一　ceph

１，前言

　　分布式文件系统（Distributed File System）：文件系统管理的物理存储资源不一定直接连接在本地节点上，而是通过计算机网络与节点相连．分布式文件系统的设计基于C/S模式

－－常用的分布式文件系统

Lustre

Hadoop

FastDFS

Ceph (前景很好)

GlusterFS

－－什么是Ceph

Ceph是一个分布式文件系统

特点：

具有高扩展，高可用，高性能的特点

可以提供对象存储，块存储，文件存储

可以提供PB级别的存储空间（PB->TB->GB）

帮助文档：<http://docs.ceph.org/start/intro>

中文文档：http://docs.ceph.org.cn/

1. Ceph核心组件

OSD：存储设备（OSD越多性能越好）

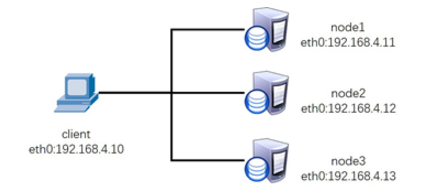
Monitor：集群监控组件　（取基数，如3，5，7）

MDS：存放文件系统的元数据（对象存储和块存储不需要该组件）

Client：ceph客户端

２，实验环境

虚拟机：4台



真实机：（centos7）或者重新新建一台虚拟机代替真机也可以

３，安装前准备

1）物理机为所有节点配置yum源服务器。

[root@room9pc01 ~]# yum -y install vsftpd

[root@room9pc01 ~]# mkdir /var/ftp/ceph

[root@room9pc01 ~]# mount -o loop \

rhcs2.0-rhosp9-20161113-x86\_64.iso /var/ftp/ceph

[root@room9pc01 ~]# systemctl restart vsftpd

2）修改所有节点都需要配置YUM源（这里仅以node1为例）。

[root@node1 ceph-cluster]# cat /etc/yum.repos.d/ceph.repo

[mon]

name=mon

baseurl=ftp://192.168.4.254/ceph/rhceph-2.0-rhel-7-x86\_64/MON

gpgcheck=0

[osd]

name=osd

baseurl=ftp://192.168.4.254/ceph/rhceph-2.0-rhel-7-x86\_64/OSD

gpgcheck=0

[tools]

name=tools

baseurl=ftp://192.168.4.254/ceph/rhceph-2.0-rhel-7-x86\_64/Tools

gpgcheck=0

3）修改/etc/hosts并同步到所有主机。

警告：/etc/hosts解析的域名必须与本机主机名一致！！！！

[root@node1 ceph-cluster]# vim /etc/hosts

... ...

192.168.4.10 　client

192.168.4.11 node1

192.168.4.12 node2

192.168.4.13 node3

警告：/etc/hosts解析的域名必须与本机主机名一致！！！！

[root@node1 ceph-cluster]# for i in 10 11 12 13

> do

> scp /etc/hosts 192.168.4.$i:/etc/

> done

[root@node1 ceph-cluster]# for i in 10 11 12 13

> do

> scp /etc/yum.repos.d/ceph.repo 192.168.4.$i:/etc/yum.repos.d/

> done

3）配置无密码连接(包括自己远程自己也不需要密码)。

[root@node1 ceph-cluster]# ssh-keygen -f /root/.ssh/id\_rsa -N ''　＃生成密钥

[root@node1 ceph-cluster]# for i in 10 11 12 13　　＃上传密钥到个个主机

> do

> ssh-copy-id 192.168.4.$i

> done

4，配置NTP时间同步

1）真实物理机创建NTP服务器。

[root@room9pc01 ~]# yum -y install chrony

[root@room9pc01~]# cat /etc/chrony.conf

server 0.centos.pool.ntp.org iburst

allow 192.168.4.0/24

local stratum 10

[root@room9pc01 ~]# systemctl restart chronyd

[root@room9pc01 ~]# iptables -F　　　　　＃如果有防火墙规则，需要清空所有规则

2）其他所有节点与NTP服务器同步时间（以node1为例）。

[root@node1 ceph-cluster]# cat /etc/chrony.conf

server 192.168.4.254 iburst

[root@node1 ceph-cluster]# systemctl restart chronyd

5，准备存储磁盘

1）物理机上为每个虚拟机准备3块磁盘。（可以使用命令，建议使用图形直接添加）

[root@room9pc01 ~]# cd /var/lib/libvirt/images

[root@room9pc01 ~]# qemu-img create -f qcow2 node1-vdb.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node1-vdc.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node1-vdd.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node2-vdb.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node2-vdc.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node2-vdd.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node3-vdb.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node3-vdc.vol 10G

[root@room9pc01 ~]# qemu-img create -f qcow2 node3-vdd.vol 10G

2）使用virt-manager为虚拟机添加磁盘。（或者直接打开虚拟机管理器添加）

[root@room9pc01 ~]# virt-manager

6，部署ceph集群

－－部署软件

1）在node1安装部署工具，学习工具的语法格式。

[root@node1 ～]# yum -y install ceph-deploy

[root@node1 ～]# ceph-deploy --help　　　　　＃查看命令格式

2）创建目录

[root@node1 ceph-cluster]# mkdir ceph-cluster

[root@node1 ceph-cluster]# cd ceph-cluster/

－－部署Ceph集群

1）创建Ceph集群配置。

[root@node1 ceph-cluster]# ceph-deploy new node1 node2 node3

2）给所有节点安装软件包。

[root@node1 ceph-cluster]# ceph-deploy install node1 node2 node3

1. 初始化所有节点的mon服务（主机名解析必须对）

[root@node1 ceph-cluster]# vim ceph.conf #文件最后追加以下内容

public\_network = 192.168.4.0/24　　　＃避免会出现如下错误加上这一行

[root@node1 ceph-cluster]# ceph-deploy mon create-initial　//拷贝配置文件，并启动mon服务

ceph.conf

常见错误及解决方法（非必要操作，有错误可以参考）：

如果提示如下错误信息：

[node1][ERROR ] admin\_socket: exception getting command descriptions: [Error 2] No such file or directory

解决方案如下（在node1操作）：

先检查自己的命令是否是在ceph-cluster目录下执行的！！！！如果时确认是在该目录下执行的create-initial命令，依然保存，可以使用如下方式修复。

[root@node1 ceph-cluster]# vim ceph.conf #文件最后追加以下内容

public\_network = 192.168.4.0/24

修改后重新推送配置文件:

[root@node1 ceph-cluster]# ceph-deploy --overwrite-conf config push node1 node2 node3

－－创建OSD

1）准备磁盘分区（node1,node2,node3都要做）

[root@node1 ceph-cluster]# parted /dev/vdb mklabel gpt

[root@node1 ceph-cluster]# parted /dev/vdb mkpart primary 1M 50%

[root@node1 ceph-cluster]# parted /dev/vdb mkpart primary 50% 100%

[root@node1 ceph-cluster]# chown ceph.ceph /dev/vdb1　＃授权给ceph用户

[root@node1 ceph-cluster]# chown ceph.ceph /dev/vdb2

提示：重启后失效，要想永久有效，１，把命令写入/etc/rc.local文件中，并给文件赋予执行权限;２，用udev写入权限

//这两个分区用来做存储服务器的日志journal盘

2）初始化清空磁盘数据（仅node1操作即可）

[root@node1 ceph-cluster]# ceph-deploy disk zap node1:vdc node1:vdd

[root@node1 ceph-cluster]# ceph-deploy disk zap node2:vdc node2:vdd

[root@node1 ceph-cluster]# ceph-deploy disk zap node3:vdc node3:vdd

提示：如果已经初始化过了，需要重新清空磁盘数据，就要取（node1,node2,node3）卸载掉

3）创建OSD存储空间（仅node1操作即可）

[root@node1 ceph-cluster]# ceph-deploy osd create node1:vdc:/dev/vdb1 node1:vdd:/dev/vdb2

//创建osd存储设备，vdc为集群提供存储空间，vdb1提供JOURNAL日志，

//一个存储设备对应一个日志设备，日志需要SSD，不需要很大

[root@node1 ceph-cluster]# ceph-deploy osd create node2:vdc:/dev/vdb1 node2:vdd:/dev/vdb2

[root@node1 ceph-cluster]# ceph-deploy osd create node3:vdc:/dev/vdb1 node3:vdd:/dev/vdb2

4）常见错误（非必须操作）

使用osd create创建OSD存储空间时，如提示run 'gatherkeys'，可以使用如下命令修复：

[root@node1 ceph-cluster]# ceph-deploy gatherkeys node1 node2 node3

－－验证测试

1) 查看集群状态

[root@node1 ceph-cluster]# ceph -s

cluster d3266b4f-5c8c-44f3-b11e-7a2cafd1f75b

health HEALTH\_OK　　　　＃出现ok就成功了，err就失败了，如果出现warning,执行 systemctl restart ceph\\*.service ceph\\*.target重启服务

monmap e1: 3 mons at {node1=192.168.4.11:6789/0,node2=192.168.4.12:6789/0,node3=192.168.4.13:6789/0}

election epoch 6, quorum 0,1,2 node1,node2,node3

osdmap e42: 6 osds: 6 up, 6 in

flags sortbitwise

pgmap v323: 64 pgs, 1 pools, 88496 kB data, 5147 objects

503 MB used, 119 GB / 119 GB avail

64 active+clean

2）常见错误（非必须操作）

如果查看状态包含如下信息：

health: HEALTH\_WARN

clock skew detected on node2, node3…

clock skew表示时间不同步，解决办法：请先将所有主机的时间都使用NTP时间同步！！！

如果状态还是失败，可以尝试执行如下命令，重启ceph服务：

[root@node1 ceph-cluster]# systemctl restart ceph\\*.service ceph\\*.target

７，创建Ceph块存储

－－创建镜像

1）查看存储池。

[root@node1 ceph-cluster]# ceph osd lspools

0 rbd,

1. 创建镜像、查看镜像

两种创建镜像（共享硬盘）方式

方式一(没有指明存储池默认rbd)

[root@node1 ceph-cluster]# rbd create demo-image --image-feature layering --size 10G

方式二（指明存储池rbd）

[root@node1 ceph-cluster]# rbd create rbd/image --image-feature layering --size 10G

[root@node1 ceph-cluster]# rbd list　　 　 #rbd ls也可以查看镜像

demo-image

image

[root@node1 ceph-cluster]# rbd info demo-image　　＃查看具体某个镜像信息

rbd image 'demo-image':

size 10240 MB in 2560 objects

order 22 (4096 kB objects)

block\_name\_prefix: rbd\_data.d3aa2ae8944a

format: 2

features: layering

－－动态调整

1）缩小容量

[root@node1 ceph-cluster]# rbd resize --size 7G image --allow-shrink

[root@node1 ceph-cluster]# rbd info image

2）扩容容量

[root@node1 ceph-cluster]# rbd resize --size 15G image

[root@node1 ceph-cluster]# rbd info image

－－通过KRBD访问

1）集群内将镜像映射为本地磁盘

[root@node1 ceph-cluster]# rbd map demo-image＃使用内核将镜像映射到块设备

/dev/rbd0

[root@node1 ceph-cluster]# lsblk

… …

rbd0 251:0 0 10G 0 disk

[root@node1 ceph-cluster]# mkfs.xfs /dev/rbd0

[root@node1 ceph-cluster]# mount /dev/rbd0 /mnt

2）客户端通过KRBD访问

#客户端需要安装ceph-common软件包

#拷贝配置文件（否则不知道集群在哪）

#拷贝连接密钥（否则无连接权限）

[root@client ~]# yum -y install ceph-common

[root@client ~]# scp 192.168.4.11:/etc/ceph/ceph.conf /etc/ceph/　　＃从集群中下载配置文件

[root@client ~]# scp 192.168.4.11:/etc/ceph/ceph.client.admin.keyring \

/etc/ceph/　　　　　　＃从集群中下载密钥

[root@client ~]# rbd map image　　　＃使用内核将镜像映射到块设备

[root@client ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sr0 11:0 1 1024M 0 rom

vda 252:0 0 20G 0 disk

├─vda1 252:1 0 1G 0 part /boot

└─vda2 252:2 0 19G 0 part

├─rhel-root 253:0 0 17G 0 lvm /

└─rhel-swap 253:1 0 2G 0 lvm [SWAP]

rbd0 251:0 0 15G 0 disk 　　　　　　　＃此处是使用的网络磁盘

[root@client ~]# rbd showmapped　　　　　＃查看使用的网络镜像（磁盘）信息

id pool image snap device

0 rbd image - /dev/rbd0

3) 客户端格式化、挂载分区并使用

[root@client ~]# mkfs.xfs /dev/rbd0

[root@client ~]# mount /dev/rbd0 /mnt/

[root@client ~]# echo "test" > /mnt/test.txt

－－创建镜像快照

磁盘快照(Snapshot)是针对整个磁盘卷册进行快速的档案系统备份，与其它备份方式最主要的不同点在于「速度」。进行磁盘快照时，并不牵涉到任何档案复制动作。就算数据量再大，一般来说，通常可以在一秒之内完成备份动作。

磁盘快照的基本概念与磁带备份等机制有非常大的不同。在建立磁盘快照时，并不需要复制数据本身，它所作的只是通知LX Series NAS服务器将目前有数据的磁盘区块全部保留起来，不被覆写。这个通知动作只需花费极短的时间。接下来的档案修改或任何新增、删除动作，均不会覆写原本数据所在的磁盘区块，而是将修改部分写入其它可用的磁盘区块中。所以可以说，数据复制，或者说数据备份，是在平常档案存取时就做好了，而且对效能影响极低

1) 查看镜像快照

[root@node1 ceph-cluster]# rbd snap ls image

2) 创建镜像快照

[root@node1 ceph-cluster]# rbd snap create image --snap image-snap1

[root@node1 ceph-cluster]# rbd snap ls image

SNAPID NAME SIZE

4 image-snap1 15360 MB

3) 删除客户端写入的测试文件

[root@client ~]# rm -rf /mnt/test.txt

[root@client ~]# umount /mnt　　　先卸载再去还原快照，不然会报下面错误

4) 还原快照

[root@node1 ceph-cluster]# rbd snap rollback image --snap image-snap1

#客户端重新挂载分区

[root@client ~]# mount /dev/rbd0 /mnt/

[root@client ~]# ls /mnt

test.txt

快照还原问题：

[root@client ~]# mount /dev/rbd0 /mnt

mount: 文件系统类型错误、选项错误、/dev/rbd0 上有坏超级块、

缺少代码页或助手程序，或其他错误

有些情况下在 syslog 中可以找到一些有用信息- 请尝试

dmesg | tail 这样的命令看看

解决办法：

１，先在客户机卸载

２，在node１在还原快照

２，在客户机上挂载

－－创建快照克隆

1）克隆快照

[root@node1 ceph-cluster]# rbd snap protect image --snap image-snap1　＃把快照保护起来

[root@node1 ceph-cluster]# rbd snap rm image --snap image-snap1 //会失败

[root@node1 ceph-cluster]# rbd clone \

image --snap image-snap1 image-clone --image-feature layering

//使用image的快照image-snap1克隆一个新的image-clone镜像

2）查看克隆镜像与父镜像快照的关系

[root@node1 ceph-cluster]# rbd info image-clone

rbd image 'image-clone':

size 15360 MB in 3840 objects

order 22 (4096 kB objects)

block\_name\_prefix: rbd\_data.d3f53d1b58ba

format: 2

features: layering

flags:

parent: rbd/image@image-snap1

#克隆镜像很多数据都来自于快照链

#如果希望克隆镜像可以独立工作，就需要将父快照中的数据，全部拷贝一份，但比较耗时！！！

[root@node1 ceph-cluster]# rbd flatten image-clone＃解除与快照的关系

[root@node1 ceph-cluster]# rbd info image-clone

rbd image 'image-clone':

size 15360 MB in 3840 objects

order 22 (4096 kB objects)

block\_name\_prefix: rbd\_data.d3f53d1b58ba

format: 2

features: layering

flags:

#注意，父快照信息没了！

－－其他操作

1） 客户端撤销磁盘映射

[root@client ~]# umount /mnt

[root@client ~]# rbd showmapped

id pool image snap device

0 rbd image - /dev/rbd0

[root@client ~]# rbd unmap /dev/rbd/rbd/image　　　＃撤销磁盘映射image

1. 删除快照与镜像

[root@node1 ceph-deploy]# rbd snap unprotect image --snap image-snap1　＃解除保护

[root@node1 ceph-cluster]# rbd snap rm image --snap image-snap１

[root@node1 ceph-cluster]# rbd list

[root@node1 ceph-cluster]# rbd rm image

7，块存储应用案例

1）创建磁盘镜像。

[root@node1 ceph-cluster]# rbd create vm1-image --image-feature layering --size 10G

[root@node1 ceph-cluster]# rbd create vm2-image --image-feature layering --size 10G

[root@node1 ceph-cluster]# rbd list

[root@node1 ceph-cluster]# rbd info vm1-image

[root@node1 ceph-cluster]# qemu-img info rbd:rbd/vm1-image

image: rbd:rbd/vm1-image

file format: raw

virtual size: 10G (10737418240 bytes)

disk size: unavailable

2）Ceph认证账户。

Ceph默认开启用户认证，客户端需要账户才可以访问，

默认账户名称为client.admin，key是账户的密钥，

可以使用ceph auth添加新账户（案例我们使用默认账户）。

[root@node1 ceph-cluster]# cat /etc/ceph/ceph.conf //配置文件

[global]

mon\_initial\_members = node1, node2, node3

mon\_host = 192.168.2.10,192.168.2.20,192.168.2.30

auth\_cluster\_required = cephx //开启认证

auth\_service\_required = cephx //开启认证

auth\_client\_required = cephx //开启认证

[root@node1 ceph-cluster]# cat /etc/ceph/ceph.client.admin.keyring //账户文件

[client.admin]

key = AQBTsdRapUxBKRAANXtteNUyoEmQHveb75bISg==

3）部署客户端环境。

注意：这里使用真实机当客户端！！！

客户端需要安装ceph-common软件包，拷贝配置文件（否则不知道集群在哪），

拷贝连接密钥（否则无连接权限）。

[root@room9pc01 ~]# yum -y install ceph-common

[root@room9pc01 ~]# scp 192.168.4.11:/etc/ceph/ceph.conf /etc/ceph/

[root@room9pc01 ~]# scp 192.168.4.11:/etc/ceph/ceph.client.admin.keyring \

/etc/ceph/

4）创建KVM虚拟机。

使用virt-manager创建2台普通的KVM虚拟机。

5）配置libvirt secret。

编写账户信息文件（真实机操作）

[root@room9pc01 ~]# vim secret.xml //新建临时文件，内容如下

<secret ephemeral='no' private='no'>

<usage type='ceph'>

<name>client.admin secret</name>

</usage>

</secret>

#使用XML配置文件创建secret

[root@room9pc01 ~]# virsh secret-define --file secret.xml

生成 secret c409ec4c-803b-46bc-b4e4-35dc86a08f85

//随机的UUID，这个UUID对应的有账户信息

编写账户信息文件（真实机操作）

[root@room9pc01 ~]# ceph auth get-key client.admin

//获取client.admin的key，或者直接查看密钥文件

[root@room9pc01 ~]# cat /etc/ceph/ceph.client.admin.keyring

[client.admin]

key = AQDExOpbE+W9BxAA9QBqQUAY0tFmSmjOWmS8Nw==

设置secret，添加账户的密钥

[root@room9pc01 ~]# virsh secret-set-value --secret c409ec4c-803b-46bc-b4e4-35dc86a08f85 --base64 AQDExOpbE+W9BxAA9QBqQUAY0tFmSmjOWmS8Nw==

//这里secret后面是之前创建的secret的UUID

//base64后面是client.admin账户的密码

//现在secret中既有账户信息又有密钥信息

[root@room9pc01 room2018]# virsh secret-list

UUID 用量

--------------------------------------------------------------------------------

c409ec4c-803b-46bc-b4e4-35dc86a08f85 ceph client.admin secret

6）虚拟机的XML配置文件。

每个虚拟机都会有一个XML配置文件，包括：

虚拟机的名称、内存、CPU、磁盘、网卡等信息

[root@room9pc01 ~]# vim /etc/libvirt/qemu/vm1.xml

//修改前内容如下

<disk type='file' device='disk'>

<driver name='qemu' type='qcow2'/>

<source file='/var/lib/libvirt/images/vm1.qcow2'/>

<target dev='vda' bus='virtio'/>

<address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/>

</disk>

不推荐直接使用vim修改配置文件，推荐使用virsh edit修改配置文件，效果如下：

[root@room9pc01] virsh edit vm1 //vm1为虚拟机名称

<disk type='network' device='disk'>

<driver name='qemu' type='raw'/>

<auth username='admin'>

<secret type='ceph' uuid='c409ec4c-803b-46bc-b4e4-35dc86a08f85'/>

</auth>

<source protocol='rbd' name='rbd/vm1-image'>

<host name='192.168.4.11' port='6789'/>

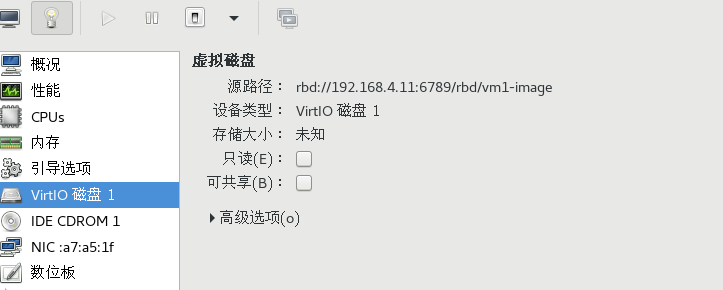
</source>

<target dev='vda' bus='virtio'/>

<address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/>

</disk>

提示：保存退出出现＂编辑了域 rhel7.4 XML 配置。＂说明成功了



源路经：rbd://192.168.4.11:6789/rbd/vm1-image

Ceph块存储总结：

１，Linux可以直接使用

２，Kvm虚拟机使用，修改kvm虚拟机xml配置文件

8，Ceph文件系统

1）添加一台新的虚拟机，要求如下：

IP地址:192.168.4.14

主机名:node4

配置yum源（包括rhel、ceph的源）

与真机主机同步时间（[root@node1 ceph-deploy]# scp /etc/chrony.conf 　　　　　root@192.168.4.14:/etc/）

node1允许无密码远程node4

[root@node1 ceph-deploy]#vim /etc/hosts #并拷贝到node1-4,client

192.168.4.10 client

192.168.4.11 node1

192.168.4.12 node2

192.168.4.13 node3

192.168.4.14 node4

2）部署元数据服务器

登陆node4，安装ceph-mds软件包

[root@node1 ceph-cluster]yum -y install ceph-mds

登陆node1部署节点操作

[root@node1 ceph-cluster]# cd /root/ceph-cluster　＃该目录，是最早部署ceph集群时，创建的目录

[root@node1 ceph-cluster]# ceph-deploy mds create node4　＃给nod4拷贝配置文件，启动mds服务

[root@node1 ceph-cluster]# ceph-deploy admin node4　　＃同步配置文件和key

1. 创建存储池

[root@node1 ceph-cluster]ceph osd pool create cephfs\_data 128　　＃创建存储池，对应128个PG

[root@node1 ceph-cluster]ceph osd pool create cephfs\_metadata 128　＃创建存储池，对应128个PG

4）创建Ceph文件系统

[root@node1 ceph-cluster]ceph mds stat //查看mds状态

e2:, 1 up:standby

[root@node1 ceph-cluster]ceph fs new myfs1 cephfs\_metadata cephfs\_data

new fs with metadata pool 2 and data pool 1

//注意，现在medadata池，再写data池

//默认，只能创建1个文件系统，多余的会报错

[root@node1 ceph-cluster]ceph fs ls

name: myfs1, metadata pool: cephfs\_metadata, data pools: [cephfs\_data ]

[root@node1 ceph-cluster]ceph mds stat

e4: 1/1/1 up {0=node4=up:creating}

5）客户端挂载

[root@client ~]# mount -t ceph 192.168.4.11:6789:/ /mnt/cephfs/ \

-o name=admin,secret=AQBTsdRapUxBKRAANXtteNUyoEmQHveb75bISg==

//注意:文件系统类型为ceph

//192.168.4.11为MON节点的IP（不是MDS节点）

//admin是用户名,secret是密钥

//密钥可以在/etc/ceph/ceph.client.admin.keyring中找到

9，创建对象存储服务器（RGW）

－－部署对象存储服务器

1）准备实验环境，要求如下：

IP地址:192.168.4.15

主机名:node5

配置yum源（包括rhel、ceph的源）

与真机主机同步时间

node1允许无密码远程node5

修改node1的/etc/hosts，并同步到所有node主机

2）部署RGW软件包

[root@node1 ceph-deploy]# ceph-deploy install --rgw node5　＃给node5安装软件包

[root@node1 ceph-deploy]# ceph-deploy admin node5　＃同步配置文件与密钥到node5

3）新建网关实例

启动一个rgw服务

[root@node1 ceph-deploy]# ceph-deploy rgw create node5

登陆node5验证服务是否启动

[root@node5 ~]# ps aux |grep radosgw

ceph 4109 0.2 1.4 2289196 14972 ? Ssl 22:53 0:00 /usr/bin/radosgw -f --cluster ceph --name client.rgw.node4 --setuser ceph --setgroup ceph

[root@node5 ~]# systemctl status ceph-radosgw@\\*

4）修改服务端口

登陆node5，RGW默认服务端口为7480，修改为8000或80更方便客户端记忆和使用

[root@node5 ~]# vim /etc/ceph/ceph.conf

[client.rgw.node5]

host = node5

rgw\_frontends = "civetweb port=80"

//node5为主机名

//civetweb是RGW内置的一个web服务

[root@node5 ~]# systemctl restart [ceph-radosgw@rgw.node4.service](mailto:ceph-radosgw@rgw.node4.service)　＃重启服务

[root@node4 ~]# ss -tunlp | grep :80　　　　＃查看端口80信息

tcp LISTEN 0 128 \*:80 \*:\* users:(("radosgw",pid=5994,fd=30))

－－客户端测试

1）curl测试

[root@client ~]# curl 192.168.4.15:8000

<?xml version="1.0" encoding="UTF-8"?><ListAllMyBucketsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/"><Owner><ID>anonymous</ID><DisplayName></DisplayName></Owner><Buckets></Buckets></ListAllMyBucketsResult>

2）使用第三方软件访问

登陆node5（RGW）创建账户

[root@node5 ~]# radosgw-admin user create \

--uid="testuser" --display-name="First User"

… …

"keys": [

{

"user": "testuser",

"access\_key": "5E42OEGB1M95Y49IBG7B",

"secret\_key": "i8YtM8cs7QDCK3rTRopb0TTPBFJVXdEryRbeLGK6"

}

],

... ...

#

[root@node5 ~]# radosgw-admin user info --uid=testuser

//testuser为用户，key是账户访问密钥

3）客户端安装软件

[root@client ~]# yum install s3cmd-2.0.1-1.el7.noarch.rpm

修改软件配置

[root@client ~]# s3cmd –configure

Access Key: 5E42OEGB1M95Y49IBG7B  
Secret Key: i8YtM8cs7QDCK3rTRopb0TTPBFJVXdEryRbeLGK6

S3 Endpoint [s3.amazonaws.com]: 192.168.4.15:8000

[%(bucket)s.s3.amazonaws.com]: %(bucket)s.192.168.4.15:8000

Use HTTPS protocol [Yes]: No

Test access with supplied credentials? [Y/n] Y

Save settings? [y/N] y

//注意，其他提示都默认回车

4）创建存储数据的bucket（类似于存储数据的目录）

[root@client ~]# s3cmd ls

[root@client ~]# s3cmd mb s3://my\_bucket

Bucket 's3://my\_bucket/' created

[root@client ~]# s3cmd ls

2018-05-09 08:14 s3://my\_bucket

[root@client ~]# s3cmd put /var/log/messages s3://my\_bucket/log/

[root@client ~]# s3cmd ls

2018-05-09 08:14 s3://my\_bucket

[root@client ~]# s3cmd ls s3://my\_bucket

DIR s3://my\_bucket/log/

[root@client ~]# s3cmd ls s3://my\_bucket/log/

2018-05-09 08:19 309034 s3://my\_bucket/log/messages

测试下载功能

[root@client ~]# s3cmd get s3://my\_bucket/log/messages /tmp/

测试删除功能

[root@client ~]# s3cmd del s3://my\_bucket/log/messages