**腾讯专有云**

**云服务器（CVM）**

**部署操作手册**

**文档版本：04**

**发布日期：2021年12月13日**

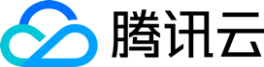
|  |  |
| --- | --- |
| 腾讯云计算（北京）有限责任公司 |  |

**【版权声明】**

本文档著作权归腾讯云计算（北京）有限责任公司（以下简称“腾讯云”）单独所有，未经腾讯云事先书面许可，任何主体不得以任何方式或理由使用本文档，包括但不限于复制、修改、传播、公开、剽窃全部或部分本文档内容。

本文档及其所含内容均属腾讯云内部资料，并且仅供腾讯云指定的主体查看。如果您非经腾讯云授权而获得本文档的全部或部分内容，敬请予以删除，切勿以复制、披露、传播等任何方式使用本文档或其任何内容，亦请切勿依本文档或其任何内容而采取任何行动。

**【商标声明】**

“腾讯”、“腾讯云”及其它腾讯云服务相关的商标、标识等均为腾讯云及其关联公司各自所有。若本文档涉及第三方主体的商标，则应依法由其权利人所有。

**【免责声明】**

本文档旨在向客户介绍本文档撰写时，腾讯云相关产品、服务的当时的整体概况，部分产品或服务在后续可能因技术调整或项目设计等任何原因，导致其服务内容、标准等有所调整。因此，本文档仅供参考，腾讯云不对其准确性、适用性或完整性等做任何保证。您所购买、使用的腾讯云产品、服务的种类、内容、服务标准等，应以您和腾讯云之间签署的合同约定为准，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【修订记录】

| 编号 | 修改日期 | 修改人 | 更新内容 |
| --- | --- | --- | --- |
| 04 | 2021-12-13 | shadowliu | 1. 修改了dcos的arm的装机镜像，从tlinux2.4 tk4 0004改成tlinux 2.4 tk3 0004 |
| 03 | 2021-05-13 | shadowliu | 1. 增加了项目管理的服务注册步骤 2. 优化了moa切换tgw的步骤 3. 增加了cdh前端包的部署 4. 去掉了product-cvm-yunjing-agent（废弃）的部署步骤，这部分由yunjing部署时部署 5. 去掉了product-cvm-agent（废弃） 包的部署 6. cvm的投放模版进行了说明 |
| 02 | 2021-01-25 | qsli | 第二次发布。增加“moa访问链路实施tgw”。 |
| 01 | 2021-01-11 | shadowliu | 第一次发布。更新成CSIG标准模板。 |

目 录

[【修订记录】 1](#_Toc90400250)

[目 录 2](#_Toc90400251)

[1 部署前说明 1](#_Toc90400252)

[2 prm部署 2](#_Toc90400253)

[2.1 安装包检查 2](#_Toc90400254)

[2.2 管控部署 2](#_Toc90400255)

[3 cvm部署 4](#_Toc90400256)

[3.1 安装包检查 4](#_Toc90400257)

[3.2 前端部署 6](#_Toc90400258)

[3.3 管控部署 6](#_Toc90400259)

[3.4 生产节点部署 7](#_Toc90400260)

[4 moa的tgw规则创建 10](#_Toc90400261)

[4.1 准备工作 10](#_Toc90400262)

[4.2 实施方案 10](#_Toc90400263)

[4.2.1 整体思路 10](#_Toc90400264)

[4.2.2 执行步骤 10](#_Toc90400265)

[5 母机投放 14](#_Toc90400266)

[6 项目管理注册 19](#_Toc90400267)

[7 标签服务注册 20](#_Toc90400268)

部署前说明

文档适用 TCE 380 CVM部署，请注意以下约束：

* **建议在切换keepalived切换tgw完成后开始部署PRM和CVM**
* **现场环境包的版本可能与文档示例不一样，按照现场的包版本执行即可**
* **370以后，前端包拆分由各服务单独部署**
* **380以后，imagestage组件容器化**
* **380以后，dcos发生变化，母机添加到cvm资源池方式发生变化**

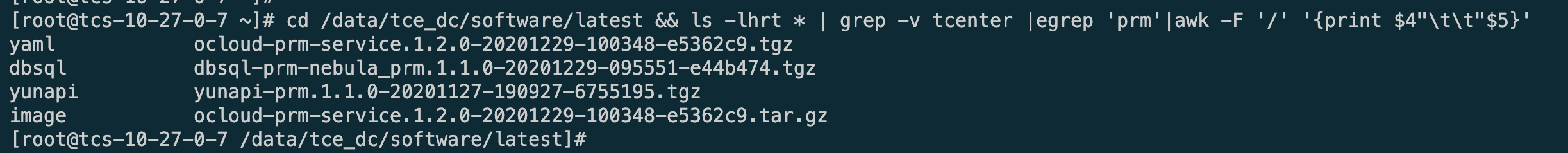
prm部署

prm是宿主机投放系统，主要负责underlay和overlay宿主机的投放。

prm只有一个镜像组件，一个数据库组件，没有生产组件，没有单独的前端包。

安装包检查

1. 登录master的部署机，必须有如下包，可能版本有差异。
2. 执行下面命令：

cd /data/tce\_dc/software/latest && ls -lhrt \* |grep -v tcenter| egrep 'prm'|awk -F '/' '{print $4"\t\t"$5}'

管控部署

开始部署(以下操作均在tcs-master执行)。

1. 刷新配置包：

kaleido\_config update –prepare

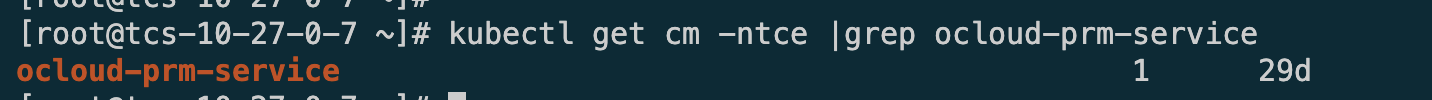
1. 刷新configmap：

echo ocloud-prm-service > /mnt/prm.list

kaleido\_config update /mnt/prm.list

检查：

kubectl get cm -ntce |grep ocloud-prm-service

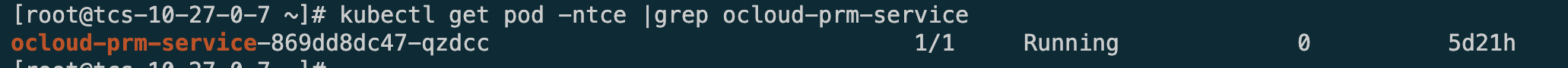


1. 拉起管控pod:

kaleido\_image upgrade /data/tce\_dc/software/latest/yaml --filter prm

检查：

kubectl get pod -ntce |grep prm



1. 刷新yunapi-prm

kaleido\_yunapi update yunapi-prm

检查：无报错

cvm部署

cvm 380以后，生产组件imagestage容器化，拉起pod即可。

安装包检查

|  |  |  |
| --- | --- | --- |
| 类别 | 数量 | 备注 |
| yaml | 16 | 和image一一对应 |
| image | 16 | 和yaml一一对应 |
| dbsql | 6 | 数据库包 |
| preset | 4 | cam和站内信相关 |
| product |  | 主要有product-cvm-basic-import、product-cvm-barad-agent、product-cvm-metadata、product-cvm-image等， |
| product前端包 | 4 | product-frontend-imgcache-ocloud-cvm、product-ffrontend-imgcache-qcbuy-cvm、product-frontend-imgcache-tcloud-cvm、product-ffrontend-imgcache-qcbuy-cdh |

登录tcs-master部署机，执行如下命令：

cd /data/tce\_dc/software/latest && ls -lhrt \* |grep 'cvm'|awk -F '/' '{print $4"\t\t"$5}'

yaml ocloud-cvm-api.100.4.2-20210106-020603-be86ad7.tgz

yaml ocloud-cvm-cvm.3.6.3-20200904-145249-acefe4b.tgz

yaml ocloud-cvm-keymanage.1.2.0-20201109-162534-e2b0933.tgz

yaml ocloud-cvm-logstash.1.2.0-20201109-162115-bacb75e.tgz

yaml ocloud-cvm-moamr.1.3.0-20201214-144356-54d9ec7.tgz

yaml tcloud-cvm-api.1.2.1-20210101-050750-eb06534.tgz

yaml tcloud-cvm-bill.1.2.1-20210101-010532-0a37678.tgz

yaml tcloud-cvm-ccdb.1.2.1-20210105-060434-770e22b.tgz

yaml tcloud-cvm-cgw.3.2.0-20201111-155116-64cc19c.tgz

yaml tcloud-cvm-des.3.2.0-20201224-204209-c9e6fbd.tgz

yaml tcloud-cvm-event.1.3.0-20201201-020532-0a425fd.tgz

yaml tcloud-cvm-image-cgw.1.3.0-20201230-125759-cc82073.tgz

yaml tcloud-cvm-image-des.3.1.0-20201224-155659-5d6c839.tgz

yaml tcloud-cvm-imagestage.1.1.1-20210105-203232-a3e4e45.tgz

yaml tcloud-cvm-vnc.1.2.0-20201109-162525-4d0084e.tgz

yaml tcloud-cvm-vstation.1.2.1-20201231-020416-a336dc1.tgz

dbsql dbsql-cvm-des\_db.1.3.0-20201109-161733-f241bd2.tgz

dbsql dbsql-cvm-metadb.1.1.0-20201109-161815-f241bd2.tgz

dbsql dbsql-cvm-MonitorAgentData.1.2.0-20201109-161818-f241bd2.tgz

dbsql dbsql-cvm-sec\_key\_manage.1.2.0-20201109-161819-f241bd2.tgz

dbsql dbsql-cvm-yhdb\_vstation.1.3.0-20201217-113052-fffb04b.tgz

dbsql dbsql-cvm-yhimage.1.5.0-20201109-161823-f241bd2.tgz

yunapi yunapi-cvm.1.1.2-20210104-165942-90663e2.tgz

preset preset-cvm-cam-ocloud-opcvm.1.0.0-20200911-162939-fe816fc.tgz

preset preset-cvm-cam-tcloud-cvm.1.0.0-20200901-175906-faccbbf.tgz

preset preset-cvm-message-svr-ocloud-CVM.1.1.0-20201109-162845-70b66ce.tgz

preset preset-cvm-message-svr-tcloud-CVM.1.1.0-20201230-175339-0c647ca.tgz

product product-cvm-barad\_agent.1.3.1-20201230-175957-7f2ea61.tgz

product product-cvm-basic.1.0.0-20200810-112145-2080a2a.tgz

product product-cvm-basic-import.1.0.0-20200810-112145-2080a2a.tgz

product product-cvm-basic-windows-import.1.0.0-20200810-112145-2080a2a.tgz

product product-cvm-image.3.2.1-20201230-180133-7f2ea61.tgz

product product-cvm-image-centos7\_6.1.1.0-20201204-173033-fa855d5.tgz

product product-cvm-image-tlinux2\_2.1.0.0-20200925-113503-c493620.tgz

product product-cvm-image-win2016cn.1.1.0-20201230-163723-f431bce.tgz

product product-cvm-image-win2016en.1.0.0-20201230-181212-4d2eb3c.tgz

product product-cvm-image-win2019cn.1.1.0-20201204-153722-085c09e.tgz

product product-cvm-image-win2019en.1.1.0-20201204-153321-1df374b.tgz

product product-cvm-metadata.1.1.0-20201109-163006-6d02649.tgz

product product-frontend-imgcache-ocloud-cvm.1.1.0-20201228-205418-9e7add1.tgz

product product-frontend-imgcache-qcbuy-cvm.1.2.1-20201231-173703-7ee7ceb.tgz

product product-frontend-imgcache-tcloud-cvm.1.1.1-20201231-163127-1ff7869.tgz

product product-frontend-imgcache-tcloud-cdh.1.1.1-20201228-163127-1ff7869.tgz

image ocloud-cvm-api.100.4.2-20210106-020603-be86ad7.tar.gz

image ocloud-cvm-cvm.3.6.3-20200904-145249-acefe4b.tar.gz

image ocloud-cvm-keymanage.1.2.0-20201109-162534-e2b0933.tar.gz

image ocloud-cvm-logstash.1.2.0-20201109-162115-bacb75e.tar.gz

image ocloud-cvm-moamr.1.3.0-20201214-144356-54d9ec7.tar.gz

image tcloud-cvm-api.1.2.1-20210101-050750-eb06534.tar.gz

image tcloud-cvm-bill.1.2.1-20210101-010532-0a37678.tar.gz

image tcloud-cvm-ccdb.1.2.1-20210105-060434-770e22b.tar.gz

image tcloud-cvm-cgw.3.2.0-20201111-155116-64cc19c.tar.gz

image tcloud-cvm-des.3.2.0-20201224-204209-c9e6fbd.tar.gz

image tcloud-cvm-event.1.3.0-20201201-020532-0a425fd.tar.gz

image tcloud-cvm-image-cgw.1.3.0-20201230-125759-cc82073.tar.gz

image tcloud-cvm-image-des.3.1.0-20201224-155659-5d6c839.tar.gz

image tcloud-cvm-imagestage.1.1.1-20210105-203232-a3e4e45.tar.gz

image tcloud-cvm-vnc.1.2.0-20201109-162525-4d0084e.tar.gz

image tcloud-cvm-vstation.1.2.1-20201231-020416-a336dc1.tar.gz

前端部署

部署节点：tcs-master部署机。

1. 部署cvm管控端

kaleido\_product install product-frontend-imgcache-ocloud-cvm --action=trender

kaleido\_product install product-frontend-imgcache-ocloud-cvm --action=deploy

1. 部署cvm租户端

kaleido\_product install product-frontend-imgcache-tcloud-cvm --action=trender

kaleido\_product install product-frontend-imgcache-tcloud-cvm --action=deploy

1. 部署租户端cvm 购买页

kaleido\_product install product-frontend-imgcache-qcbuy-cvm --action=trender

kaleido\_product install product-frontend-imgcache-qcbuy-cvm --action=deploy

1. 部署租户端cvm购买页

kaleido\_product install product-frontend-imgcache-qcbuy-cdh --action=trender

kaleido\_product install product-frontend-imgcache-qcbuy-cdh --action=deploy

管控部署

部署节点：tcs-master部署机。

1. 配置包更新：

kaleido\_config update --prepare

检查：无报错

1. configmap拉起：

cd /data/tce\_dc/software/latest/image && ls -lhrt \* |grep 'cvm'|awk -F '/' '{print $5}'|cut -d"." -f1 > /mnt/cvm.list

kaleido\_config update /mnt/cvm.list

检查：无报错

1. pod拉起：

kaleido\_image upgrade /data/tce\_dc/software/latest/yaml/ --filter cvm

检查：

kubectl get pod -ntce|grep cvm

所有pod为running

图片包含 文本

描述已自动生成

1. 导入预设数据：

kaleido\_preset update /data/tce\_dc/software/latest/preset/preset-cvm-cam-ocloud-opcvm.1.0.0-20200911-162939-fe816fc.tgz

kaleido\_preset update /data/tce\_dc/software/latest/preset/preset-cvm-cam-tcloud-cvm.1.0.0-20200901-175906-faccbbf.tgz

kaleido\_preset update /data/tce\_dc/software/latest/preset/preset-cvm-message-svr-ocloud-CVM.1.1.0-20201109-162845-70b66ce.tgz

kaleido\_preset update /data/tce\_dc/software/latest/preset/preset-cvm-message-svr-tcloud-CVM.1.1.0-20201230-175339-0c647ca.tgz

1. 刷新yunapi：

kaleido\_yunapi update yunapi-cvm

生产节点部署

生产组件全部在tcs-master-installer上部署。

1. product-cvm-barad\_agent 部署：

将software 下面的包上传到hdfs

kaleido\_product install product-cvm-barad\_agent --action=trender

kaleido\_product install product-cvm-barad\_agent --action=deploy

1. product-cvm-basic部署

将software下面的包上传到hdfs

kaleido\_product install product-cvm-basic --action=trender

kaleido\_product install product-cvm-basic --action=deploy

1. product-cvm-basic-import部署

\*\* 依赖csp/cos部署完成，部署过程实际是将software下的包上传到hdfs

kaleido\_product install product-cvm-basic-import --action=trender

kaleido\_product install product-cvm-basic-import --action=deploy

kaleido\_product install product-cvm-basic-windows-import --action=trender

kaleido\_product install product-cvm-basic-windows-import --action=deploy

1. product-cvm-image部署：

主要作用是将上传创建overlay cvm和underlay cvm时候需要的镜像到hdfs，数据库需要录入的镜像信息已经在拉起pod初始化dbsql的时候完成。由于涉及的镜像较多，需要找一台公网的机器下载镜像

kaleido\_product install product-cvm-image --action=trender

cd /data/tce\_dc/workspace/TCE

将download.sh的脚本复制到一台可以访问公网的机器，并执行脚本下载镜像，这里需要一个空间较多的机器

将下载的镜像复制到/data/tce\_dc/workspace/software/ 下面，并执行下面命令安装

kaleido\_product install product-cvm-image --action=deploy

1. 其余镜像部署

\*\* 依赖 cos/csp部署完成

例如product-cvm-image-centos7\_6.1.1.0-20201204-173033-fa855d5.tgz、product-cvm-image-win2019en.1.1.0-20201204-153321-1df374b.tgz等格式为product-cvm-image-镜像类型\*.tgz的镜像部署方式相同，如下

kaleido\_product install product-cvm-image-centos7\_6 --action=trender

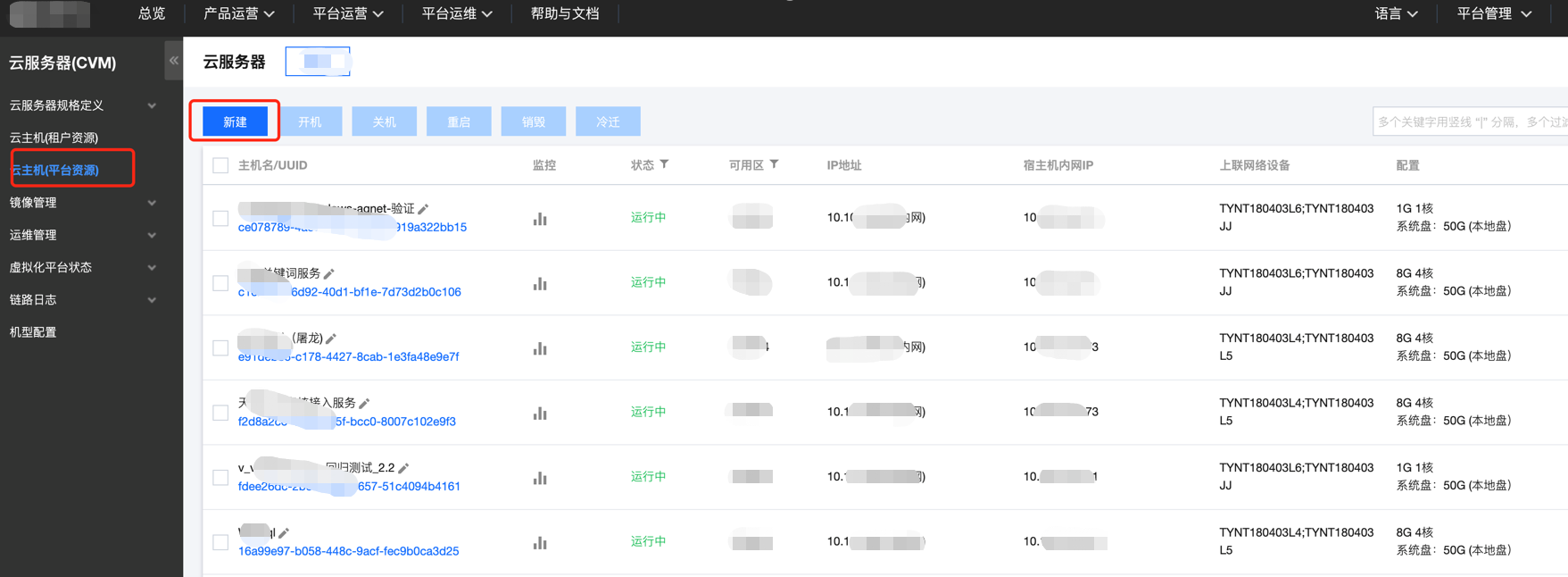
kaleido\_product install product-cvm-image-centos7\_6 --action=deploy

1. product-cvm-metadata部署：

## metadata安装可以选择物理服务器或者underlay cvm，underlay cvm依赖prm宿主机投放完成，请先看本文第5节-母鸡投放

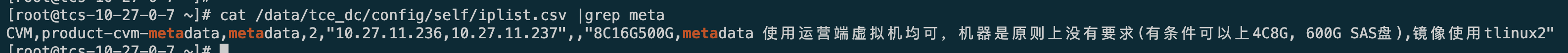
配置：underlay cvm（8C 16G 500G）

在运营端控制台购买两台underlay cvm



社交网络的手机截图

描述已自动生成



记录购买的metadata服务器ip，并将写入iplist

在tcs-master执行如下命令：

刷新配置：

kaleido refresh iplist

kaleido update config –prepare

部署metadata：

kaleido\_product install product-cvm-metadata --action=trender

kaleido\_product install product-cvm-metadata --action=deploy

moa的tgw规则创建

准备工作

**检查包版本：**

ocloud-cvm-moamr 包版本**大于等于** 1.1.1-20201208-190432-4c82c8c 时

实施方案

整体思路

CVM宿主机通过udp协议访问ocloud-cvm-moamr服务上报母机的信息，为规避kubernetes Service iptables链路问题，需要手动使用CLB创建一个udp规则供宿主机上报信息。

需要执行两个步骤：

1.  手动创建TGW规则

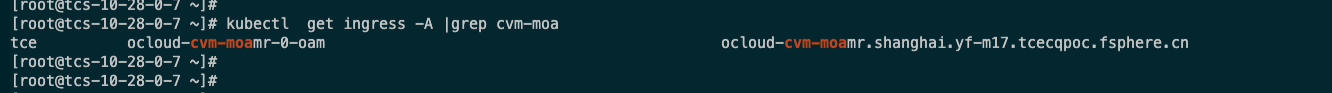
2.  新增域名解析，使生产流量指向手动创建的VIP规则

执行步骤

确认需要新增的域名

在tcs master上执行下面命令

kubectl get ingress -A |grep cvm-moa



例如这里获取到的域名为ocloud-cvm-moamr.shanghai.yf-m17.tcecqpoc.fsphere.cn，则建议客户环境域名为

udp.ocloud-cvm-moamr.shanghai.yf-m17.tcecqpoc.fsphere.cn

手动新增4层tgw规则

登陆运营端 -–>产品运营 -–> CLB ，申请新的CLB规则

图形用户界面, 应用程序

描述已自动生成

点击提交后到LB业务运维审核该规则并赋予vip

图形用户界面, 应用程序, 网站

描述已自动生成

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

查看我们刚刚申请的4层规则

应用程序

中度可信度描述已自动生成

点击管理，点击管理绑定后端rs

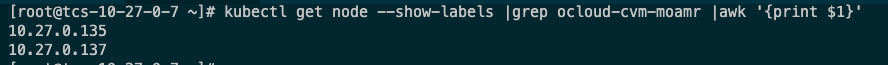
图形用户界面, 应用程序

描述已自动生成

\*\* 这里填写的ip为下面查到的ip，因为ocloud-cvm-moamr的pod是hostnetwork的，并且udp的规则是通过ping去检测，所以这里建议拉起的ocloud-cvm-moamr的pod数量和打了标签的node数量一样，并将所有的nodeip加到RS里面。

在tcs-master上执行如下命令

kubectl get node --show-labels |grep ocloud-cvm-moamr |awk '{print $1}'



指向moa容器所在node节点, 协议:udp, 端口:9600, 后端也是9600, udp

配置域名解析

1. 在master节点查询本集群基础信息。

curl -X GET http://127.0.0.1:30150/api/v1alpha1/peer/local | python -m json.tool

1. 获取到返回。

{

"cluster": "8b135502-7f15-469a-99a1-e24e11f8b9e3",

"id": "8b135502-7f15-469a-99a1-e24e11f8b9e3",

"name": "8b135502-7f15-469a-99a1-e24e11f8b9e3",

"region": "chongqing"

}

1. 根据上述返回结果, 构造请求, 创建DNS解析。

curl -X POST --header "Content-Type:application/json" http://127.0.0.1:30150/api/v1alpha1/service/register -d '{"serviceID" : "ocloud-cvm-moamr.udp", "region" : "替换", "underlay" : true, "cluster" : "替换", "zone" : "0", "endpoint" : { "host" : "*udp.ocloud-cvm-moamr.wh.yun.ccb.com*(替换为4.2.2.1自定义的域名)", "ip" : "(# 手动创建的tgw规则分配的ip)", "port" : "9600" }}'

变更后检查

nslookup *${前面自定义的域名}*

**预期获得IP**： (# 手动创建的tgw\_ip)。

母机投放

380母机投放之前需要先把宿主机从dcos划分到cvm资源池

前置准备：

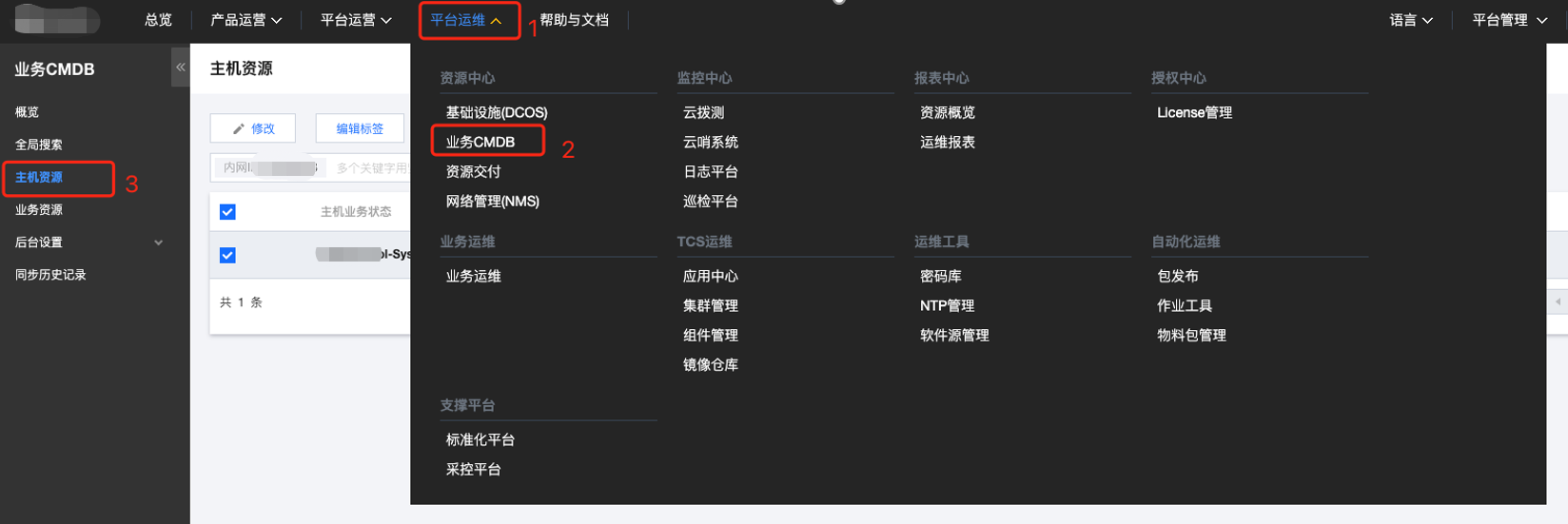
* 母机信息已经录入dcos并安装操作系统。
* 母机操作系统：

X86环境：TencentOS Server 2.2（TK2-0052），装机完成后uname -r查看内核为0052。

ARM环境：TencentOS Server 2.4（TK3-0004）

1.划分母机到cvm资源池（underlay和overlay母机划分方式相同）。

登录管控端，依次点击 平台运维-业务CMDB-主机资源。



输入母机ip，搜索母鸡，勾选并选择CVM资源池，确认划分。

图形用户界面, 应用程序

描述已自动生成

点击业务资源-选择cvm-搜索ip 可以查找到划分到cvm资源池的物理机，将机器划分到cvm资源池，不要将机器再单独划分到underlay或者overlay资源池，否则再投放界面无法选取机器。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

母机划分完成。

2.母机投放

前置准备：

* 母机在dcos五张表信息录入无误并完成装机
* 母机配置ntp时间同步：与tcloud-cvm-vstation、ocloud-cvm-moamr等pod所在node时间同步
* 母机yum源可用、dns配置正确
* moa的tgw规则创建完成

登录运营端 依次点击产品运营-宿主机投放



按照实际需求完成投放

underlay cvm： 投放类型：基础网络：kvm3.0\_非vpc母机

KVM30 Underlay Host for ARM(HuaweiTaishanV2): kvm3.0 arm鲲鹏 cpu underlay母机模版

KVM30 Underlay Host for NFV Gateway：给x86架构的网络NFV 母机是用的模板

KVM30 Underlay Host for NFV SR-IOV Gateway：给x86架构的网络NFV 母机是用的模板

kvm3.0 非vpc母机：x86 常用underlay 模版

KVM30 Underlay Host for ARM(Phytium): kvm3.0 arm飞腾 cpu underlay母机模版

图形用户界面, 文本, 应用程序

描述已自动生成

overlay cvm： 投放类型：vpc网络  
KVM30 BigData VPC Host for ARM(HuaweiTaishanV2) :arm的大数据机型模版

KVM3.0\_vpc母机:x86常用模版，大多数x86机型都是用它

KVM30 VPC Host for ARM(HuaweiTaishanV2): kvm3.0 arm鲲鹏 cpu overlay母机模版

KVM30 VPC Host for ARM(Phytium): kvm3.0 arm飞腾 cpu overlay母机模版

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

图形用户界面, 应用程序

描述已自动生成

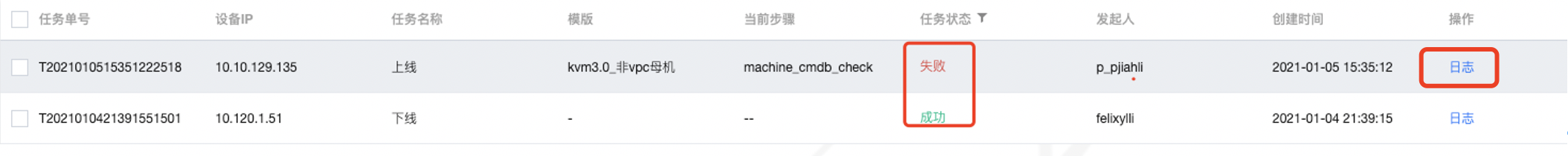
任务状态成功则表示投放成功：



3.投放失败的排查办法

目前有三个地方提供了报错日志：

* 投放任务失败后的日志按钮



* prm的pod ocloud-prm-service 可以根据上图任务id（任务单号）查询

目录 /data/log/prm\_framework

* 投放的母鸡的/tmp目录下

可以查询nohup.out的文件

项目管理注册

380新增了项目管理功能，需要在注册，在tcs-master上执行以下请求

curl http://tcloud-tcenter-project-org.tcenter:50050 -d '{

"APIModule":"tpo",

"Action":"SubscribeTopic",

"Version":"2020-09-20",

"RequestId":"12345612345",

"ServiceType":"cvm",

"Region":"shanghai",

"ResourcePrefix":"ins",

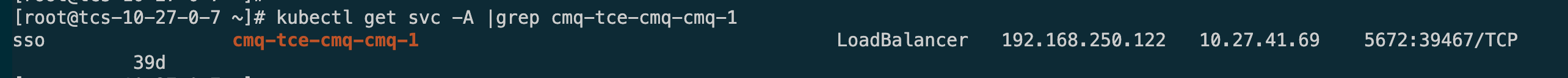
"Endpoint":"http://tcloud-cvm-event.cvm:80"

}'

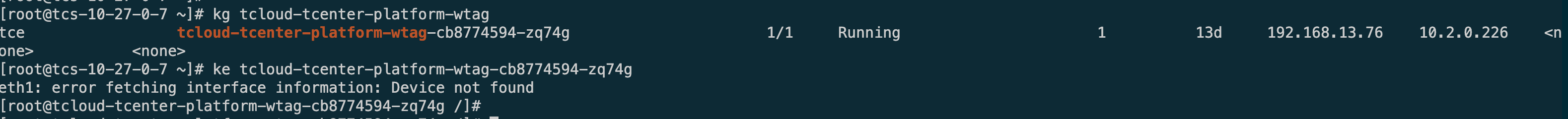
\*\* 这里的region名需要换成客户环境的

标签服务注册

1. 查询cmq的svcip。



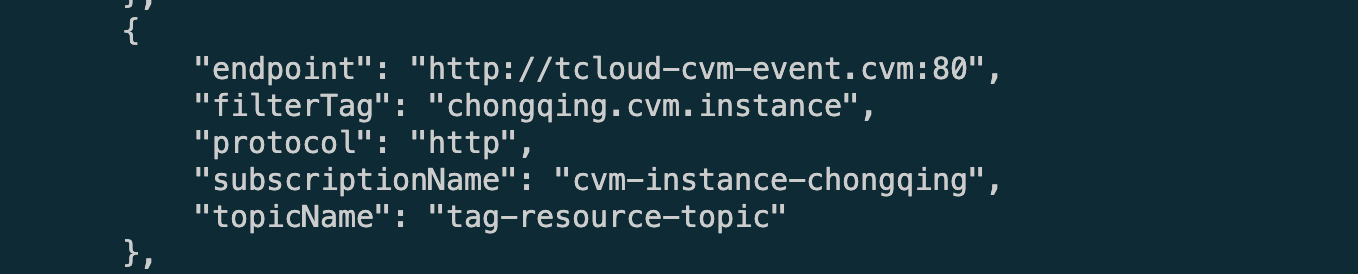
1. 登录pod tcloud-tcenter-platform-wtag。



1. 查询cvm是否注册。

curl http://$cmq-svc-ip:5672/api/v1//subs/tag-resource-topic | python -m json.tool

如果有如下注册的信息，则不需要注册



1. 注册。

curl http://tcloud-tcenter-platform-wtag.tcenter:50031 -d '{"version":"v1.0","componentName":"MC","eventId":1828660475,"timestamp":1449455214,"interface":{"interfaceName":"qcloud.topic.subscribeTopic","para":{"serviceType":"cvm","region":"XXXX","resourcePrefix":"instance","endpoint":"http://tcloud-cvm-event.cvm:80"}}}'