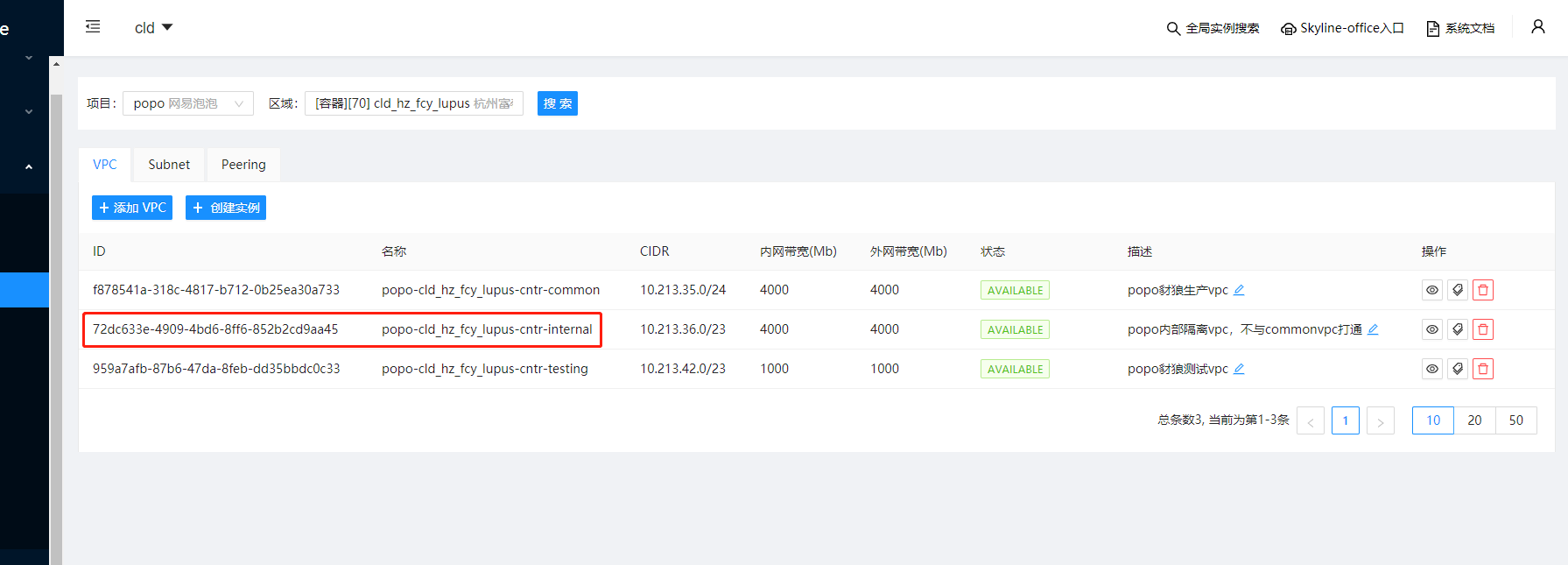
**popo项目杜鹃和豺狼internal vpc跟gcp富春云的vpc的peering**

跨项目集群接通peering需要workflow申请，找小小云值班处理

[http://workflow2.x.netease.com/#/approval?defid=kua\_ji\_qun\_peering\_shen\_qing:7:25930392&instid=28467224](http://workflow2.x.netease.com/" \l "/approval?defid=kua_ji_qun_peering_shen_qing:7:25930392&instid=28467224)



popo的财狼座和杜鹃座分别有名称为internal的vpc，只需接通internal的vpc到gcp的peering



**Dumbo部分旧实例，存在内存泄漏问题**

[gcp项目redmine的三个节点和redmine02的从库需要调整，找gzyuhaojian@corp.netease.com确认操作时间](mailto:gcp项目redmine的三个节点和redmine02的从库需要调整，找gzyuhaojian@corp.netease.com确认操作时间)----15号6点

冼镇东 11:02:46

因为是双主，所以只重启standby节点就行，只在切换的时候大概会有10秒不可用

冼镇东 11:03:18

从库重启则是会有1分钟不可用

大概的流程是这样

**重启从库->重启现在的备库->切换主备->重启新的备库**

redmine02----10.191.14.78(Slave)，重启从库

Redmine----10.90.248.54-Master(Primary)--切换后最后再重启

10.90.249.55--Master(Standby)--先重启从库再操作----备库的重启耗时较长

10.90.248.55--Slave--从库先重启；

**Popo压测机器申请**

@SRE|张日军 正式环境的配置大概要这样

link-app 40c128g

persist memCached 40c64g

mysql 40c256g2Tssd

没有64g的机器的话就直接上128g吧

东冠和富春云都是内外网万兆的，尽量在这两个机房，尽量申请机架服务器

<https://matrix.nie.netease.com/ant/spare/apply?global_project=popo>

ssd需要作为备件申请

42.186.140.186添加两个ssd盘，机房帮忙做了raid1

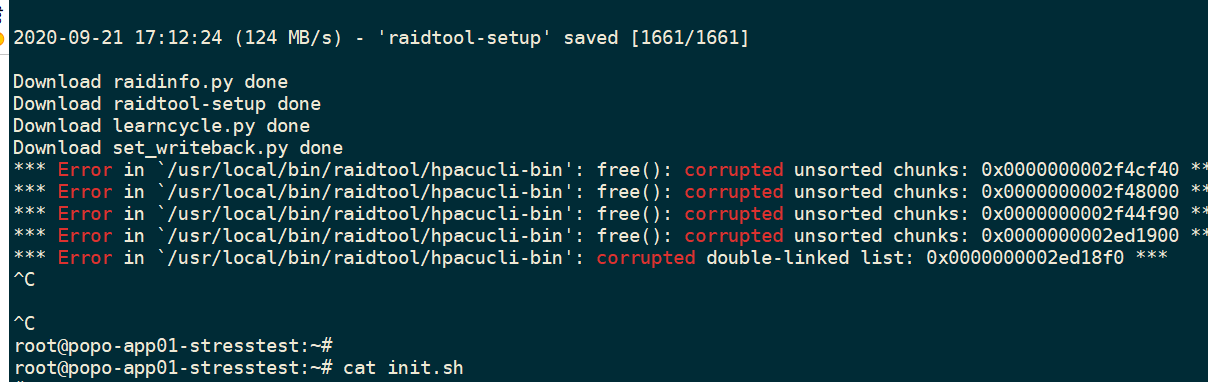
之后分区，格式化，挂载到/home/mysql目录

<https://g.126.fm/04zVBHq>

<https://www.cnblogs.com/wholj/p/10924129.html>

42.186.151.104机器初始化

https://g.126.fm/044E5Pi初始化脚本



igi(陈文钦) 17:25:32

看起来像是 raidtool这一套很老的软件不兼容新的硬件

igi(陈文钦) 17:26:04

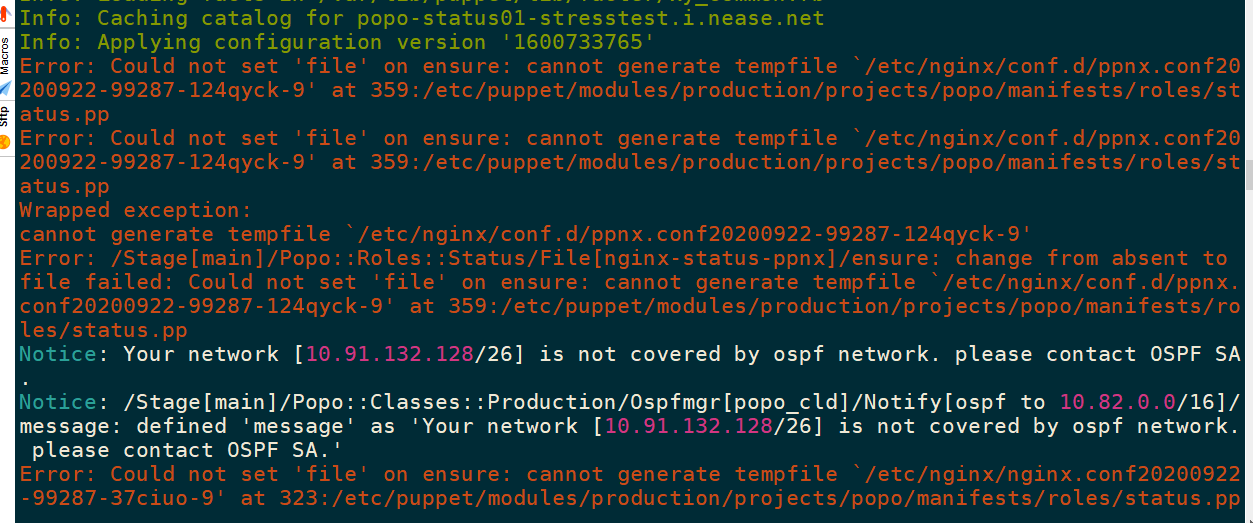
不过这一套raidtool 应该不需要使用了才是，你先跳过继续跑

通用初始化脚本使用--skip 50hwinfo.sh跳过；

平台初始化脚本init\_platform.sh 通过修改脚本里opts\_stored=' --skip 50hwinfo.sh '来跳过

42.186.151.93 puppet报错

为nginx未安装导致



Debian使用官方源安装nginx

添加/etc/apt/sources.list.d/nginx.list

deb http://nginx.org/packages/mainline/debian/ jessie nginx

deb-src http://nginx.org/packages/mainline/debian/ jessie nginx

# 导入key

wget http://nginx.org/keys/nginx\_signing.key

apt-key add nginx\_signing.key

# 更新列表并安装

apt-get update

apt-get install nginx

平台机器使用systemd管理，debian8跑初始化的时候得加一下-systemd的参数



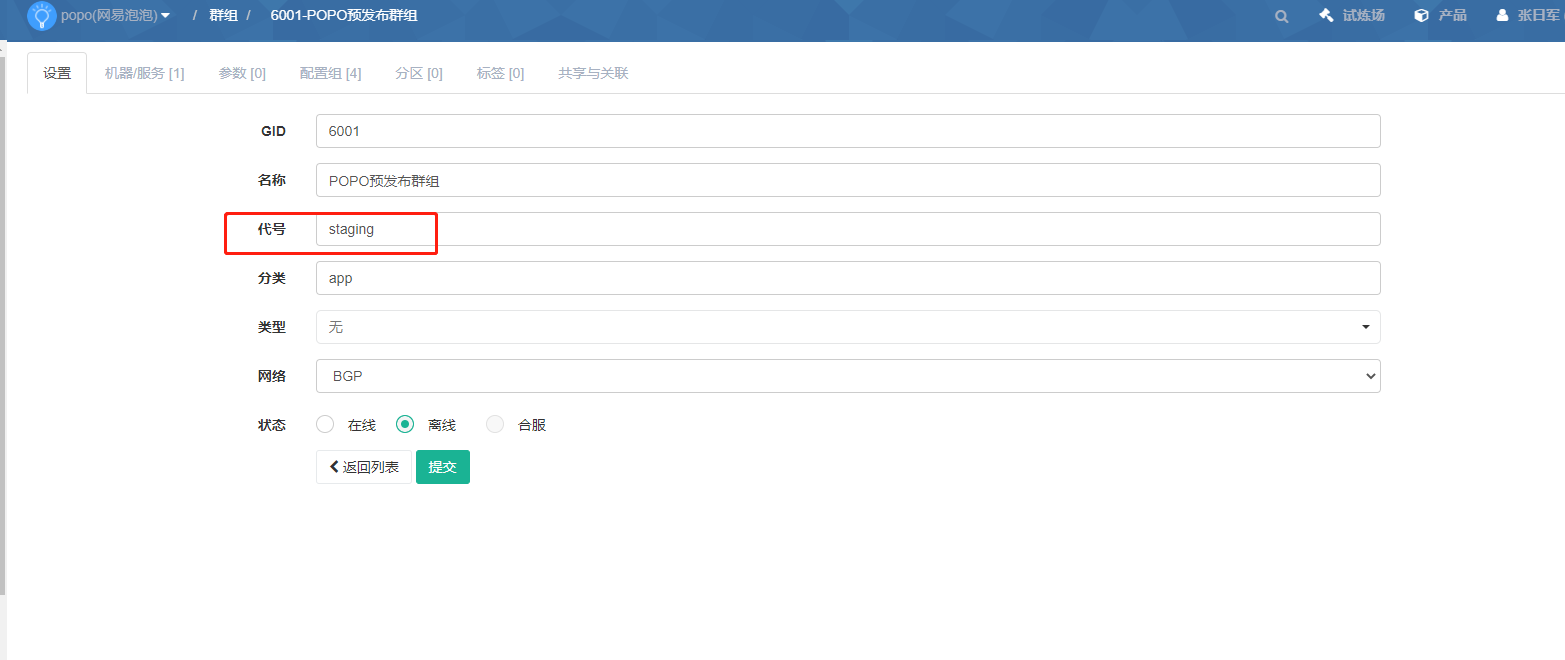


如果初始化没加-systemd参数，需要用apt-get remove sysvinit-core来修复；

不过是要重启机器的，所以得在交付前试试systemctl命令会不会报Failed to get D-Bus connection: Unknown error -1

wget -O init\_platform.sh http://webtech:ZVop4cVYiCmks@webtech.x.netease.com/init/init\_platform.sh && bash init\_platform.sh -systemd

Popo添加预发布群组6001



**群组代号不能复用app，不然会生成同名机器，无法检验提交；**

**高危操作，会导致原来app01机器42.186.33.95的jupiter上配置的端口防火墙被清空----万幸机器上防火墙没被触发刷新**

机器需要绑定三个服务app ,appserverhttp,link

用平台部的初始化脚本初始化重启，puppet正常后，管理机操作

**cbc 42.186.20.114 cbcsync cbrc**

**cbc 42.186.20.114 sync cbrc**

Popo小小云vdev机器10.212.27.135-popo-vdev08-vdev，初始化后按照docker

https://mirrors.tuna.tsinghua.edu.cn/help/docker-ce/

用平台初始化脚本初始化，不需管puppet，直接关闭应该也没问题

高唐小小云添加一台10.212.27.136，添加到3001 dev群组，同样的问题初始化重启后，32200不通，需要手动放通。Jupiter上管理也有问题，添加端口或白名单会报错。

**ncc-bs-portal-home服务需要再启2个实例**

云开(蔡晶龙) 10:35:50

@建鸿(陈建鸿) 我们uat环境在做压测，

现在ncc-bs-portal-home服务需要再启2个实例，再申请两个ip，现在还有资源吧？

source(王烁斯) 11:16:25

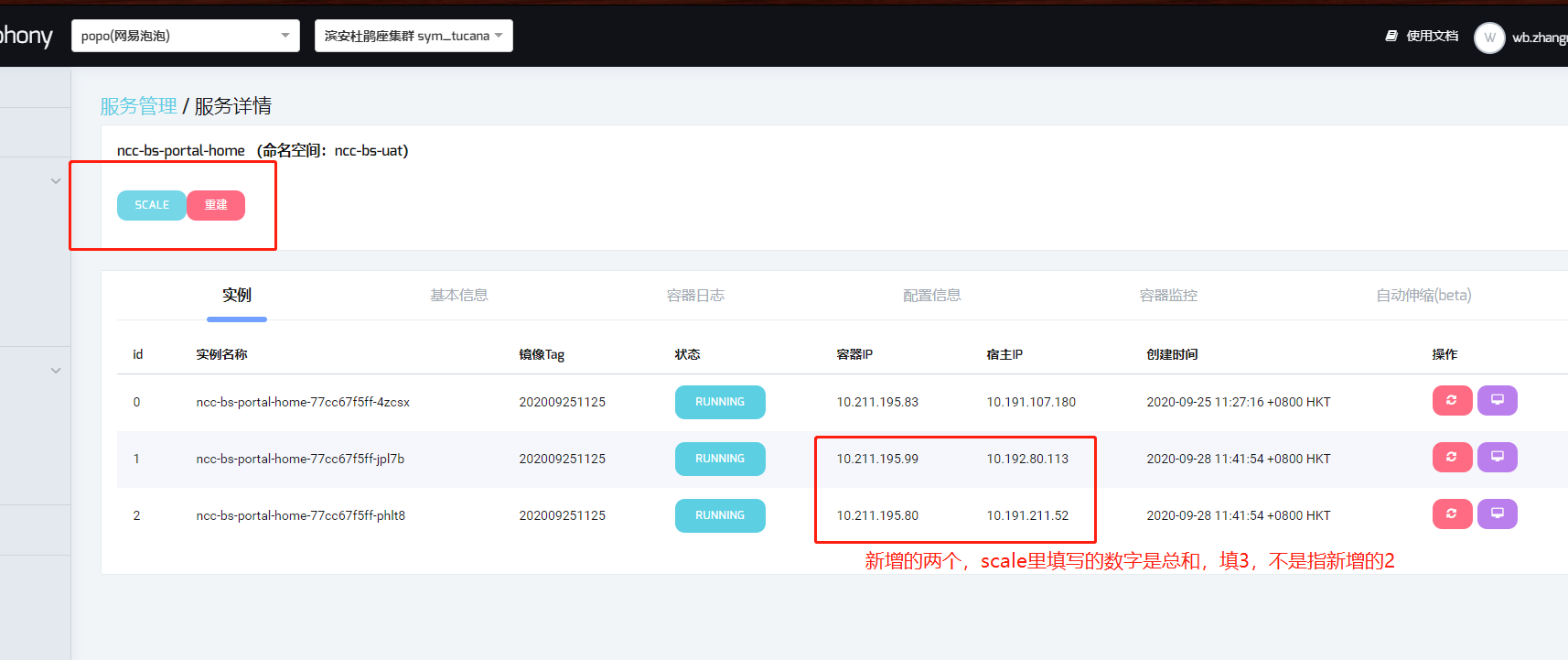
要先确认是否有用 fixed ip

source(王烁斯) 11:16:29

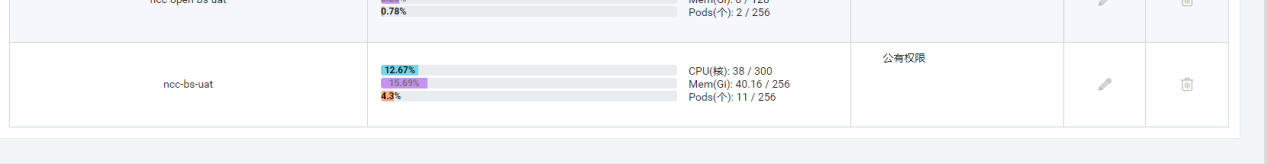
也就是容器有没固定 ip

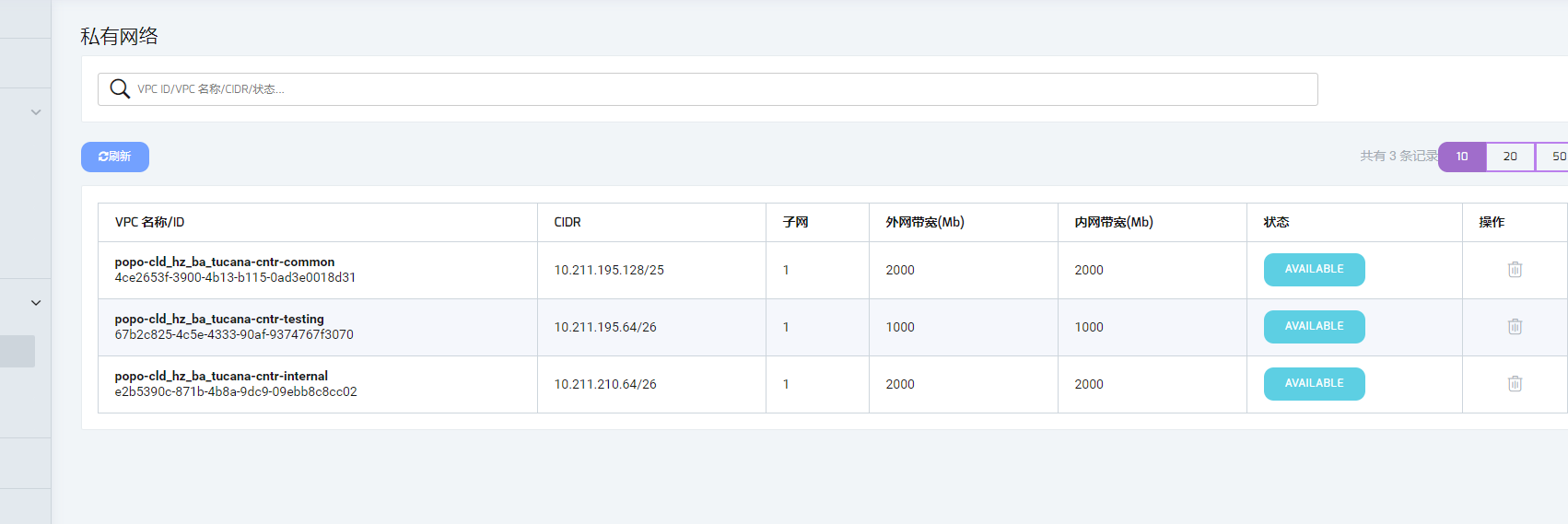
source(王烁斯) 11:16:44

没有的话，是可以直接 scale up 增加副本



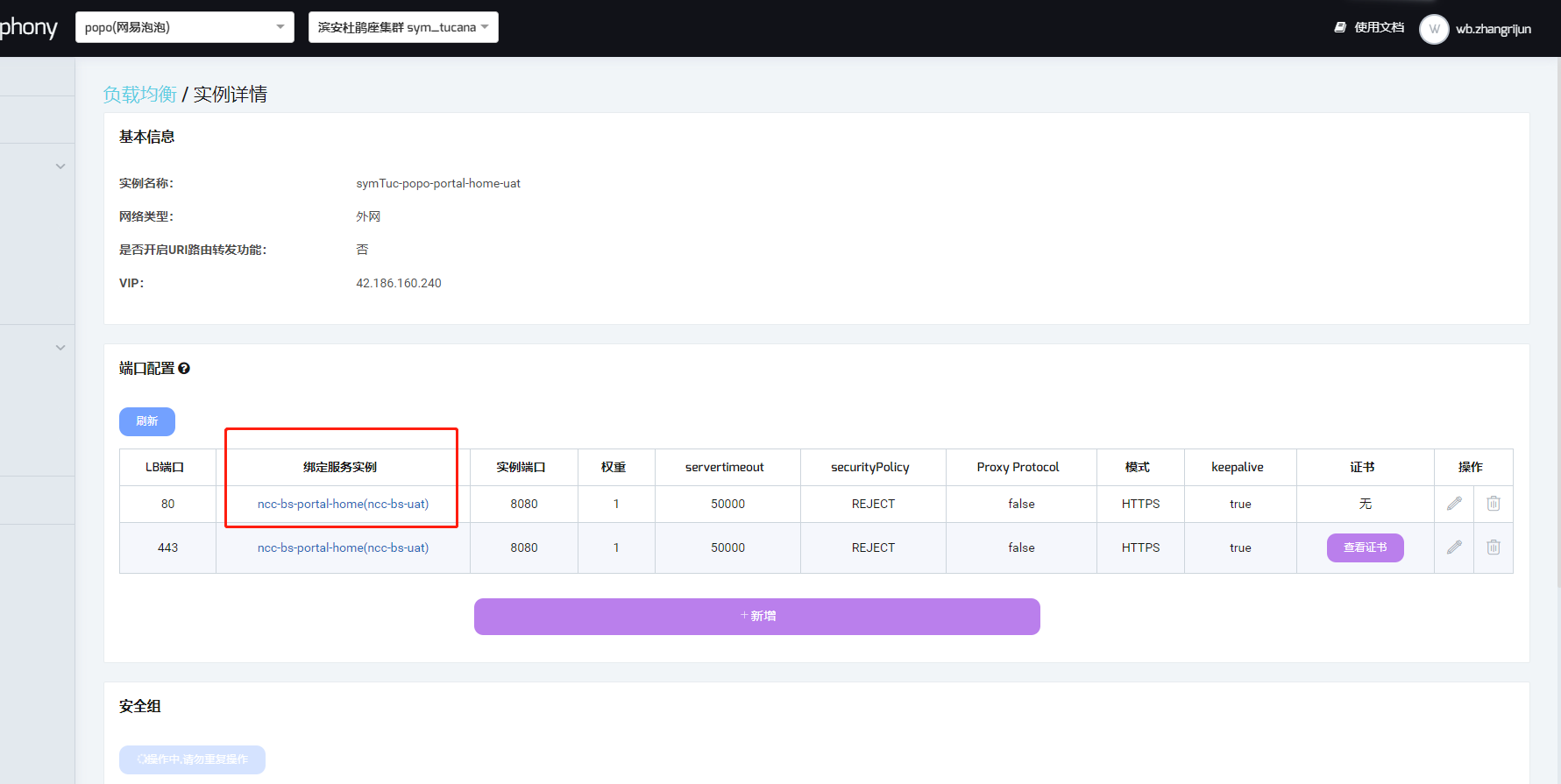
需要确认资源是否足够，ip资源是否足够，ip不够需要新申请vpc





10.211.195.64/26资源足够

服务通过sym2的lbc管理，扩容后新增实例会自动加入到rs里



source(王烁斯) 11:47:06

如果是用 sym 的负载均衡 提供服务，是不需要手动去调整 rs 了

source(王烁斯) 11:47:21

服务扩容后新容器 ip 会自动注册到 rs 列表

**2020-09-29popo压测机器加权限，分组管理的问题**

sh adduser.sh [xuwei01@corp.netease.com](mailto:xuwei01@corp.netease.com) ##app和db机器

sh gameusersadd.sh [xuwei01@corp.netease.com](mailto:xuwei01@corp.netease.com) ##status机器，注意脚本目录，添加到8号

42.186.151.93

42.186.151.104

42.186.140.186

##使用了阿拉丁流程https://aladdin.nie.netease.com/\_popo/aladdin/processes/26601/execute

根据群组来添加权限

**popo压测服 42.186.151.104 麻烦帮我调一下tomcat的堆大小**

热浪(韦明伦) | POPO丨服务端 15:52:49

@张日军 popo压测服 42.186.151.104 麻烦帮我调一下tomcat的堆大小 -Xmx4g -Xms4g -XX:NewRatio=1 -XX:PermSize=256m -XX:+UseConcMarkSweepGC

热浪(韦明伦) | POPO丨服务端 15:53:00

/home/tomcat8-appserverhttp

**如果使用了server.xml管理启动进程，需要修改galaxy配置模板**

冼镇东 16:01:26

https://galaxy.nie.netease.com/\_popo/galaxy/configtpls/39850/settings

冼镇东 16:01:46

貌似没绑定stress的配置组

冼镇东 16:02:01

可以绑定一下

冼镇东 16:07:05

https://galaxy.nie.netease.com/\_popo/galaxy/configtpls/35354/params

冼镇东 16:07:47

server.xml这个是有根据群组判断的逻辑的，你问问他们用的什么域名做连接的----这部分是在内容与参数里面定义，预览是看不到的



冼镇东 16:10:45

不太对劲，他们已经在用了的话就跟正式环境不是用同一种启动方式的

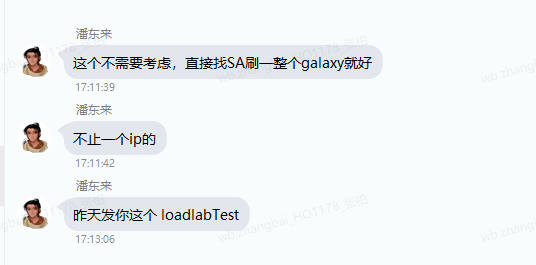
冼镇东 16:11:09

你得从头跟一下看看怎么加上参数了

**##由于压测服采用不同启动方式**

修改/home/tomcat8-appserverhttp/bin/setenv.sh即可

**42.186.175.235压测白名单配置**



42.186.175.235是sym2上的lbc，需要到sym2上操作添加安全组 loadlabTest，端口8009

--一开始以为需要对42.186.175.235机器8009配置特定的白名单，纠结如何绑定到服务；

其实42.186.175.235是lbc，通过安全组去配置即可，sym2的负载均衡需要在sym2上操作，不能在lbc上直接修改； 而且loadlabTest这个是压测环境的白名单，已经存在，不需要额外操作。

**sidekiq进程数报警配置**

冼镇东 12:38:49

[美术制作协作平台][sidekiq进程数小于1][2020-10-12 12:35:38]

机器:

10.214.8.12|acp-apc201-2001|APC2对内正式服务器

冼镇东 12:39:34

日军，这个报警有个需求，就是在sidekiq挂掉的时候跑一下aladdin流程重启一下sidekiq

冼镇东 12:39:41

[阿拉丁2][acp]

详情：流程已完成

描述：无

提交者：xianzhendong-冼镇东

流程：[服务启停][sidekiq] APC重启sidekiq 服务

链接：https://g.126.fm/03BmNj6

冼镇东 12:40:07

黑洞应该有报警触发aladdin流程的功能的，看看怎么做一下？

看一下绑有sidekiq监控这个服务的机器，有些机器有多个sidekiq实例，需要的重启aladdin步骤是不一样的



冼镇东 12:52:19

对了，另外现在都是sidekiq进程数量小于1才报警，对于一些多个实例的机器报警可能会有缺失，4个挂了1个这样，顺便帮忙完善下这些多实例机器报警吧

冼镇东 12:52:57

<https://monitor.nie.netease.com/_acp/blackhole/alarmcategory>

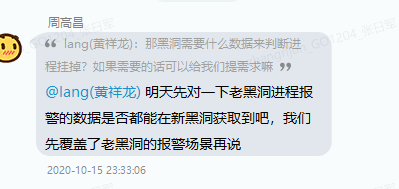
Galaxy搜索服务sidekiq，筛选机器

10.214.8.17机器上有四个sidekiq进程 ##后续看又是只有一个实例了

10.82.6.14上有6个进程

10.214.8.12机器上一个sidekiq进程

**##提单给黑洞，支持process指标的聚合**



**[10.214.8.3](https://galaxy.nie.netease.com/_acp/galaxy/machines/259209/hw" \t "https://galaxy.nie.netease.com/_acp/galaxy/)和10.214.8.12，实例为一个，通过指标报警，<1报警或无数据报警---生效的是无数据报警，不会存在0的情况；**

10.82.6.14七个和10.214.8.17两个，组合报警（指标无数据加变化，无数据针对两台机器，变化只针对多实例的这台机器）##这两台机器上实例数会有变化

无数据报警会受到gpostd采集上传的影响，会出现进程正常，gpostd数据采集上报延迟，这个指标值消失，也会触发报警

**Nginx修改配置client\_max\_body\_size 200m;**

<https://blog.csdn.net/z69183787/article/details/83070275>

冼镇东 | SRE 12:07:52

https://galaxy.nie.netease.com/\_popo/galaxy/configtpls/41398/params

冼镇东 | SRE 12:08:12

这个是测试服001-vdev-tomcat-proxy.conf的

你把admin的那个也弄到galaxy上吧，省得又发生这种问题

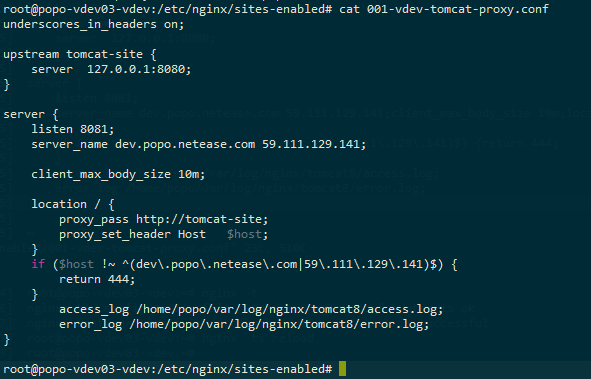
**暂时是手动去修改sites-enabled目录下的配置，之后nginx -s reload生效**

配置组----popovdev\_config----新建admin的配置模板，绑定到这个配置组

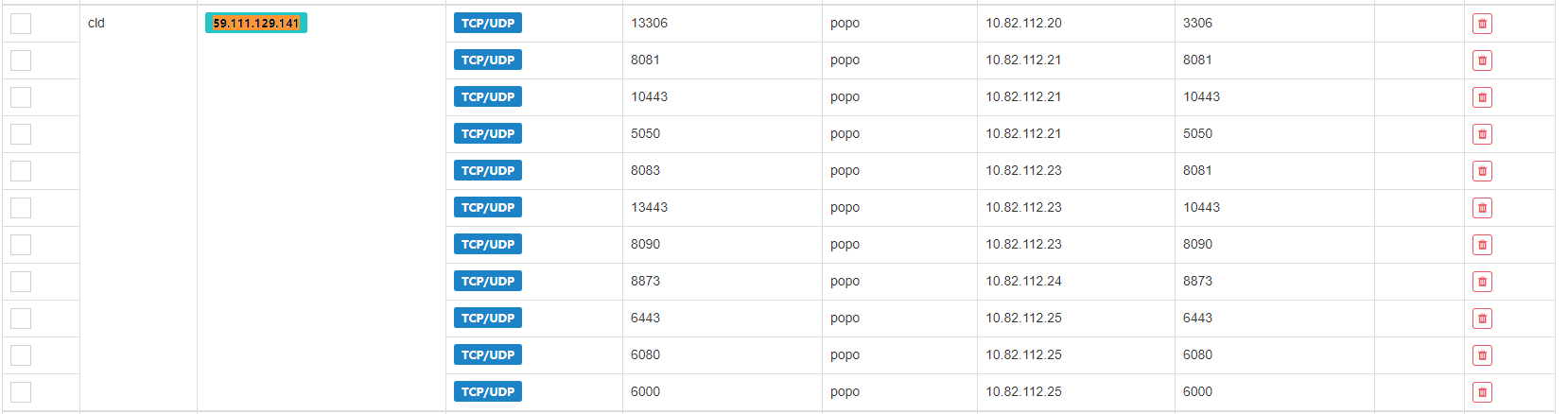
热浪(韦明伦) | POPO丨服务端 -- 2020-10-13 11:40:54

popoadmin是 8083 admin-v2是8090

popoadmin的java进程是8080端口吧



这个是对应的8083端口的nginx配置



59.111.129.141:8083映射到10.82.112.23:8081(所以机器上监听的是8081端口)

上述测试环境里的admin-v2跟下面的admin2popo应该不是一个东西，配置不同。

galaxy上服务admin2popo，暂时绑定到机器为10.214.0.196，属于20003-POPO周边HTTP接口服务群组，其nginx配置在puppet上

<https://git-sa.nie.netease.com/puppet/popo/blob/master/manifests/roles/admin2popo.pp>

主配置路径

<https://git-sa.nie.netease.com/puppet/popo/blob/master/files/conf/nginx/admin2popo/nginx.conf>

这个服务对应的配置模板为

<https://galaxy.nie.netease.com/_popo/galaxy/configtpls/58282/params>

client\_max\_body\_size 200m;这个应该在这里加即可

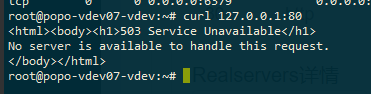
**办公区访问10.214.0.204的80端口**

10.214.0.204为测试环境文档服务机器，需要开放给sym2-hz的容器访问；通过创建外网lbc配置白名单实现----sym2-hz属于办公网，开放office白名单即可

vip 59.111.0.110

程序测试curl 59.111.0.110报错503，怀疑lbc有问题；

直接到机器上curl localhost:80同样是503，lbc应该没问题，可能服务部署问题



**高唐mysql实例saastest调小规格**

操作参考https://g.126.fm/035KNny

触发了内存报警

[处理][dumbo->gcp][内存使用率过高][2020-10-14 19:53:50]

[异常][gcp\_saastest][dumbo-mysql01-130063][10.212.10.42]内存使用率为93.6238%

如需协助，请@曹鹏(17600209442)

处理指引: https://g.126.fm/006H9iU

详情: https://g.126.fm/02oOZA8

张日军 正在处理,预计2小时处理完成。

因为innodb\_buffer\_pool\_size参数没有调整，调小后恢复



**测试环境s3申请**

测试环境申请前端：https://test-clds3-skyline.nie.netease.com/gajira/bucket

需要SA权限才能申请，如果没有权限，请联系项目SA协助申请

需要先创建用户，流程中心先申请quota

不是https://skyline-office.nie.netease.com/

popo负责人：[gzliujingyu@corp.netease.com刘劲宇](mailto:gzliujingyu@corp.netease.com刘劲宇)

[zhulianxin@corp.netease.com祝联新](mailto:zhulianxin@corp.netease.com祝联新)

[weiminglun@corp.netease.com韦明伦](mailto:weiminglun@corp.netease.com韦明伦)

Gameacademy：[zhulianxin@corp.netease.com祝联新](mailto:zhulianxin@corp.netease.com祝联新)

[lvzeyi@corp.netease.com吕泽义](mailto:lvzeyi@corp.netease.com吕泽义)

gcp: [gzyuhaojian@corp.netease.com余浩坚](mailto:gzyuhaojian@corp.netease.com余浩坚)

[gzliuchaomin@corp.netease.com刘朝敏](mailto:gzliuchaomin@corp.netease.com刘朝敏)

[操作类指引wb.zhouzhaoheng@mesg.corp.netease.com](mailto:操作类指引wb.zhouzhaoheng@mesg.corp.netease.com)

Acp: [gzfanzhifeng@corp.netease.com范智锋](mailto:gzfanzhifeng@corp.netease.com范智锋)

**gcp初始化部署新实例流程报错**

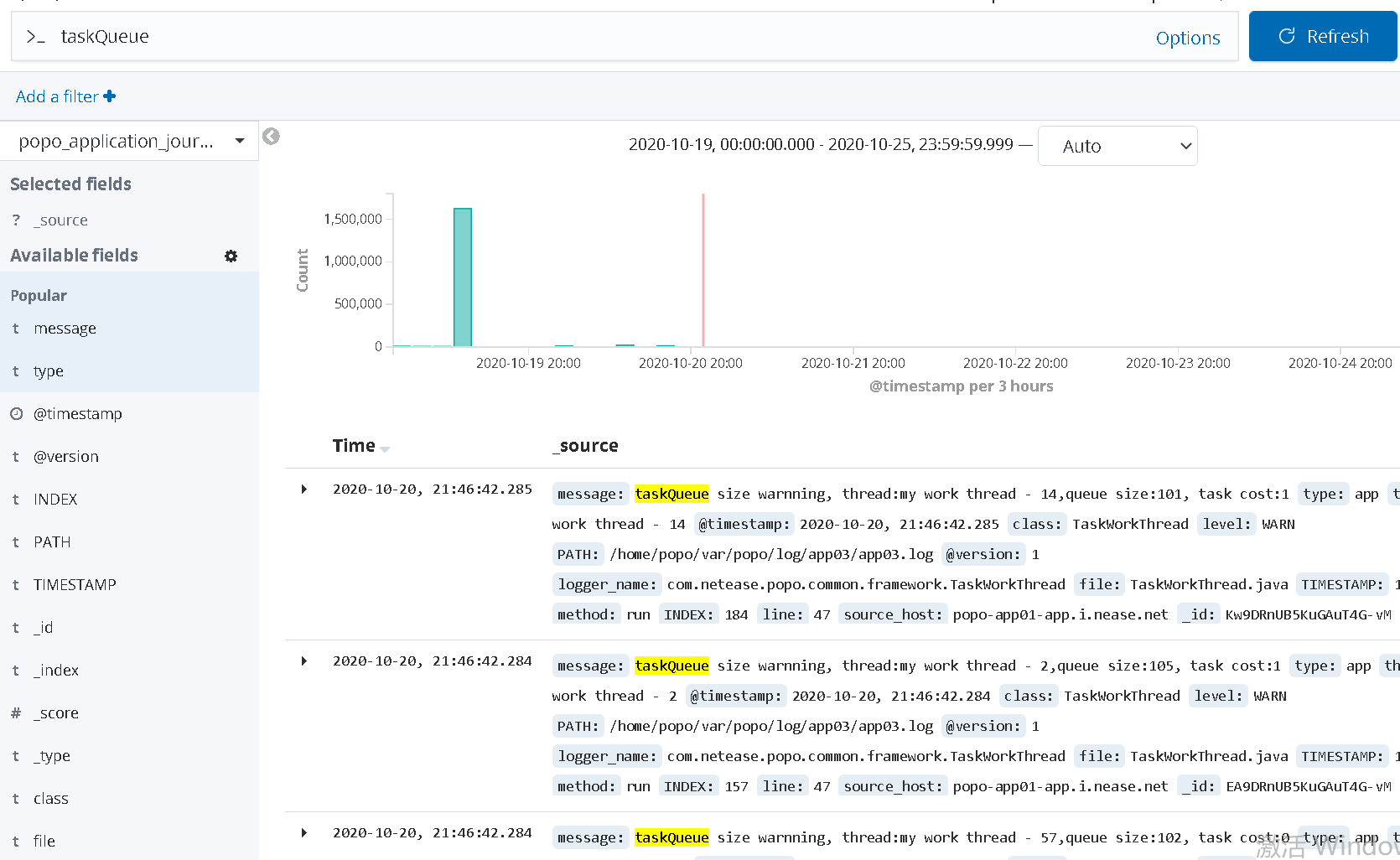


检查init参数



其实是选择错了机器导致流程报错。

**popo elk** popo\_application\_journal索引**日志配置taskQueue消息告警**



这个日志出现的时候就是任务有堆积了

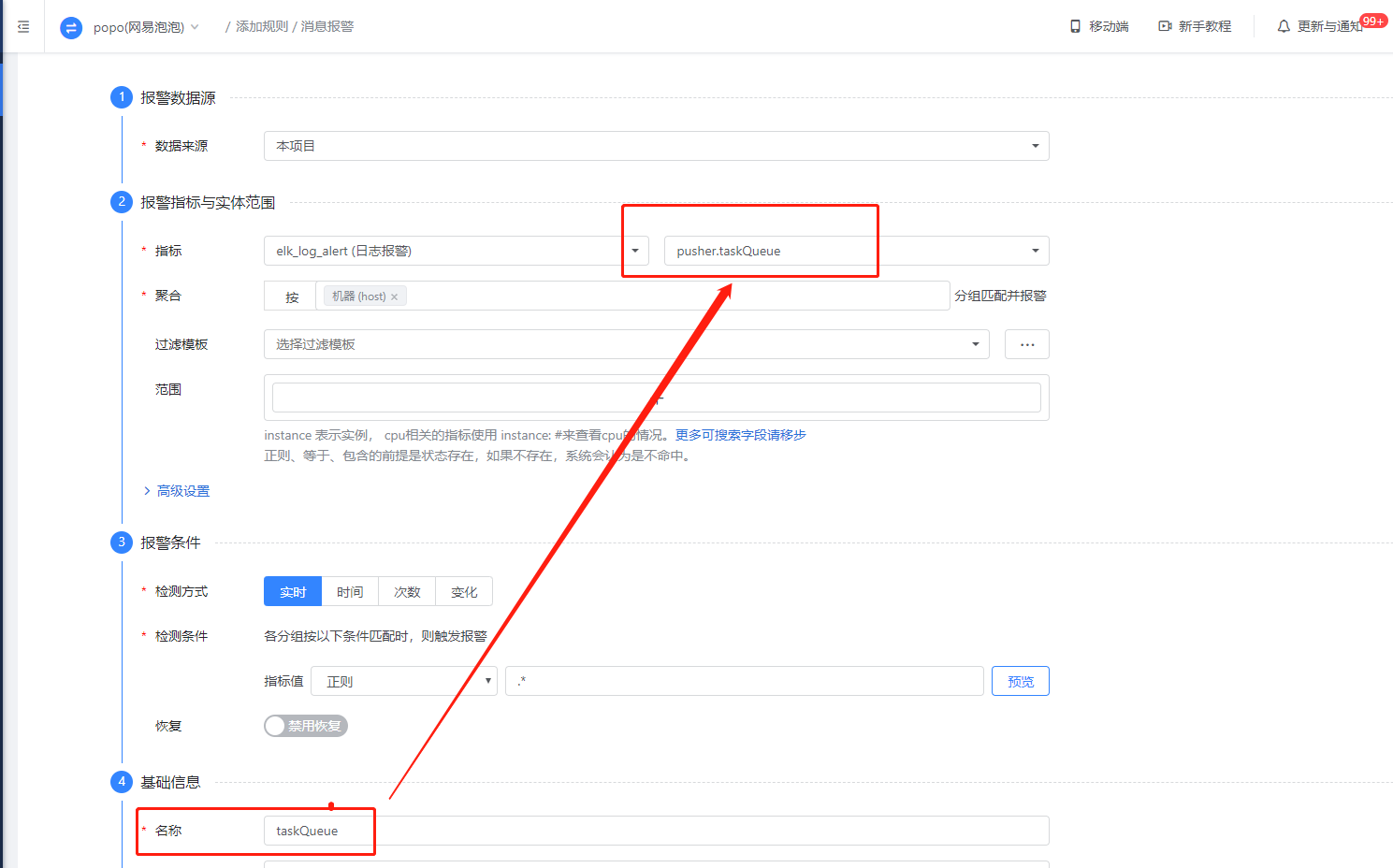
一平(刘劲宇) | 效率工程部 2020-10-20 21:51:38

这个可以elk里面加个报警

<http://elk.x.netease.com/ntes/alert/2654>

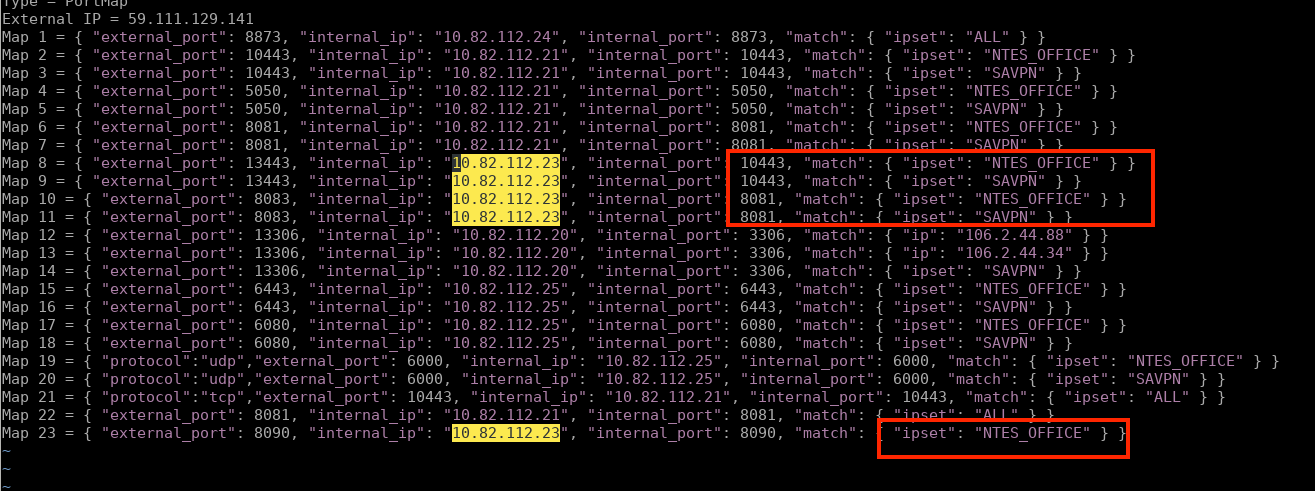
Elk新建告警之前，需要先保存过滤条件，通过新建filter，点击save保存过滤条件。一开始过滤条件命名为[message:taskQueue](http://elk.x.netease.com/app/kibana" \l "/discover/44533620-1344-11eb-b809-77369b0e7ef6)，保存告警一直报错，无法调试，修改为[message\_taskQueue](http://elk.x.netease.com/app/kibana" \l "/discover/44533620-1344-11eb-b809-77369b0e7ef6)后正常。

type的配置需要对应新黑洞的通知规则，先在elk上填写type名称（也就是黑洞规则的名称），直接点击跳转到新黑洞规则编辑界面，才能保证指标值能正常显示



**59.111.129.141 8083端口有白名单吗，杜鹃连不上**

Ip为小小云机器，找值班



需要重新申请wf申请·小小云服务器开通外网端口映射·



杜鹃座出口ip通过skyline上网络管理的vpc查eip获取

**popo测试服10.82.112.21:8091 -> 59.111.129.141:8091 添加映射**

wf申请添加映射后，测试内外网telnet不了；机器是galaxy管理的防火墙，需要添加绑定8091端口----新建端口，参考8081，绑定到服务vdev

**在galaxy服务绑好端口和白名单，生成配置同步后上管理机 cbc <ip> sync conf ,cbc <ip> iptables limit**

冼镇东 | SRE 18:19:32

正式环境就是用jupiter的

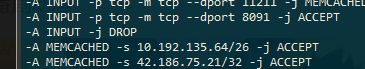
冼镇东 | SRE 18:21:05

应该说popo除了vdev这个机器以外都用的jupiter

**关于手动加防火墙未生效**

iptables -A INPUT -p tcp -m tcp --dport 8091 -j ACCEPT

-A，加到了最后，会被前面的规则drop掉



一般可以iptables-save出来，改完文本加iptables-apply回去稳一点

**Sym2各个环境，以及实例漂移宿主机**

Symphony值班 16:34:10

symv2 前端：https://sym2.nie.netease.com/

symv2 文档：https://doc.sym2.nie.netease.com/guide/quickstart/

接入指引：https://km.netease.com/wiki/show?page\_id=43419

新版 NCR dockerhub ：https://ncr.nie.netease.com/harbor/projects（左下角用户手册）

旧版 dockerhub：https://symphony.nie.netease.com/

配置网络（新建VPC、申请EIP） https://skyline.nie.netease.com/cld/network

开发环境：

https://sym2-office.nie.netease.com/

开发环境 dockerhub：

旧版杭州： https://hzindex.nie.netease.com/

旧版广州： https://gzindex.nie.netease.com/

新版：https://ncr-dev.nie.netease.com/

Symphony值班 16:34:16

https://sym2-office.nie.netease.com/

Symphony值班 16:34:19

这进去

Symphony值班 16:34:21

点高唐

Symphony值班 16:34:28

然后选择对应的项目，找到这个pod

Symphony值班 16:34:29

点一下重建

张日军 16:34:30

直接重启实例就漂移吗

Symphony值班 16:34:34

就飘到其他机器上了

Symphony值班 16:39:54

实例重启，看几个副本

Symphony值班 16:40:04

如果只有一个pod 那就没区别，都行

Symphony值班 16:40:20

多个pod ， 就看哪个pod 在这个宿主上， 然后重启这 一个pod就行

张日军 16:40:48

那就是只需要重启对应的pod，操作实例就行吗

Symphony值班 16:40:54

嗯嗯

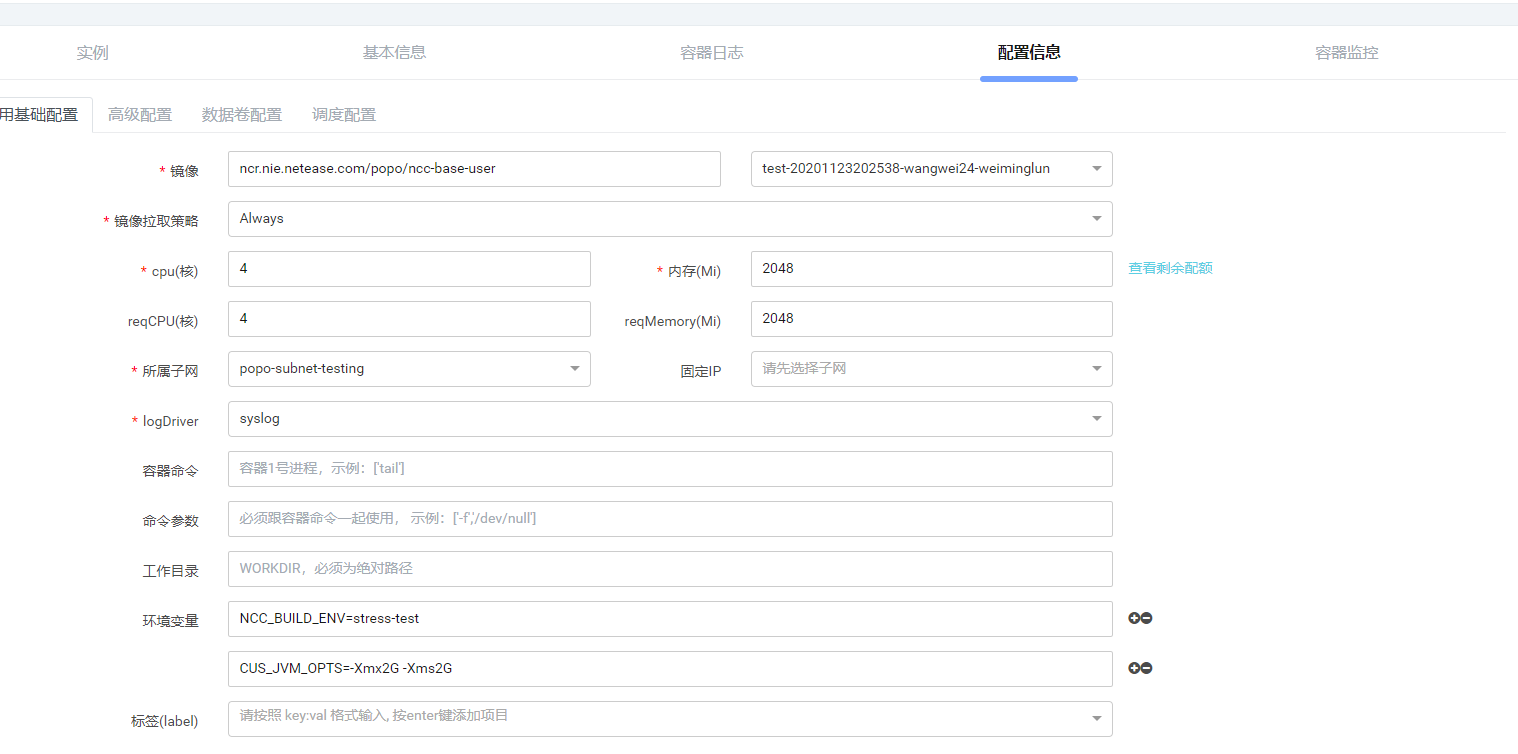
张日军 16:41:02

不需要管服务这里的重建

Symphony值班 16:41:25

因为一个服务可以对应多个pod ， 多个pod 可以分布在多个宿主上

**容器扩容：找到对应服务，调整配置信息后重建实例**



**安全漏洞--敏感目录泄露**

<https://soc.nie.netease.com/#/vuln/user-hole?id=6733>

根据报警里的链接，找到lbc的vip，去sym2上的负载均衡找到对应端口转发规则；

跟程序确认（popo的主要几个程序问下归谁使用即可），一般处理是将securityPolicy设置为REJECT，之后在安全组那里配置对应白名单，放开特定ip去访问；

最后点处理，补充说明；

端口开放office和vpn的可能会有误报，直接点处理即可。

**Apc gcp临时用户权限添加**

冼镇东 | SRE 13:12:34

智峰(范智锋) | PMTG|后端|56130 13:05:49

@冼镇东

给我开一下apc的用户 12主机

我要执行一个更新积分表的脚本

冼镇东 | SRE 13:12:41

日军来搞下试试

冼镇东 | SRE 13:13:06

临时开的话直接visudo改就行

##机器上直接visudo编辑即可，会保存到 cat /etc/sudoers里

**较长时间需要使用，但不加puppet里，则在/etc/sudoers.d里一个文件即可**

**root@popo-status01-status:/etc/sudoers.d# cat dev**

#popo ALL=(ALL:ALL) NOPASSWD: APT

chenjiecheng ALL=(ALL:ALL) NOPASSWD: /bin/su - popo

可以root切换的对应用户，然后测试，不用每次让用户去测试。

gameacademy或者popo项目，长期添加的需要修改puppet文件；

https://git-sa.nie.netease.com/puppet/gameacademy/blob/master/files/sudoers.gameacademy

路径gameacademy/ files/sudoers.gameacademy

或者添加阿拉丁插件添加https://g.126.fm/00RrsHi，需要监控系统上给对应项目绑定cbcadduser插件，之后添加阿拉丁流程。

**acp2001添加机器**

冼镇东 | SRE 16:40:44

跑初始化脚本的时候顺便刷一下acp管理机的防火墙（**这一步刷新管理机防火墙**）->确定新机器reboot后puppet没有报错->在管理机上**sync cbrc到新机器，sync conf**到新机器->刷新新机器的防火墙（**这一步刷新新机器防火墙**），新机器的防火墙刷新依赖cbrc模块

张日军 16:41:03

1

冼镇东 | SRE 16:42:27

40 8 \* \* \* apc cd /home/apc/cbc3/cbc && cbc hostlist/list-admin-admin galaxy\_api generate\_all\_ipset\_config all using\_and\_swap && cbc hostlist/list-all sync ipset ; cbc hostlist/list-all ipset ipset\_update\_setname whitlistcodesList Slice\_Skip\_While\_SetName\_Not\_Exist ; cbc hostlist/list-all ipset ipset\_init\_setname Slice\_Skip\_While\_SetName\_Not\_Exist >> ${LOG\_IPTABLES\_FD} 2>&1

冼镇东 | SRE 16:42:51

这是是apc刷新防火墙的cron，每天刷一次的。

冼镇东 | SRE 16:43:08

刷admin和新机器防火墙的时候用这个就行

**sync cbrc报错**

冼镇东 | SRE 18:02:34

新机器没有同步sync模块到新机器

冼镇东 | SRE 18:02:48

所以要用自带cbcsync

冼镇东 | SRE 18:03:04

正常是两个都新的

冼镇东 | SRE 18:00:01

跑一下**cbcsync cbrc**试试

冼镇东 | SRE 18:01:15

再跑一下**sync cbrc**

**cbc 42.186.20.114 cbcsync cbrc**

**cbc 42.186.20.114 sync cbrc**

**Dumbo创建新用户dumbo交付https://g.126.fm/01qHcje**

冼镇东 | SRE 17:48:34

show grants for projsa@'%';

看看projsa有啥权限，创建个假root就行

冼镇东 | SRE 17:48:52

GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER ON \*.\* TO 'root'@'%' WITH GRANT OPTION;

冼镇东 | SRE 17:51:31

就是这样，然后创建root用户的时候随机生成个强密码给他们就行

冼镇东 | SRE 17:51:56

我一般用这个生成密码https://suijimimashengcheng.51240.com/

ALTER USER 'root'@'%' identified by '密码'; ##密码字段不是password了，修改密码需要使用update user set authentication\_string

##高唐测试环境的网络，使用本机去连接即可；其他idc的，使用管理机去跳转，避免使用正式环境里的机器，避免误操作

**popo测试库10.82.112.20新建数据库并创建用户授权**

登录到机器后直接执行mysql进入数据库，不需要密码

CREATE DATABASE aitest\_popoim;

CREATE USER 'aitest\_popoim'@'%' IDENTIFIED BY 'TCzHv16TzCPc30lO';

GRANT ALL PRIVILEGES ON `aitest\_popoim`.\* TO 'aitest\_popoim'@'%'; ##对改用户授权所有权限，针对aitest\_popoim库里所有的表；

只授权部分权限，根据实际需要的权限对应调整

**创建超级用户**：CREATE USER 'root'@'%' IDENTIFIED BY 'vuIyzhi4NUEJZnrr';

**授权，参照projsa用户的权限**

GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES, INDEX, ALTER, SHOW DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER ON \*.\* TO 'root'@'%' WITH GRANT OPTION;

**mydumper测试库备份指定的表**

mydumper -u root -B popo -T pnotify -o /home/wb.zhangrijun/mysqlbak/

-v显示详细信息

**mydumper是支持对一张表多个线程备份的，参数-r**

**popo新增两个配置中心mysql库**

100434-NccConfigCenterUat popo-nccconfigcenteruat-100434-m.hz2.dumbo.nie.netease.com

root:4pcsT78ucDgCr5Sq

ncc\_config\_user：afDxsCOxDCw44UaF

创建项目库和登录用户

103694-NccConfigCenterProd popo-nccconfigcenterprod-103694-m.hz2.dumbo.nie.netease.com

root :vuIyzhi4NUEJZnrr

ncc\_config\_user：HeNONbO0HRtcxkgA

**tidb修改 root 密码**

mysql -u projsa -p -h popo-testing-200043-tidb.hz2.dumbo.nie.netease.com -P 3306

远程登陆 tidb，默认空密码

mysql -u root -p -h { tidb } -P 3306

修改 root 密码

ALTER USER 'root'@'%' IDENTIFIED BY 'xxx';

<https://docs.pingcap.com/zh/tidb/stable/alert-rules> ##文档指引

**Sym2宿主机更换，需要把旧宿主禁止调度，通知业务重建服务**

Popo杜鹃座3台10.191.204.66 10.191.204.67 10.191.204.68

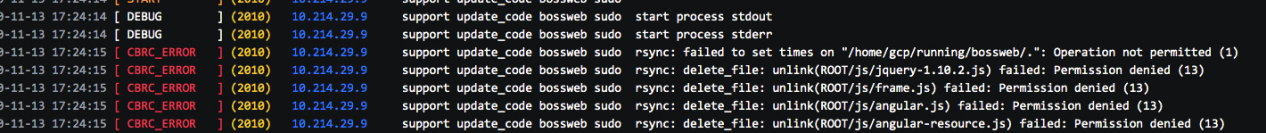
新申请一台10.191.204.65

财狼座5台7.32.52.10-14

Gameacademy 财狼一台7.32.51.33

<https://g.126.fm/00zsph2> 已经将旧宿主禁止调度

gcp阿拉丁流程部署失败https://g.126.fm/017fXNi



rsync: failed to set times on "/home/gcp/running/bossweb/.": Operation not permitted

权限不足，需要修改目录权限，这个目录开始权限是puppet，改为gcp报错，参考其他机器发现是tomcat8用户和群组，调整后正常

chown -R tomcat8:tomcat8 /home/gcp/running/bossweb/

**10.214.8.11数据库停止无法启动**

1. 一开始显示unmask，systemctl mask后启动报错，



root@acp-testdb02-700:/home/apc/var/mysql/log# systemctl start mysql

Failed to start mysql.service: Unit mysql.service is masked.

root@acp-testdb02-700:/home/apc/var/mysql/log# systemctl unmask mysql.service

Removed symlink /etc/systemd/system/mysql.service.

root@acp-testdb02-700:/home/apc/var/mysql/log# systemctl start mysql

Job for mysql.service failed. See 'systemctl status mysql.service' and 'journalctl -xn' for details.

root@acp-testdb02-700:/home/apc/var/mysql/log# journalctl -xn

-- Logs begin at 日 2020-09-13 04:20:01 HKT, end at 五 2020-11-13 18:08:04 HKT. --

11月 13 18:08:02 acp-testdb02-700.i.nease.net mysql[23432]: -su: /usr/bin/mysqld\_safe: No such file or directory

11月 13 18:08:02 acp-testdb02-700.i.nease.net su[23435]: pam\_unix(su:session): session closed for user mysql

11月 13 18:08:02 acp-testdb02-700.i.nease.net mysql[23432]: [61B blob data]

11月 13 18:08:02 acp-testdb02-700.i.nease.net mysql[23432]: error: 'Can't connect to local MySQL server through socket '/var/run/mysqld/mysqld.sock' (2)'

11月 13 18:08:02 acp-testdb02-700.i.nease.net mysql[23432]: Check that mysqld is running and that the socket: '/var/run/mysqld/mysqld.sock' exists!

11月 13 18:08:02 acp-testdb02-700.i.nease.net mysql[23432]: Mysql Server Failed to Start, Please check logs for more details..

11月 13 18:08:02 acp-testdb02-700.i.nease.net systemd[1]: mysql.service: control process exited, code=exited status=1

11月 13 18:08:02 acp-testdb02-700.i.nease.net systemd[1]: Failed to start LSB: Start/ Stop MySQL Community Server daemon.

-- Subject: Unit mysql.service has failed

-- Defined-By: systemd

-- Support: http://lists.freedesktop.org/mailman/listinfo/systemd-devel

--

-- Unit mysql.service has failed.

--

-- The result is failed.

11月 13 18:08:02 acp-testdb02-700.i.nease.net systemd[1]: Unit mysql.service entered failed state.

11月 13 18:08:04 acp-testdb02-700.i.nease.net sshd[23447]: Did not receive identification string from 10.90.64.27

root@acp-testdb02-700:/home/apc/var/mysql/log# free -m

**###最后是重装mysql，原来的数据库还是保持不变，重新读取数据库**

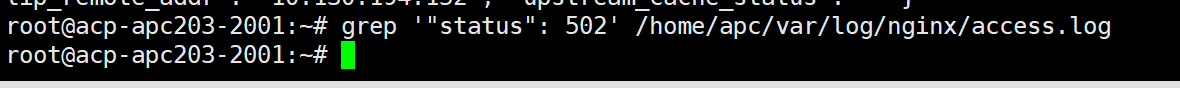
**Apt-get 重装，先把之前的配置conf文件备份，以防被覆盖掉**

**##apc 2001群组新加机器，挂到lbc 106.2.94.161测试**

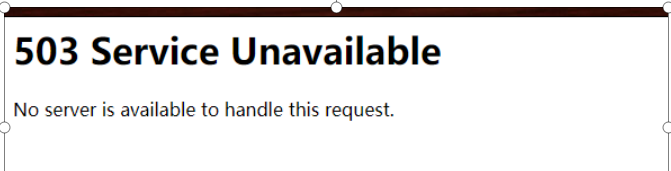
修改hosts文件 106.2.94.161 apc.pm.netease.com



配443端口的时候吧proxy protocol配成了否，导致访问lbc 502



调整proxy protocol为是后，访问是503，符合预期？



proxy protocol：haproxy 和 nginx 将会在发给客户端的包内插入客户端的真实 IP（仅限 TCP mode），如果 RS 不支持解析 proxy protocol 将会得到 400 bad request 等错误，因此，如果你不知道这是什么，请不要打开。

测试正常，之后需要在阿拉丁流程加上这台机器，指定操作对象为apc2服务角色

<https://aladdin.nie.netease.com/_acp/aladdin/processes/12310/execute>

**m.km.netease.com申请lbc海外加速**

先申请lbc，km-oversea实例，指向当前域名的指向服务ip 113.108.224.243



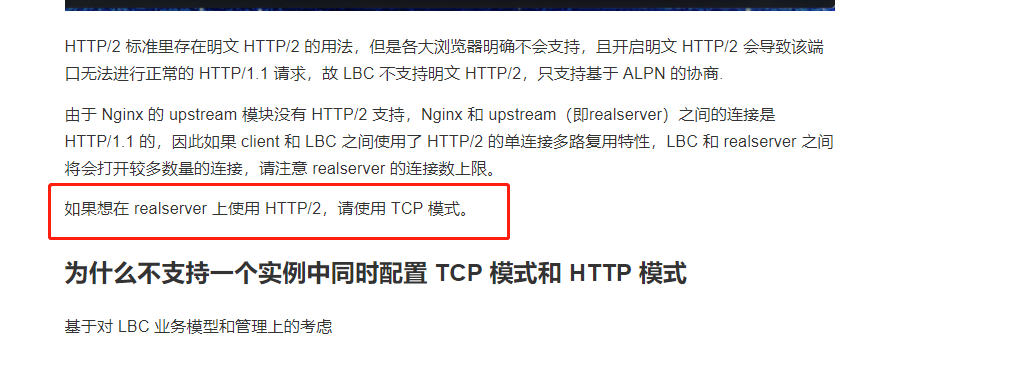
启用uri转发，default配置权重为0，域名不对直接返回502

之后邮件申请，根**据lbc回复配置域名cname；原来A记录也要修改为cname**

指向m.km.ntes53.netease.com.

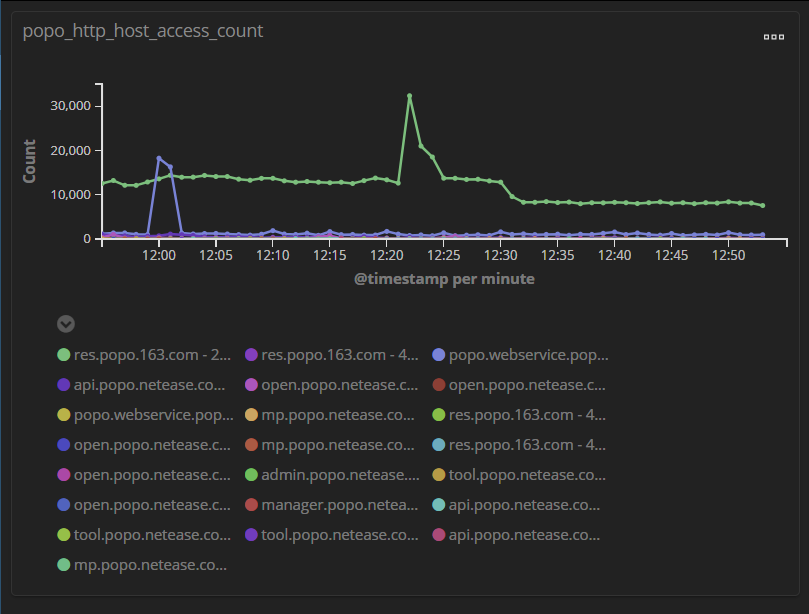
<https://help.aliyun.com/knowledge_detail/39787.html>

rs如果需要使用http2，端口选择tcp模式



**popo请求连接dashboard图**

<http://elk.x.netease.com/goto/9a734004a397ba2cdc346153acead183>



**大厦办公网异常中断导致大量重连，引发机器负载异常**

报警机器人 12:23:09

[网易泡泡][网卡流量监控][2020-12-07 12:23:02]

机器:

42.186.33.95|popo-app01-app|POPO应用服务群组:网卡每秒接收包数量 {bond1:[218062]} 大于 200000,网卡每秒发送包数量 {bond1:[224511]} 大于 200000

值班SA：张日军, 17608990421, wb.zhangrijun@mesg.corp.netease.com

详情页: https://g.126.fm/03iBv1e

报警机器人 12:24:06

[网易泡泡][TCP端口异常][2020-12-07 12:24:06]

机器:

42.186.20.89|popo-app04-app|POPO应用服务群组:10.200.101.89:443

值班SA：张日军, 17608990421, wb.zhangrijun@mesg.corp.netease.com

详情页: <https://g.126.fm/03VME8w>

**Popo ai搜索 elk es服务**

<http://elk-saas.nie.netease.com/saas/project/popo>在这个url进入实例具体界面，然后跳转到kibana管理界面，下面访问url不是用来直接登录连接的。

正式popo-214.elk.x.netease.com

测试popo-215.elk.x.netease.com



**小小云测试lbc popo\_vdev实例添加端口**

1.10.82.112.21这台对应

59.111.129.141:10443

http://59.111.129.141:8081

2.10.82.112.23这台对应

59.111.129.141:13443

http://59.111.129.141:8083

现在希望把10.82.112.22也同样开一下 目前并行项目多测试环境不够

59.111.129.141:12443

<http://59.111.129.141:8082>

机器上监听的是10443和8080,8081，作为rs的端口----8080是tomcat的端口，8081是nginx端口；具体使用哪一个，需要程序确认。

Lbc的端口通过8082，8083以及13443，12443等来区分不同的端口

<https://hz.lbc.nie.netease.com/_popo/lbc/instances?_num=50&_page=1&virtualip.ip=42.186.122.253>

#10.82.112.22防火墙未开8080，导致lbc配置后端口依然不通，需要修改防火墙；

正常流程是galaxy上新建一个端口，绑定到vdev服务，生成同步配置后到管理机操作

cbc 同步的机器ip sync conf，再cbc <ip> iptables limit生效新防火墙。

不过不确定这个端口其他机器是否需要放通，就没使用galaxy，直接手动机器上修改添加。

**机器添加项目用户后无法通过密钥对登录**

debug1: No more authentication methods to try.

Permission denied (publickey).

排查：

systemctl status sshd ##提示用户被锁

Dec 09 16:26:26 popo-dev01-dev sshd[25943]: User frontend not allowed because account is locked

Dec 09 16:26:26 popo-dev01-dev sshd[25943]: input\_userauth\_request: invalid user frontend [preauth]

tail -f /var/log/auth.log ##同样的报错

参考解决

<https://unix.stackexchange.com/questions/193066/how-to-unlock-account-for-public-key-ssh-authorization-but-not-for-password-aut>

usermod -p '\*' username ##把用户设置成禁止密码登录后秘钥登陆成功

**win机器远程桌面密码qwer;1234**

10.214.0.198|popo-doc01-win|windows机器

win gpostd管理https://g.126.fm/01jyKnS

重启Agent:

/etc/rc.d/gpostd restart

查看状态和信息

检查Agent运行状态：

/etc/rc.d/gpostd status

查看Agent日志：

/etc/rc.d/gpostd tail

**12-13popo富春云容器宿主故障需要重建实例**



操作异常问题：

1. Namespace没有权限，无法通过api操作其上的实例----10个容器需要操作，其中6个没有权限。Postman操作一通后，应该成功四个（分配到零散的宿主上）；后面程序上线帮忙操作另外6个，刚好碰上资源不足，申请资源后这六个都在7.32.52.145这个新宿主上
2. 节点资源不足，导致无法创建

Postman header里加以下参数

X-Auth-Token:

X-Auth-Project:popo

X-Auth-Region:cld\_hz\_fcy\_lupus

post 方法api.cube.nie.netease.com/api/v2/cube/process

body内容

{

"process": {

"type": "evacuate",

"ip": "10.213.35.10" ##需要重建的实例ip

}

}

### **[elk -> popo] 集群状态不为 Green**

[重复通知][elk->popo][警告][集群状态不为 Green]2020-12-14 17:12:50

有索引缺失副本，如持续出现可能潜在问题

当前共1个，警告1个

影响:

popo-im-es-test|215 当前值: 1 警告

如未及时处理，请联系值班:

姓名: 林邦骏 手机号: 18588717163

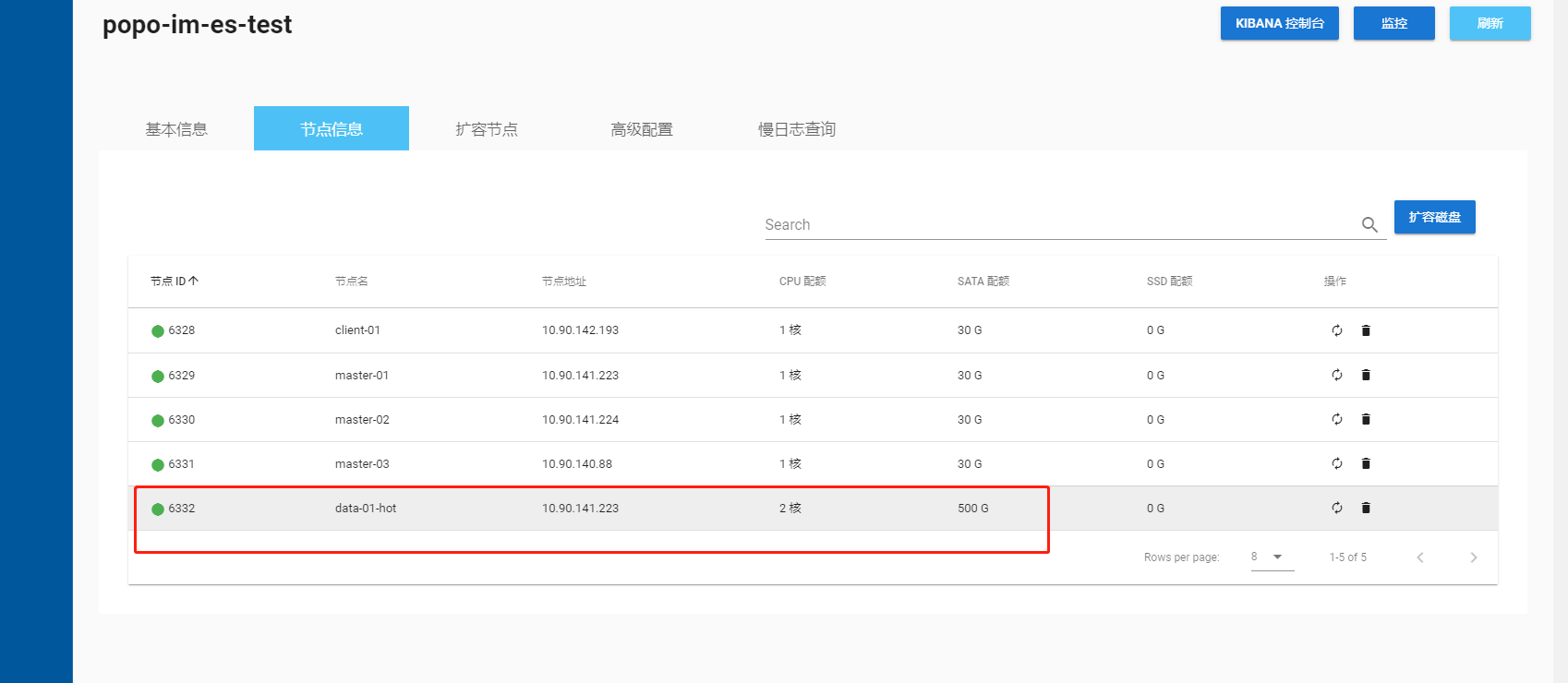
处理: https://g.126.fm/02UW3tO

[触发保底通知策略]所有策略的通知对象为空，报警到项目SA值班

因为测试实例单节点，导致主备shard无法分配，从而引发报警；

https://blog.csdn.net/u011127242/article/details/106883754

业务确认可以忽略，先屏蔽报警。



**数据库查询语句监控--插件**

select count(\*) as trx\_count from information\_schema.INNODB\_TRX a join information\_schema.processlist b ' \

'on a.trx\_mysql\_thread\_id=b.id where b.time>10800 and b.user not in ("dumbo\_rpl","monitor","event\_scheduler","root","system user")

**omega机器访问gcp lbc不通**

炒面(刘朝敏) 14:02:10

邹小波12:26:03

[img]bbdd30fdc159925a093d539f99ec8491[/img]我在自己本机还能访问，但是在Omega作业里反而访问受限了，这个 要怎么处理

彭翌12:41:27

白名单的问题？

炒面(刘朝敏) 14:03:20

@冼镇东 帮忙看看网络访问的问题

炒面(刘朝敏) 14:03:30

http://apiauth2.pm.netease.com

新接入项目请使用http://apiauth2.pm.netease.com域名，原域名后续会被废弃，接入事项请看 apiauth2迁移事项

如果服务器在小小云，使用internal-apiauth-gateway.netease.com访问

##

域名分别指向sym2两个内外网lbc（宝瓶座）10.204.2.212，42.186.120.193；

内网没有白名单限制，建议优先使用；外网访问需要对应地加来源ip白名单

**popo畅写服务访问控制**

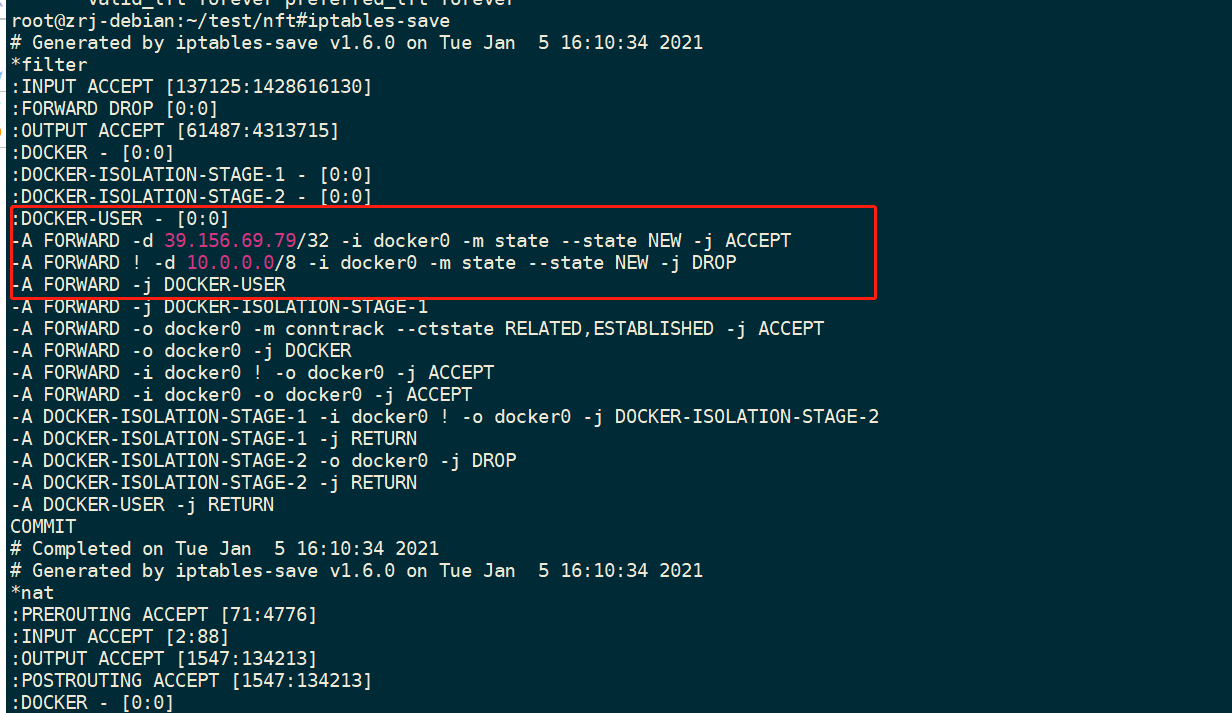
对外接lbc；docker对外访问通过防火墙配置；

<https://www.jianshu.com/p/811d84fa0fe9>

# 禁止容器访问外部所有网络

sudo iptables -A FORWARD -i docker0 -d 0.0.0.0/0 -m state --state NEW -j DROP

##用-A看能否插入到最后



-A FORWARD -d 39.156.69.79/32 -i docker0 -m state --state NEW -j ACCEPT

-A FORWARD ! -d 10.0.0.0/8 -i docker0 -m state --state NEW -j DROP

目前通过放通特定外网，限制为特定内网访问；

如果是多容器，需要确认网桥网卡，指定对应网卡的链去添加

测试环境的10.214.0.214上面多个容器，默认网桥不是docker0，是br-32a522b338bc，应该需要加到对应的链里