# SFVM-MC管理系统功能节点部署指南

## 基本框架

## 

## 服务器信息

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| **Haproxy** |  |  |  |  |  |  |  |
| **IP** | **CPU** | **MEM（GB）** | **DISK（GB）** | **HOSTNAME** | **OS版本** | **VIP** | **HOST** |
| \*\*\*.\*\*\*.\*\*\*.105 | 2 | 4 | 50 | haproxy01 | Centos7.2 |  | kvmhost01 |
| \*\*\*.\*\*\*.\*\*\*.106 | 2 | 4 | 50 | haproxy02 | Centos7.2 |  | kvmhost02 |
|  |  |  |  |  |  |  |  |
| **Control** |  |  |  |  |  |  |  |
| **IP** | **CPU** | **MEM（GB）** | **DISK（GB）** | **HOSTNAME** | **OS版本** | **HOST** | **用途** |
| \*\*\*.\*\*\*.\*\*\*.101 | 2 | 4 | 50 | console1 | Centos7.2 |  | HOST、VM管理操作 |
| \*\*\*.\*\*\*.\*\*\*.102 | 2 | 4 | 50 | console2 | Centos7.2 |  | HOST、VM管理操作 |
| \*\*\*.\*\*\*.\*\*\*.103 | 2 | 4 | 50 | console3 | Centos7.2 |  | HOST、VM状态获取 |
| \*\*\*.\*\*\*.\*\*\*.104 | 2 | 4 | 50 | console4 | Centos7.2 |  | HOST、VM状态获取 |
|  |  |  |  |  |  |  |  |
| **Image** |  |  |  |  |  |  |  |
| **IP** | **CPU** | **MEM（GB）** | **DISK（GB）** | **HOSTNAME** | **OS版本** | **HOST** |  |
| \*\*\*.\*\*\*.\*\*\*.108 | 2 | 4 | 100 | image01 | Centos7.2 |  |  |
| \*\*\*.\*\*\*.\*\*\*.109 | 2 | 4 | 100 | image02 | Centos7.2 |  |  |
|  |  |  |  |  |  |  |  |
| **Kafka** |  |  |  |  |  |  |  |
| **类型** | **名称** | **校验码** |  |  |  |  |  |
| 主题 | ESG\_CLOUD\_PORTAL |  |  |  |  |  |  |
| 消费者 | ESG\_CLOUD\_PORTAL\_CONSUMER | |  |  |  |  |  |
| 集群 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| ZOOKEEPER |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **DB** |  |  |  |  |  |  |  |
| **域名** | **IP** | **端口** | **类型** | **账号** | **连接串** |  |  |
| \*\*\*.\*\*\*.\*\*\*.\*\*\* | \*\*\*.\*\*\*.\*\*\*.\*\*\* | 3306 | Mysql | name/psword | \*\*\*.\*\*\*.\*\*\*.\*\*\*:3306/name | |  |
| DB跳板机 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Image Cache** |  |  |  |  |  |  |  |
| **IP** | **CPU** | **MEM（GB）** | **DISK（GB）** | **HOSTNAME** | **OS版本** | **用途** | **HOST** |
|  | 4 | 4 | 200 | imagecache01 | Centos6.8 |  |  |
|  | 4 | 4 | 200 | imagecache02 | Centos6.8 |  |  |
|  |  |  |  |  |  |  |  |

## 部署步骤

|  |
| --- |
| 1、申请专用物理机，配置不限，做Raid5，本地盘可用空间3TB。交换机端口开Trunk，做主备模式Bond,交换机需要放通所需要的VLAN |
| 2、创建VM，此VM虚拟网卡必须挂在主机标准化时创建的桥br\_bond0.X 上，安装Kafka |
| 3、创建VM，此VM虚拟网卡必须挂在主机标准化时创建的桥br\_bond0.X 上，安装mariadb，再初始化数据库 |
| 4、自行创建VM: VM虚拟网卡必须挂在主机标准化时创建的桥br\_bond0.X 上 haproxy2台，配置：2Core、2GB； Control4台，配置：4Core4GB； 镜像2台，配置：2Core、2GB、200GB； Cache：每个网络区域2台，配置：2Core、2GB、200GB。 |
| 5、配置Haproxy： |
| 6、配置Control 节点： |
| 7、配置镜像服务器： |
| 8、配置Cache： |
| 9、发布程序包，更新镜像 |

## 物理机操作指引

#### 配置前检查

##### 机器硬件配置检查

###### 检查CPU是否打开超线程(HT)及BIOS虚拟化支持是否

1）检查机器有多少个物理CPU：

# cat /proc/cpuinfo | grep "physical id" | sort -u

physical id : 0

physical id : 1

physical id : 2

physical id : 3

2）检查每个物理CPU包含多少个核心：

# cat /proc/cpuinfo | grep "cpu cores" | sort -u

cpu cores : 10

3）检查系统中总共有多少个逻辑CPU：

# cat /proc/cpuinfo | grep "processor" | wc -l

80

4）计划总共的CPU核心数量，以及是否开了超线程(HT)：

a.机器有4个物理CPU，每个物理CPU有10个核心，因此机器总共的CPU核心数量为：10 \* 4 = 40

b.机器的逻辑CPU为80个，而CPU核心为40个，即：80(逻辑CPU数量) = 40(CPU核心数量) \* 2，因此机器是开了超线程(HT)的。

如果：逻辑CPU数量 = CPU核心数量，则机器没有打开超线程(HT)，需要在BIOS中进行打开并重启机器。

5) 检查系统是否有打开虚拟化支持

# grep -E '(svm|vmx)' /proc/cpuinfo

检查是否有输出，svm为AMD的虚拟化技术，vmx为Intel的虚拟化技术

###### 检查内存大小

# free -g

total used free shared buffers cached

Mem: 504 123 380 0 1 113

-/+ buffers/cache: 8 495

Swap: 19 0 19

如上显示为504GB，由于进率的误差，实际内存大小为512GB。

###### 网卡连接状态检查

**A．检查方法：**

# nmcli device status

DEVICE TYPE STATE CONNECTION

em1 ethernet connected em1

em2 ethernet disconnected --

em3 ethernet unavailable --

em4 ethernet unavailable --

lo loopback unmanaged --

注：需要安装net-tools包才能使用ifconfig命令

**B．检查结果分析及处理：**

1）验证已连接网卡的输出信息中是否正确：

# ethtool em1

速率是否为千兆"Speed: 1000Mb/s"

是否是全双工"Duplex: Full"

链路检测是否为"Link detected: yes"

###### 内存、硬盘配置要求

内存和硬盘大小可参照当前vmware环境硬件配置

默认采用RAID5，如对读写性能要求较高可采用RAID10

#### 基本网络配置

##### 网口初始配置

###### 检查是否有非标准的网口名称并做修改

注：DELL服务器在RHEL7中的网口名称可能是em\*，而不是eth\*，需要统一修改成eth\*名称，以方便管理。如网卡名称已标准化为eth\*，请跳过这部分的配置

1）检查是否有非标准的网口名称：

# ip a |grep mtu

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN

2: eno1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER\_UP> mtu 1500 qdisc mq master bond1 state UP qlen 1000

3: ens3f0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN qlen 1000

4: eno2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN qlen 1000

2）如果有非标准的网口名称，则按如下方法进行修改：

注：使用自动搭建安装的OS不需要做如下操作，如果没有非标准的网口名称则跳过下面的步骤！

A. 修改Kernel启动参数：

# vim /etc/default/grub

……

对当前运行的GRUB\_CMDLINE\_LINUX的行进行修改，添加后面红色的部分：

GRUB\_CMDLINE\_LINUX="crashkernel=auto rd.lvm.lv=VolGroup00/LVroot rd.lvm.lv=VolGroup00/LVswap rhgb quiet biosdevname=0 net.ifnames=0"

B. 重新生成grub配置：

# grub2-mkconfig -o /boot/grub2/grub.cfg

C. 移除网口配置文件（除ifcfg-lo和ifcfg-usb0外）：

# cp -a /etc/sysconfig/network-scripts/ifcfg-lo\* /tmp/

# cp -a /etc/sysconfig/network-scripts/ifcfg-usb\* /tmp/

# rm -rf /etc/sysconfig/network-scripts/ifcfg-\*

# cp -a /tmp/ifcfg-lo\* /etc/sysconfig/network-scripts/

# cp -a /tmp/ifcfg-usb\* /etc/sysconfig/network-scripts/

D. 重新生成标准的网口配置文件：

# i=0;for HWADDR in ` ip a|grep ether|awk '{print $2}' `

do

ETH="eth${i}"

cat << EOF > /etc/sysconfig/network-scripts/ifcfg-$ETH

DEVICE=$ETH

HWADDR=$HWADDR

ONBOOT=no

EOF

i=`expr ${i} + 1`

done

E. 重启机器

# shutdown -r now

F. 重启后验证网口名称是否修改成功：

# ip a

##### 网络配置

###### 配置主机名、DNS

主网采用bond0配置，包含bonding、IP、网关、主机名、DNS、Hosts文件的配置；

1）配置主机名及网关：

# hostnamectl <hostname>

# vi /etc/sysconfig/network

NETWORKING=yes

GATEWAY=<gateway>

2）配置DNS客户端：

# vi /etc/NetworkManager/NetworkManager.conf

在 “main” 的段落加入:

dns=none

# systemctl stop NetworkManager.service

# 配置DNS服务器地址

3）配置hosts文件：

# vi /etc/hosts

127.0.0.1 localhost.localdomain localhost

<主网IP> <hostname>

###### 配置HOST管理网

1）添加bonding模块：

# modprobe --first-time bonding

2）配置ifcfg-bond0：

# vi /etc/sysconfig/network-scripts/ifcfg-bond0

DEVICE=bond0

TYPE=Ethernet

NAME=bond0

BONDING\_MASTER=yes

BOOTPROTO=none

ONBOOT=yes

BONDING\_OPTS="miimon=100 mode=1"

3）配置主网的第一块网卡：

# vi /etc/sysconfig/network-scripts/ifcfg-<ethX>

DEVICE=<ethX>

HWADDR=<mac\_addr>

TYPE=Ethernet

BOOTPROTO=none

ONBOOT=yes

MASTER=bond0

SLAVE=yes

4）配置主网的第二块网卡：

# vi /etc/sysconfig/network-scripts/ifcfg-<ethY>

DEVICE=<ethX>

HWADDR=<mac\_addr>

TYPE=Ethernet

BOOTPROTO=none

ONBOOT=yes

MASTER=bond0

SLAVE=yes

5）配置bond0.X子网卡

其中X代表HOST管理网的vlan id

# vim /etc/sysconfig/network-scripts/ifcfg-bond0.X

DEVICE=bond0.X

BOOTPROTO=none

ONBOOT=yes

VLAN=yes

IPADDR=<IPADDR>

NETMASK=<NETMASK>

BRIDGE=br\_bond0.X

重启网络，验证IP连通性

#systemctl restart network

6) 配置br\_bond0.X网桥

删除bond.X的配置文件中IPADDR和NETMASK对应的行

创建网桥文件,并将IP填写到此处

# vim /etc/sysconfig/network-scripts/ifcfg-br\_bond0.X

DEVICE=br\_bond0.X

TYPE=Bridge

BOOTPROTO=static

IPADDR=<IPADDR>

NETMASK=<NETMASK>

ONBOOT=yes

重启网络，验证IP连通性

#systemctl restart network

###### 做bond0的网卡切换验证

在bond0中分别切换到不同的网卡做ping网关的连通性验证

1）查看bond0中的网卡及状态：

# cat /proc/net/bonding/bond0 | egrep "Active Slave:|Slave Interface:"

Currently Active Slave: eth10

Slave Interface: eth10

Slave Interface: eth14

2）切换到第1块网卡上进行测试：

# ifenslave -c bond0 eth10 （由于原来的active网卡就是eth10，该命令可以省略）

# cat /proc/net/bonding/bond0 | grep "Active Slave:"

Currently Active Slave: eth10

# ping 10.8.6.200

PING 10.8.6.200 (10.8.6.200) 56(84) bytes of data.

…前面32个包可能会丢失…

64 bytes from 10.8.6.200: icmp\_seq=33 ttl=255 time=1.09 ms

64 bytes from 10.8.6.200: icmp\_seq=34 ttl=255 time=1.33 ms

……

3）切换到第2块网卡上进行测试：

# ifenslave -c bond0 eth14

# cat /proc/net/bonding/bond0 | grep "Active Slave:"

Currently Active Slave: eth14

# ping 10.8.6.200

PING 10.8.6.200 (10.8.6.200) 56(84) bytes of data.

…前面32个包可能会丢失…

64 bytes from 10.8.6.200: icmp\_seq=33 ttl=255 time=1.09 ms

64 bytes from 10.8.6.200: icmp\_seq=34 ttl=255 time=1.33 ms

……

注：由于系统需要进行ARP学习的原因，在进行ping测试时，前面的32个包可能会丢失。

#### 系统其它配置

##### 内核版本调整及软件包安装

###### 调整内核参数

1）添加用户进程与打开文件限制

#vi /etc/security/limits.conf

添加或修改成以下数值

\* soft nproc 524288

\* hard nproc 524288

\* soft nofile 524288

\* hard nofile 524288

2）将 /etc/security/limits.d/20-nproc.conf文件中的以下行改掉

\* soft nproc 1024

修改为

\* soft nproc 524288

3）

#vi /etc/sysctl.d/sys-sysctl.conf

加入以下行

kernel.pid\_max = 524288

fs.file-max = 6553600 （使用sysctl -a查看此参数，少于6553600时需要修改）

*注：*

在RHEL7/CentOS7上，由init 1号进程启动的子进程初始max open files为65536，建议应用程序的启动脚本里添加更改ulimit的命令 # ulimit -n <value>

# cat /proc/1/limits ...

Max open files 65536 65536 files

###### OS版本及yum源

OS版本为Centos7.3，yum源文件配置如下：

[CENTOS7.3-basic]

name=CENTOS7.3 basic repository

baseurl=http://yumserver/centos7/U3/$basearch/OS

enabled=1

gpgcheck=0

gpgkey=http:// yumserver /centos7/U3/$basearch/OS/RPM-GPG-KEY-redhat-release

[CENTOS7-others]

name=CENTOS7 update packages

baseurl=http:// yumserver /centos7/others/$basearch

enabled=1

gpgcheck=0

[CENTOS7-epel]

name=CENTOS7 epel resource

baseurl=http:// yumserver /epel7Server/$basearch

enabled=1

gpgcheck=0

###### 安装工具软件包

1）安装工具软件包：

# yum install iftop

# yum install iotop

# yum install strace ftp telnet

# yum install net-snmp

# yum install sg3\_utils

# yum install tuned-utils-systemtap kernel-debuginfo

# yum install powertop

# yum update openssl

默认不安装httpd，如果后续安装httpd，请执行如下命令更新版本

# yum update httpd

2）修改sysstat配置：

a.修改历史数据收集频率为1分钟：

# vi /etc/cron.d/sysstat

# run system activity accounting tool every 10 minutes

\*/1 \* \* \* \* root /usr/lib64/sa/sa1 1 1

# generate a daily summary of process accounting at 23:53

53 23 \* \* \* root /usr/lib64/sa/sa2 -A

b.重新加载配置以生效：

# systemctl restart sysstat

# cat /etc/cron.d/sysstat

3) 停止并禁用NetworkManager服务

# systemctl stop NetworkManager

# systemctl disable NetworkManager

##### 系统管理用户创建

###### 建立组usr01

# groupadd -g 600 usr01

###### 创建cloudlog账户

# useradd –g usr01 –d /home/cloudlog cloudlog

# passwd cloudlog

##### Lvm及挂载点配置

###### Lvm配置

Lvm配置标准可参考如下，所有空间都划分给LVapp

[root@XXXXXXX ~]# lvs

LV VG Attr LSize Pool Origin Data% Meta% Move Log Cpy%Sync Convert

LVapp VolGroup00 -wi-ao---- 16.20t

LVhome VolGroup00 -wi-ao---- 30.00g

LVkdump VolGroup00 -wi-ao---- 50.00g

LVroot VolGroup00 -wi-ao---- 30.00g

LVswap VolGroup00 -wi-ao---- 20.00g

LVtmp VolGroup00 -wi-ao---- 20.00g

LVvar VolGroup00 -wi-ao---- 20.00g

###### 挂载点配置

/dev/mapper/VolGroup00-LVroot / xfs defaults 0 0

/dev/mapper/VolGroup00-LVhome /home xfs defaults 0 0

/dev/mapper/VolGroup00-LVkdump /kdump xfs defaults 0 0

/dev/mapper/VolGroup00-LVtmp /tmp xfs defaults 0 0

/dev/mapper/VolGroup00-LVvar /var xfs defaults 0 0

/dev/mapper/VolGroup00-LVswap swap swap defaults 0 0

/dev/mapper/VolGroup00-LVapp /app xfs defaults 0 0

###### 卷配置

# mkdir /app/image

##### 虚拟化相关软件安装

#此部分为第一次安装环境需要，后续可依靠平台的一键加入host功能自动实现

###### 虚拟化账户创建

#useradd –g user01 cloudkvm –u 2000 –d /home/cloud

###### 虚拟化软件安装

# yum install –y cyrus-sasl-md5 cyrus-sasl-devel libguestfs-tools qemu-kvm.x86\_64 qemu.x86\_64 qemu-kvm-tools libvirt.x86\_64 libvirt-python.x86\_64 qemu-kvm-common qemu-img

###### Libvirt相关配置文件修改

# sed -i 's/#LIBVIRTD\_ARGS/LIBVIRTD\_ARGS/g' /etc/sysconfig/libvirtd

###### Libvirt配置文件替换

# rm -rf /etc/libvirt/libvirtd.conf

# rm -rf /etc/sasl2/libvirt.conf

用当前文件夹下同名文件进行替换

###### 系统参数修改

# echo "DefaultLimitCORE=infinity" >> /etc/systemd/system.conf

# echo "DefaultLimitNOFILE=524288" >> /etc/systemd/system.conf

# echo "DefaultLimitNPROC=524288" >> /etc/systemd/system.conf

# echo "DefaultLimitCORE=infinity" >> /etc/systemd/user.conf

# echo "DefaultLimitNOFILE=524288" >> /etc/systemd/user.conf

# echo "DefaultLimitNPROC=524288" >> /etc/systemd/user.conf

# echo "options nf\_conntrack hashsize=131072" >> /etc/modprobe.d/nf\_conntrack.conf

###### 创建sasl用户并配置密码

创建如下脚本文件

#vi create\_sasl\_user.sh

#!/bin/bash  
user\_pass=$1  
root\_pass=$2  
(echo $user\_pass;sleep 1;echo $user\_pass)|/sbin/saslpasswd2 -a libvirt cloudkvm  
echo $user\_pass | /bin/passwd --stdin cloudkvm  
echo $root\_pass | /bin/passwd --stdin root

运行脚本

# /bin/bash /root/create\_sasl\_user.sh $libvirt\_pass $root\_pass

其中libvirt\_pass 和 root\_pass为libvirt账户的密码和root账户的密码

## Haproxy操作指引

|  |
| --- |
| 1. 在Haproxy01、02上面操作，创建用户 #useradd -u 800 opr #useradd -u 801 user #usermod -a -G opr user |
| 1. 在Haproxy01、02上面操作，创建目录 #chown opr.opr /app  #mkdir /app/haproxy #chown opr.opr /app/haproxy  #mkdir /app/tmp #chown opr.opr /app/tmp |
| 1. 在Haproxy01、02上面操作，配置Sudo #visudo opr ALL=(ALL) NOPASSWD: /usr/sbin/tcpdump,/bin/netstat opr ALL=(ALL) NOPASSWD: /usr/cat /var/log/\*  opr ALL=(ALL) NOPASSWD: /etc/init.d/keepalived opr ALL=(ALL) NOPASSWD: /usr/bin/vim /etc/keepalived/\* opr ALL=(ALL) NOPASSWD: /usr/bin/vim /etc/init.d/keepalived  opr ALL=(ALL) NOPASSWD: /app/haproxy/sbin/haproxy,/app/haproxy/stopHaproxy.sh,/app/haproxy/startHaproxy.sh  opr ALL=(ALL) NOPASSWD: /usr/bin/sysauto\_CLOUD mysql ALL=(ALL) NOPASSWD: /usr/bin/sysauto\_CLOUD |
| 1. 在Haproxy01、02上面操作，配置内核参数 #vim /etc/sysctl.conf kernel.shmmni = 4096 kernel.sem = 500 256000 250 8192 net.ipv4.ip\_local\_port\_range = 10000 65000 net.ipv4.tcp\_tw\_reuse = 1 net.ipv4.tcp\_tw\_recycle = 0 net.ipv4.tcp\_max\_tw\_buckets = 10000 net.ipv4.tcp\_fin\_timeout = 30 net.core.rmem\_default = 1048576 net.core.rmem\_max = 1048576 net.core.wmem\_default = 262144 net.core.wmem\_max = 262144  net.ipv4.tcp\_keepalive\_time = 1800 net.ipv4.tcp\_retries2 = 5 net.core.somaxconn = 1024 |
| 1. 在Haproxy01、02上面操作，安装Haproxy yum install -y zlib.x86\_64 zlib-devel.x86\_64 openssl-devel.x86\_64 openssl.x86\_64 pcre-devel.x86\_64 pcre.x86\_64 haproxy |
| 1. 在Haproxy01、02上面操作，使用opr账号操作，创建启停脚本 $cd /app/haproxy/  $vim startHaproxy.sh  #! /bin/bash # start haporxy # by Lijianfang  . /etc/init.d/functions  num=`ps -C haproxy --no-header |wc -l` if [ $num -eq 0 ]; then  /sbin/haproxy -f /etc/haproxy/haproxy.cfg  if [ `ps -C haproxy --no-header |wc -l` -eq 0 ];then  exit 1  fi  exit 0 fi $vim stopHaproxy.sh  #! /bin/bash . /etc/init.d/functions PID=`cat /var/run/haproxy.pid` kill -9 $PID #chmod 755 \* |
| 1. 在Haproxy01、02上面操作，配置Haproxy [root@haproxy02 haproxy]# cat /etc/haproxy/haproxy.cfg global  log 127.0.0.1 local3 info   chroot /var/lib/haproxy  pidfile /var/run/haproxy.pid  maxconn 32768  user opr  group opr  daemon  nbproc 1   defaults  mode http  log global  option httplog  option dontlognull  option httpclose  option forwardfor  option http-no-delay  option redispatch  retries 3  timeout connect 5000ms  timeout client 5m  timeout server 5m  timeout check 10s  stats uri /haproxy\_stats  stats hide-version  stats auth opr:kvm@748  frontend secure  bind \*:8888  stats enable  stats uri /health\_check  stats auth admin:admin  stats hide-version  stats admin if TRUE  stats refresh 30s  frontend http\_kvm  bind \*:80  acl red\_login path\_beg -i /cloudkvm  #redirect scheme https if !red\_login  redirect scheme https if red\_login   mode http  maxconn 32768  option forwardfor  acl url\_kvm path\_beg -i /cloudkvm  acl default hdr\_beg(host) -i havip  use\_backend http\_kvm if url\_kvm  use\_backend login if default !url\_kvm  default\_backend login  frontend https\_kvm  bind \*:443 ssl crt /etc/ssl/xip.io/xip.io.pem  acl red\_login path\_beg -i /instance/console  redirect scheme http if red\_login   mode http  maxconn 32768  option forwardfor  acl url\_kvm path\_beg -i /cloudkvm  acl default hdr\_beg(host) -i havip  use\_backend http\_kvm if url\_kvm  use\_backend login if default !url\_kvm  default\_backend login   frontend instance\_console  bind \*:6080  maxconn 32768  option forwardfor  acl instance\_console hdr\_beg(host) -i havip  use\_backend instance\_console\_back if instance\_console  default\_backend instance\_console\_back  frontend host\_console  bind \*:4200  maxconn 32768  option forwardfor  acl host\_console hdr\_beg(host) -i havip  use\_backend host\_console\_back if host\_console  default\_backend host\_console\_back  backend http\_kvm  cookie SERVERID insert indirect   hash-type consistent  balance leastconn  option httpchk HEAD cloudkvm/login.html HTTP/1.0\r\n\r\n  server http\_kvm\_01 console03:8080/cloudkvm cookie app1 check inter 2000 rise 3 fall 5  server http\_kvm\_02 console04:8080/cloudkvm cookie app2 check inter 2000 rise 3 fall 5  backend login  cookie SERVERID insert indirect   hash-type consistent  balance leastconn  option httpchk HEAD /cloudkvm/login.html HTTP/1.0\r\n\r\n  server http\_kvm\_01 console03:8080 cookie app1 check inter 2000 rise 3 fall 5  server http\_kvm\_02 console04:8080 cookie app2 check inter 2000 rise 3 fall 5   backend instance\_console\_back  cookie SERVERID insert indirect  hash-type consistent  balance leastconn  server http\_kvm\_01 console03:6080 cookie app1 check inter 2000 rise 3 fall 5  server http\_kvm\_02 console04:6080 cookie app2 check inter 2000 rise 3 fall 5   backend host\_console\_back  cookie SERVERID insert indirect  hash-type consistent  balance leastconn  server http\_kvm\_01 console03:4200 cookie app1 check inter 2000 rise 3 fall 5  server http\_kvm\_02 console04:4200 cookie app2 check inter 2000 rise 3 fall 5 |
| 1. 在Haproxy01、02上面操作，安装keepalived #yum install keepalived #systemctl enable keepalived # vim /etc/keepalived/check\_haproxy.sh  #! /bin/bash num=`ps -C haproxy --no-header |wc -l` if [ $num -eq 0 ]; then  /sbin/haproxy -f /etc/haproxy/haproxy.cfg  sleep 3  if [ `ps -C haproxy --no-header |wc -l` -eq 0 ]; then  /bin/systemctl stop keepalived  fi fi #chmod 755 /etc/keepalived/check\_haproxy.sh |
| 1. 在Haproxy01上面操作，配置keepalived # vim /etc/keepalived/keepalived.conf ! Configuration File for keepalived  vrrp\_script chk\_http\_haproxy {  script "/etc/keepalived/check\_haproxy.sh"  interval 10  weight 2 }  global\_defs {  notification\_email {  itop\_mw@express.com  }  notification\_email\_from itop\_mw@express.com  smtp\_server mail.express.com  smtp\_connect\_timeout 30  router\_id kvm }   vrrp\_instance VI\_1 {  state MASTER  interface eth0  virtual\_router\_id 10  priority 100  advert\_int 1  authentication {  auth\_type PASS  auth\_pass 1111  }   track\_script {  #run the script   chk\_http\_haproxy  }   virtual\_ipaddress {  havip  } } |
| 1. 在Haproxy02上面操作，配置keepalived # vim /etc/keepalived/keepalived.conf ! Configuration File for keepalived  vrrp\_script chk\_http\_haproxy {  script "/etc/keepalived/check\_haproxy.sh"  interval 10  weight 2 }  global\_defs {  notification\_email {  itop\_mw@-express.com  }  notification\_email\_from itop\_mw@-express.com  smtp\_server mail.express.com  smtp\_connect\_timeout 30  router\_id kvm }   vrrp\_instance VI\_1 {  state BACKUP  interface eth0  virtual\_router\_id 10  priority 99  advert\_int 1  authentication {  auth\_type PASS  auth\_pass 1111  }   track\_script {  #run the script   chk\_http\_haproxy  }   virtual\_ipaddress {  havip  } } |
| 1. 配置ssl证书-有效期为365天,在配置证书的步骤中请默认回车就行 在haproxy01上配置如下： cd /etc/ssl mkdir xip.io cd /etc/ssl/xip.io/ openssl genrsa -out /etc/ssl/xip.io/xip.io.key 2048 openssl req -new -key /etc/ssl/xip.io/xip.io.key -out /etc/ssl/xip.io/xip.io.csr openssl x509 -req -days 365 -in /etc/ssl/xip.io/xip.io.csr -signkey /etc/ssl/xip.io/xip.io.key -out /etc/ssl/xip.io/xip.io.crt cat /etc/ssl/xip.io/xip.io.crt /etc/ssl/xip.io/xip.io.key | tee /etc/ssl/xip.io/xip.io.pem 在haproxy02上配置如下： 将haproxy01上生成的xip.io.pem文件拷贝至haproxy02的对应目录下 |
| 1. 在Haproxy01、02上面操作，使用opr账号操作，测试配置是否正常 $cd /app/haproxy/ $sudo ./startHaproxy.sh  $ps -ef |grep haproxy $ sudo ./stopHaproxy.sh  $ps -ef |grep haproxy |
| 1. 在Haproxy01、02上面操作，启动Keepalive，做VIP切换测试 #systemctl start keepalived 若主机备机的keepalived服务都是启动成功状态，关闭主机的keepalived服务则VIP漂移到备机，当主备机keepalived服务都启动状态则VIP漂移到主机。 |

## 控制节点部署指引

1. 创建4台VM做为控制节点，控制节点虚拟网卡必须挂在主机标准化时创建的桥br\_bond0.X 上
2. 4台控制节点需要新增以下内核参数配置并重启控制节点  
    vi /etc/systemd/system.conf   
    设置如下参数  
    DefaultLimitCORE=infinity  
    DefaultLimitNOFILE=524288  
    DefaultLimitNPROC=524288  
     
    vi /etc/systemd/user.conf   
    设置如下参数  
    DefaultLimitCORE=infinity  
    DefaultLimitNOFILE=524288  
    DefaultLimitNPROC=524288  
      
    注意：完成修改后需要重启控制节点才能生效
3. 前提说明  
    1.在/app下新建kvmmgr\_dashboard目录，项目代码放在该目录下  
    mkdir -p /app/kvmmgr\_dashboard  
     
    2.项目的deploy目录，不需创建  
    （1）pip\_cache存放所有需要安装的python依赖包  
    （2）packages存放安装pip\_cache中的包时需要提前安装的依赖包  
    （3）install\_package\_local.sh 批量安装包到本地环境的脚本  
    （4）install\_package.sh 批量安装包到虚拟环境的脚本（暂不用）
4. yum安装相关文件  
    yum install -y gcc.x86\_64 openssl-devel.x86\_64 ipmitool.x86\_64 mysql-devel python-devel libffi-devel.x86\_64 openldap-devel.x86\_64 libvirt-python supervisor ansible sshpass python-websockify virt-v2v python-libguestfs mariadb-server mariadb libjpeg-turbo-1.2.90-5.el7.x86\_64 libjpeg-turbo-devel-1.2.90-5.el7.x86\_64 freetype-devel.x86\_64 openjpeg.x86\_64
5. 启动mariadb  
    [root@console03 deploy]# systemctl start mariadb  
    [root@console03 deploy]# systemctl enable mariadb
6. 拉取平台代码

a.拉取代码  
 略

b.将代码放到/app/kvmmgr\_dashboard目录下，并解压  
 [root@console03 kvmmgr\_dashboard]# unzip kvmmgr\_dashboard.zip  
 将文件夹重命名  
 mv kvmmgr\_dashboard\* kvmmgr\_dashboard

c.修改/app/kvmmgr\_dashboard/kvmmgr\_dashboard/config/default.py文件中各组件IP及用户名密码

1. 安装pip  
    1.切换目录  
    到项目的deploy下的packages目录下  
    [root@console03 ~]# cd /app/kvmmgr\_dashboard/kvmmgr\_dashboard/deploy/packages/  
     
    2.先安装其他依赖包  
    这些包有前后依赖关系，所以请依次安装。  
    安装方法都类似，解压-切换-安装，具体请参照下面2.1  
    2.1 packaging  
    [root@console03 packages]# tar -zxvf pyparsing-2.2.0.tar.gz  
    [root@console03 packages]# cd pyparsing-2.2.0/  
    [root@console03 packaging-16.8]# python setup.py install  
    2.2 packaging  
    2.3 appdirs  
    2.4 setuptools(为zip包，使用unzip命令)  
    3.解压pip安装包  
    [root@console03 packages]# tar -zxvf pip-9.0.1.tar.gz  
    [root@console03 packages]# cd pip-9.0.1/  
     
    4.安装pip  
    [root@console03 pip-9.0.1]# python setup.py install  
     
    5.验证  
    此时执行pip list可查看到已安装的依赖包  
    [root@console03 pip-9.0.1]# pip list
2. 创建平台运行的用户及其环境  
    1.新建cloudkvm、cloudblj用户  
    [root@console03 ~]# useradd -u 2000 -o -g usr01 -d /home/cloudkvm cloudkvm  
    [root@console03 ~]# useradd -g usr01 -d /home/cloudblj cloudblj  
     
    2.授权  
    [root@console03 kvmmgr\_dashboard]# chown -R cloudkvm:usr01 /app/kvmmgr\_dashboard  
     
    3.创建物理机性能收集目录  
    [root@console03 app]# mkdir -p /app/hostinfo  
    [root@console03 app]# chown -R cloudkvm:usr01 /app/hostinfo
3. 执行shell命令安装python依赖包  
    1.切换目录  
    [root@console03 ~]# cd /app/kvmmgr\_dashboard/kvmmgr\_dashboard/deploy/  
     
    2.安装其他依赖包  
    在执行install\_package\_local.sh脚本之前，需要安装其他依赖包，包都在deploy/packages目录下  
    由于这些包有前后依赖关系，所以请依次安装  
    2.1 Werkzeug  
    2.2 MarkupSafe  
    2.3 Jinja2  
    2.4 itsdangerous  
    2.5 poster  
    2.6 pyasn1  
    2.7 pycparser  
    2.8 cffi  
    2.9 ipaddress  
    2.10 enum34  
    2.11 asn1crypto  
    2.12 idna  
    2.13 cryptography  
    2.14 paramiko  
    2.15 tabulate  
    2.16 greenlet  
    2.17 gevent(安装有warning，请忽略)  
    2.18 Imaging  
    2.19 Flask-HTTPAuth  
    2.20 pyvmomi  
    2.21 lml  
    2.22 xlrd  
    2.23 xlwt  
    2.24 pyexcel-io  
    2.25 pyexcel-xls  
    2.26 pbr-3.1.1  
    2.27 lockfile-0.12.2

2.28 paginator  
 3.执行脚本  
 [root@console03 deploy]# sh install\_package\_local.sh  
 需要保证没有错误和异常报出来

1. supervisor管理程序配置  
    a.首先确定已经安装supervisor包  
    [root@console03 app]# rpm -aq | grep supervisor  
    supervisor-3.0-1.el7.noarch  
    b.修改配置文件，在最后加入以下配置  
    [root@console03 app]# vi /etc/supervisord.conf  
    \*\*\*在文件最后加入以下配置\*\*\*  
      
    控制节点console1,2的supervisord配置文件新增以下配置  
   ############################################################  
   [program:kvmmgr\_dashboard\_get\_host\_status]  
   command=/bin/python main.py  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/collect\_data/host  
   stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/get\_host\_status.log  
   redirect\_stderr=true  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_get\_instances\_status]  
   command=/bin/python main.py  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/collect\_data/instances  
   stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/get\_instances\_status.log  
   redirect\_stderr=true  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   控制节点console3,4的supervisord配置文件新增以下配置  
   ############################################################  
   [program:kvmmgr\_dashboard]  
   command=/bin/python main.py api\_web\_prd2 start  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/bin  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_instance\_console]  
   command=/bin/python vnc\_console.py --debug  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/lib/vnc\_console  
   autostart=true  
   autorestart=true  
   stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/vnc-console.log  
   redirect\_stderr=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_v2v\_openstack\_worker]  
   command=/bin/python v2v\_openstack\_do.py  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/controller/web\_api/v2v  
   stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/v2v\_openstack\_worker.log  
   redirect\_stderr=true  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_v2v\_esx\_worker]  
   command=/bin/python v2v\_esx\_do.py  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/controller/web\_api/v2v  
   stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/v2v\_esx\_worker.log  
   redirect\_stderr=true  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_kafka\_consumer]  
   command=/bin/python main\_consumer\_kafka.py start consumer1  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/bin  
   autostart=true  
   autorestart=true  
   user=cloudkvm  
     
   [program:kvmmgr\_dashboard\_requests\_worker]  
   command=/bin/python trace\_requests.py  
   directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/collect\_data/requests\_status  
   autostart=true  
   autorestart=true  
   user=cloudkvm

[program:kvmmgr\_dashboard\_image\_sync\_worker]

command=/bin/python image\_sync\_do.py

directory=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/controller/web\_api/image\_sync

stdout\_logfile=/app/kvmmgr\_dashboard/kvmmgr\_dashboard/log/image\_sync\_worker.log

autostart=true

autorestart=true

user=cloudkvm  
 c.创建log目录  
 mkdir -p /app/kvmmgr\_dashboard/kvmmgr\_dashboard/log  
 chown -R cloudkvm:usr01 /app/kvmmgr\_dashboard/kvmmgr\_dashboard/log

1. ansible配置文件  
    a.首先确定已经安装ansible包  
    [root@console03 app]# rpm -aq | grep ansible  
    ansible-1.9.1-1.el7.noarch  
    b.修改ansible配置文件  
    [root@console03 app]# vi /etc/ansible/ansible.cfg  
    找到以下三个配置项并把前面'#'号去掉以打开配置  
    ask\_sudo\_pass = True  
    ask\_pass = True  
    host\_key\_checking = False
2. 数据库初始化  
    DB部分操作用sql导入
3. shellinabox程序配置（只需要配置控制节点3、4）  
    a.确定上面已经新建cloudblj用户  
    b.将目录/kvmmgr\_dashboard/deploy/host\_terminal中所有文件拷贝到目录/home/cloudblj中  
    [root@console03 app]# cp /app/kvmmgr\_dashboard/kvmmgr\_dashboard/deploy/host\_terminal/\* /home/cloudblj  
    c.修改文件权限  
    [root@console03 app]# cd /home/cloudblj  
    [root@console03 app]# chown cloudblj:usr01 outipmi.sh  
    [root@console03 app]# chown cloudblj:usr01 runipmi.sh  
    [root@console03 app]# chown cloudblj:usr01 ssh\_login.py  
    [root@console03 app]# chmod 755 runipmi.sh  
    [root@console03 app]# chmod 755 outipmi.sh  
    d.安装webssh\_shellinabox.tgz  
    [root@console03 cloudblj]# cd /home/cloudblj  
    [root@console03 cloudblj]# tar -zxvf webssh\_shellinabox.tgz  
    [root@console03 cloudblj]# cd webssh\_shellinabox  
    [root@console03 webssh\_shellinabox]# ./configure  
    [root@console03 webssh\_shellinabox]# make;make install  
      
    e.修改cloudblj用户.bashrc、.barh\_logout两个文件配置  
    [root@console03 cloudblj]#vi /home/cloudblj/.bashrc   
    # 在文件最后加入以下配置  
    ./runipmi.sh  
    exit  
      
    [root@console03 cloudblj]#vi /home/cloudblj/.bash\_logout   
    # 在文件最后加入以下配置  
    ./outipmi.sh  
      
      
    f.设置公钥认证  
    [root@console03 cloudblj]# su - cloudblj  
    [cloudblj@console03 cloudblj]# ssh-keygen -t rsa -f ~/.ssh/id\_rsa -P ""  
    [cloudblj@console03 cloudblj]# cd /home/cloudblj/.ssh  
    [cloudblj@console03 cloudblj]# cp id\_rsa.pub authorized\_keys  
      
    g.启动shellinabox  
    [root@console03 cloudblj]# cd /usr/local/share/doc/shellinabox  
    [root@console03 shellinabox]# /usr/local/bin/shellinaboxd -t -b -s /:cloudblj:usr01:/home/cloudblj:'/bin/python /home/cloudblj/ssh\_login.py'

## 数据库初始化

### access

CREATE TABLE `access` ( `role\_id` int(11) NOT NULL COMMENT '关联角色表ID', `group\_id` int(11) NOT NULL COMMENT '关联组表ID', `area\_id` int(11) NOT NULL COMMENT '关联区域ID', KEY `access\_group\_id\_area\_id` (`group\_id`,`area\_id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='权限表'

### area

CREATE TABLE `area` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `parent\_id` int(11) DEFAULT NULL COMMENT '自关联id，表示父区域ID',  
  `name` varchar(50) DEFAULT NULL COMMENT '名称',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '显示名称',  
  `area\_type` enum('0','1') DEFAULT '0' COMMENT '区域类型：0表示总部，1表示地区',  
  `manager` varchar(50) DEFAULT NULL COMMENT '区域管理员',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=143 DEFAULT CHARSET=utf8 AVG\_ROW\_LENGTH=1 CHECKSUM=1 DELAY\_KEY\_WRITE=1 ROW\_FORMAT=DYNAMIC COMMENT='区域表'

### datacenter

CREATE TABLE `datacenter` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT '机房名称',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '中文显示名',  
  `dc\_type` enum('0','1','2','3','4','5','6','7','8','9') DEFAULT NULL COMMENT '机房类型：0:其他、1:测试SIT、2:准生产STG、3:研发DEV、4:生产PRD、5:容灾DR、6:微应用DR、8''容灾压测PST、''9''测试压测IST',  
  `area\_id` int(11) DEFAULT NULL COMMENT '关联区域ID',  
  `province` varchar(4) DEFAULT NULL COMMENT '所在省',  
  `address` varchar(50) DEFAULT NULL COMMENT '地址',  
  `description` varchar(255) DEFAULT NULL COMMENT '机房描述',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除。0表示没有删除，1表示已经删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`),  
  KEY `datacenter\_area\_id\_id\_isdeleted` (`area\_id`,`id`,`isdeleted`)  
) ENGINE=InnoDB AUTO\_INCREMENT=239 DEFAULT CHARSET=utf8 MIN\_ROWS=1 AVG\_ROW\_LENGTH=1 CHECKSUM=1 DELAY\_KEY\_WRITE=1 ROW\_FORMAT=DYNAMIC COMMENT='机房表'

### feedback\_info

CREATE TABLE `feedback\_info` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `user\_id` varchar(20) NOT NULL COMMENT '关联用户工号',  
  `problem\_category` varchar(60) NOT NULL COMMENT '问题类型',  
  `problem\_id` varchar(20) NOT NULL COMMENT '关联问题分类ID',  
  `problem\_description` varchar(255) DEFAULT NULL COMMENT '问题描述',  
  `network\_address` varchar(100) DEFAULT NULL COMMENT '出现问题的相关页面',  
  `submit\_time` datetime DEFAULT NULL COMMENT '反馈时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=93 DEFAULT CHARSET=utf8

### flavor

CREATE TABLE `flavor` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `name` varchar(20) DEFAULT NULL COMMENT '规格的名称',  
  `vcpu` int(11) DEFAULT NULL COMMENT 'VCPU个数',  
  `memory\_mb` int(11) DEFAULT NULL COMMENT '内存大小MB',  
  `root\_disk\_gb` int(11) DEFAULT NULL COMMENT '根分区大小GB',  
  `isenable` enum('1','0') DEFAULT '1' COMMENT '是否启用，1表示启用，0表示未启用',  
  `isdeleted` enum('0','1') DEFAULT '0' COMMENT '是否删除，1表示已经删除，0表示未删除',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=493 DEFAULT CHARSET=utf8 COMMENT='规格表'

### host

CREATE TABLE `host` ( `id` int(11) NOT NULL AUTO\_INCREMENT, `name` varchar(50) DEFAULT NULL COMMENT '物理机主机名', `displayname` varchar(50) DEFAULT NULL COMMENT '中文名称或描述', `sn` varchar(20) DEFAULT NULL COMMENT '序列号', `ipaddress` varchar(50) DEFAULT NULL COMMENT 'IP地址', `manage\_ip` varchar(50) DEFAULT NULL COMMENT '管理IP', `status` enum('99','1','0') DEFAULT '0' COMMENT '物理机稳定的状态：运行中0，关机1，错误99', `typestatus` enum('2','1','0') DEFAULT '0' COMMENT '业务状态：正常0，锁定1，维护2', `libvirtd\_status` enum('0','1') DEFAULT '0' COMMENT 'libvirt状态，正常0，异常1', `cpu` int(11) DEFAULT '0' COMMENT '逻辑cpu 核数(逻辑core个数)', `hostpool\_id` int(11) DEFAULT NULL COMMENT '关联hostpool 表', `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除：0表示未删除', `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间', `created\_at` datetime DEFAULT NULL COMMENT '创建时间', `updated\_at` datetime DEFAULT NULL COMMENT '更新时间', `memory\_mb` int(11) DEFAULT '0' COMMENT '内存剩余大小MB（排除了保留）', `ostype` varchar(50) DEFAULT NULL COMMENT '物理机操作系统版本', `app\_disk\_mb` int(11) DEFAULT '0' COMMENT '虚拟机所在盘的剩余大小', `hold\_mem\_gb` int(11) DEFAULT '0' COMMENT '保留内存', `instances\_collect\_time` datetime DEFAULT NULL COMMENT '物理机下所有虚拟机的上次收集时间', `host\_collect\_time` datetime DEFAULT NULL COMMENT '物理机状态的上次收集时间', `host\_performance\_collect\_time` datetime DEFAULT NULL COMMENT '物理机性能数据上一次收集时间', `host\_clone\_status` enum('0','1') DEFAULT NULL COMMENT '物理机当前是否有克隆任务;''0''代表没有''1''代表有', `libvirt\_port` varchar(50) DEFAULT NULL COMMENT 'libvirt服务端口', `libvirt\_status` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前libvirt状态', `start\_time` varchar(50) DEFAULT NULL COMMENT '开机时间', `images` varchar(2000) DEFAULT NULL COMMENT '物理机上所有镜像', `cpu\_core` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机cpu', `mem\_size` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机内存', `disk\_size` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机磁盘', `net\_size` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机网速', `current\_cpu\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前cpu使用率', `current\_mem\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前内存使用率', `current\_disk\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前磁盘使用率', `week\_cpu\_p95\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机一周cpu p95', `week\_mem\_p95\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机一周内存 p95', `current\_net\_rx\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前上行带宽', `current\_net\_tx\_used` varchar(50) DEFAULT NULL COMMENT '性能数据：物理机当前下行带宽', PRIMARY KEY (`id`), KEY `host\_hostpool\_id\_id\_ipaddress\_isdeleted` (`hostpool\_id`,`id`,`ipaddress`,`isdeleted`)) ENGINE=InnoDB AUTO\_INCREMENT=119 DEFAULT CHARSET=utf8 AVG\_ROW\_LENGTH=1 CHECKSUM=1 DELAY\_KEY\_WRITE=1 ROW\_FORMAT=DYNAMIC COMMENT='物理机表'

### hostpool

CREATE TABLE `hostpool` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT 'hostpool的名称',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '显示名称',  
  `least\_host\_num` int(11) DEFAULT NULL COMMENT '集群主机数目下限',  
  `hostpool\_type` enum('1','0') DEFAULT '0' COMMENT '''0''共享类型，''1''独占类型',  
  `app\_code` varchar(100) DEFAULT NULL COMMENT '特殊集群相关的项目编码',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除： 0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `net\_area\_id` int(11) DEFAULT '1' COMMENT '关联网络区域表',  
  PRIMARY KEY (`id`),  
  KEY `hostpool\_net\_area\_id\_displayname\_id\_isdeleted` (`net\_area\_id`,`displayname`,`id`,`isdeleted`)  
) ENGINE=InnoDB AUTO\_INCREMENT=140 DEFAULT CHARSET=utf8 COMMENT='物理机池表'

### image

CREATE TABLE `image` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `displayname` varchar(200) DEFAULT NULL COMMENT '显示名',  
  `name` varchar(50) DEFAULT NULL COMMENT '镜像的名称唯一',  
  `system` varchar(50) DEFAULT NULL COMMENT '操作系统',  
  `version` varchar(20) DEFAULT NULL COMMENT '系统版本',  
  `url` varchar(255) CHARACTER SET utf8 COLLATE utf8\_bin DEFAULT NULL COMMENT '镜像服务器上镜像的路径',  
  `md5` varchar(50) DEFAULT NULL COMMENT '镜像的MD5值',  
  `format` varchar(20) DEFAULT NULL COMMENT '镜像的格式',  
  `actual\_size\_mb` float DEFAULT NULL COMMENT '镜像实际大小',  
  `size\_gb` int(11) DEFAULT NULL COMMENT '镜像预分配大小',  
  `type` enum('0','1') DEFAULT NULL COMMENT '类型：0表示系统盘，1表示数据盘',  
  `description` varchar(200) DEFAULT NULL COMMENT '描述',  
  `isdeleted` enum('1','0') DEFAULT '1' COMMENT '是否删除： 0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `tag` varchar(100) DEFAULT NULL COMMENT '镜像序列',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=86 DEFAULT CHARSET=utf8 COMMENT='镜像表'

### image\_manage

CREATE TABLE `image\_manage` (  
  `id` int(6) NOT NULL AUTO\_INCREMENT,  
  `eimage\_name` varchar(100) DEFAULT NULL,  
  `status` enum('-1','0','1','2','-10000','3') DEFAULT NULL COMMENT '-1:初始化;0:使用中;1:编辑中;2:待发布;3:发布中;-10000:错误',  
  `message` text COMMENT '反馈的message信息',  
  `template\_vm\_ip` varchar(50) DEFAULT NULL COMMENT '镜像模板机的ip地址',  
  `template\_status` enum('0','1') DEFAULT NULL COMMENT '0:运行中,1:已关机',  
  `related\_image\_tag` varchar(100) DEFAULT '-1',  
  `create\_time` datetime DEFAULT NULL,  
  `update\_time` datetime DEFAULT NULL,  
  `displayname` varchar(100) DEFAULT NULL,  
  `version` varchar(100) DEFAULT NULL,  
  `create\_type` enum('0','1','2','-1') DEFAULT NULL COMMENT '"-1":已有镜像的补录;"0":全新创建"1":从已有镜像创建"2":使用中',  
  `os\_type` varchar(50) DEFAULT NULL COMMENT '镜像模板的OS类型:linux;windows',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=37 DEFAULT CHARSET=utf8

### image\_sync\_schedule

CREATE TABLE `image\_sync\_schedule` (  
  `id` int(255) NOT NULL AUTO\_INCREMENT,  
  `image\_task\_id` int(20) DEFAULT NULL COMMENT '对应的image\_task的id值',  
  `sch\_state` enum('0','1','2','3') DEFAULT NULL COMMENT '任务状态:0未开始;1进行中;2已完成;3错误',  
  `sch\_starttime` datetime DEFAULT NULL COMMENT '任务开始时间',  
  `sch\_endtime` datetime DEFAULT NULL COMMENT '任务结束时间',  
  `isdeleted` enum('0','1') DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=76 DEFAULT CHARSET=utf8

### image\_sync\_task

CREATE TABLE `image\_sync\_task` (  
  `id` int(20) NOT NULL AUTO\_INCREMENT,  
  `image\_id` varchar(20) NOT NULL COMMENT 'image的id值',  
  `host\_ip` varchar(20) NOT NULL COMMENT '目标host的ip',  
  `status` enum('0','1','2','3') NOT NULL COMMENT '0:未开始;1:进行中;2:已完成;3:错误',  
  `type` enum('0','1') DEFAULT NULL COMMENT '0:立即执行;1:计划任务',  
  `schedule\_num` varchar(5) DEFAULT NULL COMMENT '计划任务数量',  
  `start\_time` datetime DEFAULT NULL COMMENT '开始时间',  
  `end\_time` datetime DEFAULT NULL COMMENT '结束时间',  
  `last\_uptime` datetime DEFAULT NULL COMMENT '最后更新时间',  
  `isdeleted` enum('0','1') DEFAULT NULL,  
  `speed\_limit` varchar(10) DEFAULT NULL COMMENT '下载限速,MB',  
  `on\_task` enum('0','1') DEFAULT NULL COMMENT '是否有节点在做',  
  `process\_id` varchar(50) DEFAULT NULL COMMENT '进程id',  
  `message` varchar(100) DEFAULT NULL COMMENT '任务当前的message信息',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=utf8

### image\_update\_status

CREATE TABLE `image\_update\_status` (  
  `id` int(8) NOT NULL AUTO\_INCREMENT,  
  `status` enum('0','1') DEFAULT NULL COMMENT '0:成功;1:失败',  
  `update\_action` varchar(100) DEFAULT NULL,  
  `message` varchar(200) DEFAULT NULL,  
  `update\_time` datetime DEFAULT NULL,  
  `value` varchar(100) DEFAULT NULL,  
  `eimage\_name` varchar(50) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=147 DEFAULT CHARSET=utf8

### imagecache

CREATE TABLE `imagecache` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'id值',  
  `net\_area\_id` int(11) NOT NULL COMMENT 'net\_area的id值',  
  `imagecache\_ip` varchar(50) NOT NULL COMMENT 'imagecache的ip地址',  
  `create\_at` datetime NOT NULL,  
  `update\_at` datetime DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=19 DEFAULT CHARSET=utf8

### increment

CREATE TABLE `increment` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT COMMENT 'ID',  
  `prex\_str` varchar(20) DEFAULT NULL COMMENT '前缀',  
  `increment\_value` int(11) DEFAULT NULL COMMENT '增量值',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=20 DEFAULT CHARSET=utf8

### instance

CREATE TABLE `instance` (  
  `id` int(20) NOT NULL AUTO\_INCREMENT,  
  `uuid` varchar(50) DEFAULT NULL COMMENT 'vm的uuid',  
  `name` varchar(50) DEFAULT NULL COMMENT '实例名',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '显示名称',  
  `status` enum('100','99','98','10','9','8','7','6','5','4','3','2','1','0','101','102','104','103','108') DEFAULT '1' COMMENT '虚拟机状态： 0创建中，1关机，2关机中，3运行中，4开机中，5挂起，6暂停，7热迁移中，8冷迁移中，9转化中，10克隆中，98其他，99错误，100虚机创建失败, ''108''微应用迁移中',  
  `typestatus` enum('2','1','0') DEFAULT '0' COMMENT '业务状态:0表示正常，1表示下线，2表示锁定',  
  `create\_source` enum('2','1','0') DEFAULT '0' COMMENT '创建来源：0:丰云，1:OpenStack 2:ESX',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除；0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `app\_info` varchar(255) DEFAULT NULL COMMENT '应用系统信息',  
  `owner` varchar(50) DEFAULT NULL COMMENT '应用管理员',  
  `password` varchar(100) DEFAULT NULL COMMENT '管理员密码',  
  `description` varchar(255) DEFAULT NULL COMMENT '描述',  
  `p\_cluster\_id` varchar(20) DEFAULT NULL COMMENT 'CMDB中物理集群的ID',  
  `p\_cluster\_name` varchar(50) DEFAULT NULL COMMENT 'CMDB中物理集群的名称',  
  `request\_id` varchar(255) DEFAULT NULL,  
  `task\_id` varchar(255) DEFAULT NULL,  
  `clone\_source\_host` varchar(50) DEFAULT NULL COMMENT '克隆源的HOST IP',  
  `clone\_source\_vm` varchar(50) DEFAULT NULL COMMENT '克隆源的vm名称',  
  `shut\_down\_time` datetime DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  KEY `instance\_id\_owner\_status\_uuid\_instance\_name` (`id`,`owner`,`status`,`uuid`,`name`,`create\_source`)  
) ENGINE=InnoDB AUTO\_INCREMENT=8433 DEFAULT CHARSET=utf8 COMMENT='VM虚拟机表'

### instance\_actions

CREATE TABLE `instance\_actions` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `task\_id` varchar(255) DEFAULT NULL COMMENT '任务ID',  
  `request\_id` varchar(255) DEFAULT NULL COMMENT '请求ID',  
  `action` varchar(255) DEFAULT NULL COMMENT '操作名称如create，start等',  
  `instance\_uuid` varchar(36) DEFAULT NULL,  
  `user\_id` varchar(255) DEFAULT NULL COMMENT '用户工号',  
  `start\_time` datetime DEFAULT NULL COMMENT '启动任务时间',  
  `finish\_time` datetime DEFAULT NULL COMMENT '完成时间',  
  `message` varchar(5000) DEFAULT NULL COMMENT '消息内容',  
  `status` int(11) DEFAULT NULL COMMENT '状态,1代表步骤执行成功，2代表步骤执行失败',  
  PRIMARY KEY (`id`),  
  KEY `instance\_uuid\_idx` (`instance\_uuid`),  
  KEY `request\_id\_idx` (`request\_id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=69902 DEFAULT CHARSET=utf8 COMMENT='instance操作表'

### instance\_actions\_events

CREATE TABLE `instance\_actions\_events` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `event` varchar(255) DEFAULT NULL COMMENT '调用的函数名称',  
  `action\_id` int(11) DEFAULT NULL COMMENT '操作ID关联instance\_actions表',  
  `start\_time` datetime DEFAULT NULL,  
  `finish\_time` datetime DEFAULT NULL,  
  `result` varchar(255) DEFAULT NULL COMMENT 'success or failed',  
  `traceback` text COMMENT '报错记录',  
  `host` varchar(255) DEFAULT NULL COMMENT '主机uuid或虚拟机uuid',  
  `details` text COMMENT '详细信息',  
  PRIMARY KEY (`id`),  
  KEY `action\_id` (`action\_id`),  
  CONSTRAINT `instance\_actions\_events\_ibfk\_1` FOREIGN KEY (`action\_id`) REFERENCES `instance\_actions` (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='操作消息表'

### instance\_clone\_create

CREATE TABLE `instance\_clone\_create` (  
  `id` int(10) NOT NULL AUTO\_INCREMENT,  
  `task\_id` varchar(255) DEFAULT NULL,  
  `source\_vm\_name` varchar(50) DEFAULT NULL,  
  `source\_host\_ip` varchar(50) DEFAULT NULL,  
  `torrent\_num` varchar(10) DEFAULT NULL,  
  `create\_time` datetime DEFAULT NULL,  
  `instance\_id` int(20) DEFAULT NULL,  
  `total\_size` double DEFAULT NULL,  
  `trans\_type` enum('0','1') DEFAULT NULL COMMENT '''0''代表BT,''1''代表wget',  
  `http\_port` varchar(10) DEFAULT NULL,  
  `md5\_check` varchar(255) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=1173 DEFAULT CHARSET=utf8

### instance\_disk

CREATE TABLE `instance\_disk` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `instance\_id` int(11) DEFAULT NULL COMMENT '虚拟机ID，关联instance表',  
  `size\_gb` int(11) DEFAULT NULL COMMENT '数据盘大小，GB',  
  `mount\_point` varchar(20) DEFAULT NULL COMMENT '挂载点',  
  `dev\_name` varchar(4) DEFAULT NULL COMMENT '磁盘dev名，例如vda,vdb,vdc',  
  `disk\_type` enum('0','1') DEFAULT NULL COMMENT '磁盘类型',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除，0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=7981 DEFAULT CHARSET=utf8 COMMENT='VM数据盘表'

### instance\_flavor

CREATE TABLE `instance\_flavor` (  
  `instance\_id` int(11) DEFAULT NULL COMMENT '管理instance表ID',  
  `flavor\_id` int(11) DEFAULT NULL COMMENT '管理规格表flavor表ID',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '修改时间',  
  KEY `pk\_instance\_id` (`instance\_id`),  
  KEY `pk\_flavor\_id` (`flavor\_id`),  
  CONSTRAINT `pk\_flavor\_id` FOREIGN KEY (`flavor\_id`) REFERENCES `flavor` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION,  
  CONSTRAINT `pk\_instance\_id` FOREIGN KEY (`instance\_id`) REFERENCES `instance` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='虚拟机机规格关联表'

### instance\_group

CREATE TABLE `instance\_group` (  
  `instance\_id` int(20) DEFAULT NULL,  
  `group\_id` int(20) DEFAULT NULL,  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  KEY `fk\_instance\_id` (`instance\_id`),  
  CONSTRAINT `fk\_instance\_id` FOREIGN KEY (`instance\_id`) REFERENCES `instance` (`id`) ON DELETE NO ACTION ON UPDATE NO ACTION  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='虚拟机管理的应用组关联表'

### instance\_host

CREATE TABLE `instance\_host` (  
  `instance\_id` bigint(20) DEFAULT NULL COMMENT '管理虚拟机表ID',  
  `instance\_name` varchar(50) DEFAULT NULL COMMENT '虚拟机的名称',  
  `host\_id` bigint(20) DEFAULT NULL COMMENT '管理物理机的ID',  
  `host\_name` varchar(50) DEFAULT NULL COMMENT '物理机的主机名称',  
  `isdeleted` enum('0','1') DEFAULT NULL COMMENT '是否删除，0未删除，1已删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '修改时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  KEY `instance\_host\_host\_id\_instance\_id` (`host\_id`,`instance\_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='虚拟机与HOST关联表'

### instance\_image

CREATE TABLE `instance\_image` (  
  `instance\_id` int(11) NOT NULL COMMENT '虚拟机ID',  
  `image\_id` int(11) DEFAULT NULL COMMENT '关联镜像ID',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '表示是否删除：0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  KEY `instance\_id` (`instance\_id`,`image\_id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='虚拟机镜像表'

### instance\_ip

CREATE TABLE `instance\_ip` (  
  `ip\_id` int(11) DEFAULT NULL COMMENT '管理IP表的ID',  
  `instance\_id` int(11) DEFAULT NULL COMMENT '管理虚拟机表的ID',  
  `mac` varchar(20) DEFAULT NULL COMMENT 'MAC地址',  
  `type` enum('0','1') DEFAULT NULL COMMENT '网卡类型，''0''主网网卡，''1''普通网卡',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除： 0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  KEY `instance\_ip\_instance\_id\_ip\_id\_isdeleted` (`instance\_id`,`ip\_id`,`isdeleted`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='虚拟机IP关联表'

### instance\_migrate

CREATE TABLE `instance\_migrate` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `instance\_id` int(11) DEFAULT NULL COMMENT '虚拟机ID，关联instance表',  
  `src\_host\_id` int(11) DEFAULT NULL COMMENT '源物理机ID，关联host表',  
  `dst\_host\_id` int(11) DEFAULT NULL COMMENT '目标物理机ID，关联host表',  
  `nc\_port` int(11) DEFAULT NULL COMMENT 'nc使用端口',  
  `migrate\_status` enum('0','1','2') DEFAULT '0' COMMENT '0为迁移中，1为迁移完成, 2为迁移失败',  
  `created\_at` datetime DEFAULT NULL COMMENT '迁移任务开始时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '迁移任务完成时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=224 DEFAULT CHARSET=utf8

### ip

CREATE TABLE `ip` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `ip\_address` varchar(15) DEFAULT NULL COMMENT 'IP 地址',  
  `segment\_id` int(11) DEFAULT NULL COMMENT '管理网段表ID',  
  `netmask` varchar(2) DEFAULT NULL COMMENT '子网掩码',  
  `vlan` varchar(10) DEFAULT NULL COMMENT 'vlan ID ',  
  `gateway\_ip` varchar(15) DEFAULT NULL COMMENT '网关',  
  `dns1` varchar(15) DEFAULT NULL COMMENT 'DNS1',  
  `dns2` varchar(15) DEFAULT NULL COMMENT 'DNS2',  
  `status` enum('0','1','2','3') DEFAULT '0' COMMENT '状态:0表示未使用，1表示已使用，2表示已保留',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `group\_id` int(11) DEFAULT NULL COMMENT 'IP所属应用组',  
  PRIMARY KEY (`id`),  
  KEY `ip\_id\_ip\_address` (`id`,`ip\_address`,`group\_id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=9327 DEFAULT CHARSET=utf8 COMMENT='IP表'

### ip\_lock

CREATE TABLE `ip\_lock` ( `id` int(11) NOT NULL AUTO\_INCREMENT, `table\_name` varchar(100) DEFAULT NULL COMMENT '表名', `istraceing` enum('0','1') DEFAULT NULL COMMENT '需要确保资源唯一性的表格是否有更新中', PRIMARY KEY (`id`)) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8

### net\_area

CREATE TABLE `net\_area` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT '网络区域名称',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '显示名',  
  `datacenter\_id` int(11) DEFAULT NULL COMMENT '管理机房ID',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除：0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`),  
  KEY `net\_area\_id\_datacenter\_id\_isdeleted` (`datacenter\_id`,`id`,`isdeleted`)  
) ENGINE=InnoDB AUTO\_INCREMENT=107 DEFAULT CHARSET=utf8 COMMENT='网络区域表'

### network\_segment

CREATE TABLE `network\_segment` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `net\_area\_id` int(11) DEFAULT NULL COMMENT '网络区域ID',  
  `segment` varchar(15) DEFAULT NULL COMMENT '网段名称',  
  `segment\_type` enum('0','1','2','3','4','5','6') DEFAULT NULL COMMENT '网段类型，0为内网，1为外网电信，2为外网联通，3为镜像模板机使用，4为内网NAS网段，5为腾讯云容灾补搭网段，6为腾讯云容灾haproxy网段',  
  `host\_bridge\_name` varchar(50) DEFAULT NULL COMMENT '大网段所在桥设备，如：br\_bond0',  
  `netmask` varchar(2) DEFAULT NULL COMMENT '子网掩码',  
  `vlan` varchar(10) DEFAULT NULL COMMENT 'vlan ID ',  
  `gateway\_ip` varchar(15) DEFAULT NULL COMMENT '网关IP',  
  `dns1` varchar(15) DEFAULT NULL COMMENT 'DNS1',  
  `dns2` varchar(15) DEFAULT NULL COMMENT 'DNS2',  
  `status` int(11) DEFAULT NULL COMMENT '状态',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=68 DEFAULT CHARSET=utf8 COMMENT='网段表'

### obj\_req\_map

CREATE TABLE `obj\_req\_map` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `obj\_id` int(11) DEFAULT NULL COMMENT '操作对象ID',  
  `obj\_type` varchar(20) DEFAULT NULL COMMENT '操作对象类型',  
  `req\_id` varchar(50) DEFAULT NULL COMMENT '请求ID',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8

### operation\_records

CREATE TABLE `operation\_records` (  
  `id` int(20) NOT NULL AUTO\_INCREMENT,  
  `operator` varchar(20) DEFAULT NULL COMMENT '操作者',  
  `operator\_ip` varchar(20) DEFAULT NULL COMMENT '操作者IP',  
  `operation\_object` varchar(10) DEFAULT NULL COMMENT '操作对象',  
  `operation\_action` varchar(20) DEFAULT NULL COMMENT '操作动作',  
  `operation\_date` datetime DEFAULT NULL COMMENT '操作时间',  
  `operation\_result` varchar(12) DEFAULT NULL COMMENT '操作结果',  
  `extra\_data` text COMMENT '描述详情',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=4996 DEFAULT CHARSET=utf8

### permission

CREATE TABLE `permission` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT '名称',  
  `displayname` varchar(50) DEFAULT NULL COMMENT '显示名称',  
  `module` varchar(20) DEFAULT NULL COMMENT '模块名称',  
  `path` varchar(50) DEFAULT NULL COMMENT 'URL',  
  `acl` enum('1','0') DEFAULT '1' COMMENT '1表示允许，0表示拒绝',  
  `area` enum('other','all') DEFAULT 'all' COMMENT '范围',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=25 DEFAULT CHARSET=utf8 COMMENT='权限功能详细表'

### problem\_category\_info

CREATE TABLE `problem\_category\_info` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `parent\_id` int(11) NOT NULL COMMENT '问题分类ID',  
  `category` varchar(150) DEFAULT NULL COMMENT '问题的详细分类',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=102 DEFAULT CHARSET=utf8

### request\_ip\_permit

CREATE TABLE `request\_ip\_permit` (  
  `ip\_permit` varchar(50) NOT NULL COMMENT '允许访问接口外部ip地址',  
  PRIMARY KEY (`ip\_permit`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8

### request\_record

CREATE TABLE `request\_record` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `api\_origin` enum('0','1','2') DEFAULT NULL COMMENT '''0''为维石平台任务，''1''为软负载任务，''2''WAF',  
  `taskid\_api` varchar(255) DEFAULT NULL COMMENT '外部接口创建虚拟机带过来的步骤id',  
  `taskid\_kvm` varchar(255) NOT NULL COMMENT '任务id',  
  `vm\_count` int(5) DEFAULT NULL COMMENT '外部接口创建虚拟机数量',  
  `user\_id` varchar(255) DEFAULT NULL COMMENT '任务操作用户id',  
  `start\_time` datetime DEFAULT NULL COMMENT '任务启动时间',  
  `finish\_time` datetime DEFAULT NULL COMMENT '完成时间',  
  `task\_status` varchar(11) DEFAULT NULL COMMENT '任务状态,0代表任务执行中，1代表任务执行成功，2代表任务执行失败，3代表任务部分执行成功',  
  `response\_to\_api` varchar(5) DEFAULT NULL COMMENT '0代表未返回接口任务执行结果，1代表已经返回接口任务执行结果',  
  `request\_status\_collect\_time` datetime DEFAULT NULL COMMENT '工单收集程序上一次收集工单状态时间',  
  `istraceing` enum('0','1') DEFAULT NULL COMMENT '判断是否有进程正在跟踪此工单',  
  `request\_ip` varchar(50) DEFAULT NULL COMMENT '请求任务的ip地址',  
  `retry\_time` enum('3','2','1','0') DEFAULT '0' COMMENT '维石创建虚拟机失败重试，最多3次',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=297 DEFAULT CHARSET=utf8

### role

CREATE TABLE `role` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT '1:管理员；2:应用管理员；3：只读用户',  
  `type` varchar(20) DEFAULT NULL COMMENT '类型',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COMMENT='角色表'

### role\_permission

CREATE TABLE `role\_permission` (  
  `role\_id` int(11) DEFAULT NULL COMMENT '关联角色ID',  
  `permission\_id` int(11) DEFAULT NULL COMMENT '管理权限ID'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='角色功能对应表'

### segment\_match

CREATE TABLE `segment\_match` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT COMMENT '记录ID',  
  `prd\_segment\_id` int(11) NOT NULL COMMENT '生产环境网段ID',  
  `dr\_segment\_id` int(11) NOT NULL COMMENT '容灾环境网段ID',  
  `isdeleted` enum('0','1') DEFAULT '0' COMMENT '是否删除',  
  PRIMARY KEY (`id`,`prd\_segment\_id`,`dr\_segment\_id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8

### send\_msg

CREATE TABLE `send\_msg` (  
  `id` int(11) DEFAULT NULL,  
  `req\_id` varchar(50) DEFAULT NULL COMMENT '请求ID',  
  `routing\_key` varchar(20) DEFAULT NULL COMMENT '消息类型KEY',  
  `send\_time` datetime DEFAULT NULL COMMENT '发送时间',  
  `message` varchar(5000) DEFAULT NULL COMMENT '消息内容',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8

### tb\_group

CREATE TABLE `tb\_group` (  
  `id` int(11) unsigned NOT NULL AUTO\_INCREMENT,  
  `name` varchar(50) DEFAULT NULL COMMENT '组名',  
  `displayname` varchar(200) DEFAULT NULL COMMENT '显示名',  
  `description` varchar(255) DEFAULT NULL COMMENT '描述',  
  `dc\_type` enum('0','1','2','3','4','5','6','7','8','9') DEFAULT NULL COMMENT '机房类型：0:其他、1:测试SIT、2:准生产STG、3:研发DEV、4:生产PRD、5:容灾DR、6:微应用DR、8''容灾压测PST、''9''测试压测IST',  
  `owner` varchar(20) DEFAULT NULL COMMENT '应用组的创建者ID',  
  `cpu` int(11) DEFAULT '120' COMMENT '应用组cpu配额',  
  `mem` int(11) DEFAULT '960' COMMENT '应用组mem配额， G',  
  `disk` int(11) DEFAULT '1000' COMMENT '应用组disk配额，G',  
  `vm` int(11) DEFAULT '200' COMMENT '应用组vm配额',  
  `p\_cluster\_id` varchar(20) DEFAULT NULL COMMENT 'CMDB中物理集群ID',  
  `isdeleted` enum('1','0') CHARACTER SET utf8 COLLATE utf8\_croatian\_ci DEFAULT '0' COMMENT '是否删除： 0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `pk\_id` (`id`) USING BTREE,  
  KEY `tb\_group\_group\_id\_group\_name` (`id`,`name`,`owner`)  
) ENGINE=InnoDB AUTO\_INCREMENT=230 DEFAULT CHARSET=utf8 COMMENT='用户组表'

### tb\_host\_perform

CREATE TABLE `tb\_host\_perform` (  
  `id` bigint(50) NOT NULL AUTO\_INCREMENT,  
  `ip` varchar(25) DEFAULT NULL COMMENT '物理机的IP',  
  `hostname` varchar(30) DEFAULT NULL COMMENT '物理机的主机名',  
  `collect\_time` datetime DEFAULT NULL COMMENT '性能采集时间',  
  `metric\_key` varchar(50) DEFAULT NULL COMMENT '性能采集的指标',  
  `data\_value` varchar(2000) DEFAULT NULL COMMENT '性能采集的值',  
  PRIMARY KEY (`id`),  
  KEY `px\_ip` (`ip`) USING BTREE,  
  KEY `px\_hostname` (`hostname`) USING BTREE,  
  KEY `px\_ip\_hostname` (`ip`,`hostname`),  
  KEY `px\_metric` (`metric\_key`),  
  KEY `host\_perform\_collect\_time` (`collect\_time`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='物理机性能数据表（实时表）'

### tb\_user

CREATE TABLE `tb\_user` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `userid` varchar(20) DEFAULT NULL COMMENT '登录用户工号',  
  `username` varchar(50) DEFAULT NULL COMMENT '姓名',  
  `email` varchar(50) DEFAULT NULL COMMENT '邮件地址',  
  `telephone` varchar(50) DEFAULT NULL COMMENT '电话号码',  
  `fengsheng` varchar(50) DEFAULT NULL COMMENT '丰声地址',  
  `status` enum('2','1','0') DEFAULT '0' COMMENT '状态(0:正常，1：锁定，2：其他)',  
  `auth\_type` int(11) DEFAULT '0' COMMENT '用户认证类型，0为ad，1为本地, 2为外部应用用户',  
  `password` varchar(50) DEFAULT NULL COMMENT '密码',  
  `isdeleted` enum('1','0') DEFAULT '0' COMMENT '是否删除：0表示已经删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=268 DEFAULT CHARSET=utf8 COMMENT='用户表'

### user\_group

CREATE TABLE `user\_group` (  
  `user\_id` varchar(20) DEFAULT NULL COMMENT '关联用户工号',  
  `user\_name` varchar(11) DEFAULT NULL COMMENT '用户工号',  
  `group\_id` int(11) DEFAULT NULL COMMENT '管理组ID',  
  `group\_name` varchar(50) DEFAULT NULL COMMENT '组名',  
  `role\_id` int(11) DEFAULT NULL COMMENT '这个字段实际作废/角色ID',  
  `status` enum('1','0') DEFAULT '0' COMMENT '状态',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `expire\_at` datetime DEFAULT NULL COMMENT '过期时间',  
  KEY `ix\_group\_id` (`group\_id`) USING BTREE,  
  KEY `ix\_user\_id` (`user\_name`) USING BTREE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='用户与组关联表'

### v2v\_instance\_info

CREATE TABLE `v2v\_instance\_info` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `instance\_id` int(11) DEFAULT NULL COMMENT '虚拟机ID，关联instance表',  
  `os\_type` varchar(20) DEFAULT NULL COMMENT '虚拟机OS类型（windows/linux）',  
  `os\_version` varchar(50) DEFAULT NULL COMMENT '虚拟机OS版本',  
  `isdeleted` enum('0','1') DEFAULT '0' COMMENT '是否删除；0表示未删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `updated\_at` datetime DEFAULT NULL COMMENT '更新时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  `request\_id` varchar(50) DEFAULT NULL COMMENT 'v2v任务id',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=63 DEFAULT CHARSET=utf8

### v2v\_task

CREATE TABLE `v2v\_task` (  
  `request\_id` varchar(255) NOT NULL COMMENT '请求ID',  
  `destory` varchar(2) DEFAULT '0' COMMENT '销毁字段,1代表销毁，0代表未销毁',  
  `start\_time` datetime DEFAULT NULL COMMENT '开始时间',  
  `finish\_time` datetime DEFAULT NULL COMMENT '结束时间',  
  `status` int(2) DEFAULT NULL COMMENT '任务状态,0为进行中,1为已完成,2为出错,3为已取消',  
  `vm\_ip` varchar(50) DEFAULT NULL COMMENT 'vmip',  
  `step\_done` varchar(50) DEFAULT NULL COMMENT '当前已完成的进度',  
  `user\_id` varchar(50) DEFAULT NULL COMMENT '用户id',  
  `dest\_dir` varchar(100) DEFAULT NULL COMMENT '目标vm路径',  
  `dest\_host` varchar(50) DEFAULT NULL COMMENT '目标kvm host',  
  `vm\_name` varchar(50) DEFAULT NULL COMMENT '虚拟机名称',  
  `vmvlan` varchar(10) DEFAULT NULL COMMENT '虚拟机vlan',  
  `flavor\_id` varchar(50) DEFAULT NULL COMMENT '虚拟机flavor\_id',  
  `cloud\_area` varchar(20) DEFAULT NULL COMMENT '虚拟机云环境',  
  `vm\_ostype` varchar(20) DEFAULT NULL COMMENT '虚拟机OS类型',  
  `vm\_app\_info` varchar(50) DEFAULT NULL COMMENT '虚拟机应用信息',  
  `vm\_owner` varchar(20) DEFAULT NULL COMMENT '应用负责人',  
  `vm\_group\_id` varchar(20) DEFAULT NULL COMMENT '应用组',  
  `cancel` varchar(10) DEFAULT NULL COMMENT '是否取消任务：1为取消，0不取消',  
  `vm\_uuid` varchar(100) DEFAULT NULL COMMENT '虚拟机uuid',  
  `vm\_mac` varchar(50) DEFAULT NULL COMMENT '虚拟机MAC',  
  `message` varchar(100) DEFAULT NULL COMMENT '任务处理详细信息',  
  `on\_task` varchar(10) DEFAULT NULL COMMENT '任务是否在处理；0为未处理，1为进行中',  
  `port` varchar(50) DEFAULT NULL COMMENT 'nc所使用port',  
  `id` int(10) NOT NULL AUTO\_INCREMENT COMMENT 'id',  
  `source` varchar(10) DEFAULT NULL COMMENT 'v2v来源',  
  `esx\_ip` varchar(20) DEFAULT NULL COMMENT 'esx ip地址',  
  `esx\_passwd` varchar(25) DEFAULT NULL COMMENT 'esx root密码',  
  `volumelist` varchar(10) DEFAULT NULL COMMENT 'vm磁盘数量',  
  `vmware\_vm` varchar(50) DEFAULT NULL COMMENT 'vmware平台vm名称',  
  PRIMARY KEY (`request\_id`),  
  KEY `id` (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=75 DEFAULT CHARSET=utf8

### vip\_info

CREATE TABLE `vip\_info` (  
  `id` int(11) NOT NULL AUTO\_INCREMENT,  
  `ip\_id` int(11) NOT NULL COMMENT '此字段为ip表中的id字段，用于表示ip唯一性',  
  `cluster\_id` varchar(20) DEFAULT NULL COMMENT '使用此ip的物理集群id',  
  `apply\_user\_id` varchar(30) DEFAULT NULL COMMENT '申请vip的用户工号',  
  `sys\_code` varchar(50) DEFAULT NULL COMMENT 'cmdb中对应系统编码',  
  `isdeleted` enum('0','1') DEFAULT '0' COMMENT '是否删除;0表示未删除，1表示已删除',  
  `created\_at` datetime DEFAULT NULL COMMENT '创建时间',  
  `deleted\_at` datetime DEFAULT NULL COMMENT '删除时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=135 DEFAULT CHARSET=utf8

### vm\_disks

CREATE TABLE `vm\_disks` (  
  `id` bigint(20) NOT NULL AUTO\_INCREMENT,  
  `instance\_uuid` varchar(11) DEFAULT NULL,  
  `instance\_name` varchar(50) DEFAULT NULL,  
  `path` varchar(255) DEFAULT NULL,  
  `image` varchar(50) DEFAULT NULL,  
  `storage\_pool` varchar(50) DEFAULT NULL,  
  `devices` varchar(20) DEFAULT NULL,  
  `file\_format` varchar(20) DEFAULT NULL,  
  `insert\_datetime` datetime DEFAULT NULL COMMENT '数据插入时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='vm实时磁盘信息'

### vm\_networks

CREATE TABLE `vm\_networks` (  
  `id` bigint(20) NOT NULL AUTO\_INCREMENT,  
  `instance\_uuid` varchar(50) DEFAULT NULL,  
  `network` varchar(50) DEFAULT NULL,  
  `nic` varchar(10) DEFAULT NULL,  
  `mac\_address` varchar(20) DEFAULT NULL,  
  `instance\_name` tinyblob,  
  `insert\_datetime` datetime DEFAULT NULL COMMENT '数据插入时间',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8 COMMENT='vm实时网络信息表'

### vm\_osinfo

CREATE TABLE `vm\_osinfo` ( `id` bigint(20) NOT NULL AUTO\_INCREMENT, `instance\_uuid` varchar(50) DEFAULT NULL, `instance\_name` varchar(50) DEFAULT NULL, `vcpu` int(11) DEFAULT NULL, `memory\_mb` int(11) DEFAULT NULL, `disks` int(11) DEFAULT NULL, `status` int(11) DEFAULT NULL, `insert\_datetime` datetime DEFAULT NULL COMMENT '数据插入时间', PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='vm实时配置信息表'

## Image操作指引

|  |
| --- |
| **1安装Apache（httpd）** |
| **1.1 安装httpd** |
| [root@image01 image]# yum install httpd |
| **1.2检查是否安装成功** |
| [root@image01 image]# rpm -qa |grep httpd |
| httpd-tools-2.4.6-40.el7.centos.x86\_64 |
| httpd-2.4.6-40.el7.centos.x86\_64 |
| **1.3将httpd配置为开机自启动** |
| [root@image01 image]# systemctl enable httpd.service |
| **1.4 启动httpd** |
| [root@image01 image]# systemctl start httpd.service |
| **1.5 查看httpd状态** |
| [root@image01 image]# systemctl status httpd.service |
| （注：image02与image01操作相同） |
|  |
| **2 httpd配置web共享** |
| 共享文件夹（/app/image/）。 |
| **2.1 修改httpd配置文件httpd.conf** |
| [root@image01 image]# vim /etc/httpd/conf/httpd.conf |
| # 配置文件中修改的部分如下，找到相应位置进行修改即可 |
| # 服务所在目录 |
| ServerRoot "/etc/httpd" |
| # 监听端口 |
| Listen 10.202.66.27:80（此处的ip请相应替换成当前vm的ip） |
| # 服务名 |
| ServerName localhost:80 |
| # 共享文件所在目录 |
| DocumentRoot "/app/image" |
| # 共享文件目录的相应配置 |
| <Directory "/app/image/"> |
| Options Indexes |
| Require all granted |
| #Deny From all |
| </Directory> |
| **2.2 重启httpd服务** |
| [root@image01 image]# systemctl restart httpd.service |
| （注：image2与image1操作相同） |

## Cache操作指引

|  |
| --- |
| Cache的技术方案采用squid标准代理方式。 |
|  |
|  |
| imagecache01\imagecache02都执行如下操作 |
| 1、安装squid #yum install squid |
| 2、配置squid #mkdir /app/squid #chown squid.squid /app/squid #vim /etc/squid/squid.conf 修改本机监听端口 http\_port 3128 允许所有客户端可以访问 http\_access allow all 配置Cache目录 cache\_dir ufs /app/squid/ 204800 16 256  # Set cache cache\_mem 1024 MB cache\_swap\_low 90 cache\_swap\_high 97 minimum\_object\_size 0 KB maximum\_object\_size 50 GB  refresh\_pattern . 129600 100% 129600 reload-into-ims |
| 3、配置/etc/hosts（初始步骤已完成该操作，可略过） #vim /etc/hosts 10.202.66.30 imagecache01 |
| 4、检查配置是否正常 #squid -k parse |
| 5、初始化squid #squid -z |
| 6、配置开机启动 #chkconfig squid on #service squid start |
| 7、测试Squid是否生效 A：登陆一台HOST，执行以下操作（先配置好环境变量） proxy\_address="http://imagecache01-IP:3128" export http\_proxy="${proxy\_address}" export https\_proxy="${proxy\_address}" export ftp\_proxy="${proxy\_address}" B：在HOST上执行以下操作，下载镜像 wget http://image01-IP/centos7.2\_jboss/centos7.2\_jboss C：在imagecache01-IP上执行以下操作，检查镜像是否被Cache df -h /app 如果占用的大小为centos7.2\_jboss的大小，说明已经被缓存。imagecache01生效。 |

## 镜像管理机操作指引

|  |
| --- |
| 1.HOST标准化配置:参考HOST标准化配置手册（简单做法，利用平台的host添加功能直接加上去后踢出来） |
| 2.将以下脚本文件拷贝至/root下并命名为vm\_create.sh #!/bin/sh  #入参获取vm名称和磁盘文件路径 dir=$1 vmname=$2 ostype=$3  #设置vm的网卡信息 vmvlan=br\_bond0  #拷贝原始xml文件至当前目录 cd $dir echo > diskfile  if [ $ostype = "linux" ];then cp /root/orginal\_linux.xml $vmname.xml else  cp /root/orginal\_windows.xml $vmname.xml fi  #修改系统盘名称并写入disk文件 cd $dir  cat << EOF >> diskfile  <disk type='file' device='disk'>  <driver name='qemu' type='qcow2'/>  <source file='$dir/$vmname'/>  <target dev='vda' bus='virtio'/>  </disk> EOF  disktag=(vdb vdc vdd vde vdf vdg vdh vdi vdj)  #修改数据盘名称并写入disk文件 i=0 m=2 for n in `ls $dir|grep $vmname|grep disk` do  cat << EOF >> diskfile  <disk type='file' device='disk'>  <driver name='qemu' type='qcow2'/>  <source file='$dir/${vmname}\_disk$m'/>  <target dev='${disktag[$i]}' bus='virtio'/>  </disk> EOF let i+=1 let m+=1 done  #将disk文件添加到vm的xml文件中 sed -i '/<devices>/r diskfile' $vmname.xml  #修改xml中vm名称 sed -i "s/orginal/$vmname/g" $vmname.xml  #create vm virsh define $vmname.xml  #remove temfile rm -rf $vmname.xml rm -rf diskfile |
| 3.将以下文件拷贝至/root目录并重命名为orginal\_linux.xml <?xml version='1.0' encoding='utf-8'?> <domain type='kvm'>  <!-- generated by virt-v2v 1.32.7rhel=7,release=3.el7.centos,libvirt -->  <name>orginal</name>  <memory unit='KiB'>4194304</memory>  <currentMemory unit='KiB'>4194304</currentMemory>  <vcpu placement='static' current='2'>16</vcpu>  <os>  <type arch='x86\_64'>hvm</type>  <bootmenu enable='yes'/>  </os>  <features>  <acpi/>  <apic/>  <pae/>  </features>  <clock offset='localtime'/>  <on\_poweroff>destroy</on\_poweroff>  <on\_reboot>restart</on\_reboot>  <on\_crash>restart</on\_crash>  <devices>  <interface type='bridge'>  <source bridge='brvlan187'/>  <model type='virtio'/>  </interface>  <video>  <model type='qxl' ram='65536' heads='1'/>  </video>  <channel type='unix'>  <target type='virtio' name='org.qemu.guest\_agent.0'/>  </channel>  <graphics type='vnc' port='-1' autoport='yes' listen='0.0.0.0'>  <listen type='address' address='0.0.0.0'/>  </graphics>  <input type='tablet' bus='usb'/>  <input type='mouse' bus='ps2'/>  <console type='pty'/>  </devices> </domain> |