**几种存储方法ceph的磁盘共享和文件系统共享**

块存储应用

块存储应用案例

Ceph创建**块存储**镜像

客户端安装部署ceph软件

客户端部署虚拟机

客户端创建secret

设置虚拟机配置文件，调用存储

**创建KVM虚拟机**

virsh list --all 列出本机上的虚拟机

在管理主机node51上创建安装虚拟的镜像文件vm-image1

rbd create vm1-image --image-feature layering --size 10G

rbd list

vm1-image

#虚拟机的镜像文件：

<source file=”/var/libvirt/...” />

vim /etc/ceph/ceph.conf //集群的配置文件

[global]

fsid = 88259769-e651-4a63-a9a3-831615e035ef

mon\_initial\_members = node51, node52, node53

mon\_host = 192.168.4.51,192.168.4.52,192.168.4.53

auth\_cluster\_required = cephx //访问机群验证

auth\_service\_required = cephx //访问服务验证

auth\_client\_required = cephx //客户端验证

vim /etc/ceph/ceph.client.admin.keyring //客户端访问密钥匙文件

[client.admin]

key = AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

真机调用ceph集群的储存空间来创建LVM虚拟机

需要使用集群的存储空间，需要安装ceph-common和集群的配置文件

客户端访问密钥匙文件

node5:scp /etc/ceph/ceph.conf ceph.client.admin.keyring

192.168.4.254:/etc/ceph/

真机连接ceph集群，是为了再ceph集群的存储上装KVM虚拟机

配置libvirt secret（libvirt 密钥） 红帽内置可以来年接ceph集群，但是需要账户验证

LVM虚拟机需要使用librbd才可以访问ceph集群

librbd访问ceph又需要账户认证

所以这里需要给libvirt设置账户信息

libvirt secret 是真机连接ceph集群的工具（必须要有）

**配置libvirt secret //在真机上操作**

1.自己书写secret.xml （编写账户信息文件）

vim /root/secret.xml

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

<usage type='ceph'>

<name>client.admin secret</name>

</usage> //admin为ceph端提供的访问账户

</secret>

2.使用xml配置文件创建secret

virsh secret-define --file /root/secret.xml //生成uuid

b7bd674d-d1eb-4e42-bc4b-56762dfb892a //这个是随即生成

Virsh secret-list //列出 secret

virsh secret-undefine 06f0c275-0641-4a84-84b2-0d2d2c3ab4c6

删除已有的UUID ,备用，

3.获取密钥

cat /etc/ceph/ceph.client.admin.keyring //该文件为ceph端地提

供的密钥

[client.admin] //admin为客户端访问的用户名

key=AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

或者ceph auth get-key client.admin

AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

4.设置secret，添加账户密钥

virsh secret-set-value --secret

b7bd674d-d1eb-4e42-bc4b-56762dfb892a --base64

AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

//以上是一条命令，secret后面是之前创建的secret的UUID

base64后面是client。Admin账户的密码（加密）

现象secret中既有账户信息又有密钥信息

5.图像化新建avpc的虚拟机，一切默认就好

所有的虚拟机的xml配置文件都再/etc/libvirt/qemu/下

- 包括：虚拟机的名称、内存、CPU、磁盘、网卡等信息

- 修改的硬件配置文件，定义装机用的硬盘

**使用图形管理工具创建的虚拟机文件，定义一个新的虚拟机xml文件**

vitsh dumpxml avpc //输出avpc的硬件配置，XML 中的域信息

virsh dumpxml avpc > /tmp/avpc1.xml //输出到新的文件中

vim /tmp/avpc1.xml

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

//磁盘类型改为网络，这个<disk>为<device>里的第一个

<driver name='qemu' type='raw'/> //设备改为读写

<auth username='admin'>

<secret

type='ceph' uuid='b7bd674d-d1eb-4e42-bc4b-56762dfb892a'/>

//这里的uuid就是secret的uuid，有client.amin账户和密钥信息

这个uuid是第四步生成的uuid

</auth>

<source protocol='rbd' name=**'rbd/vm1-image**'> //**块存储应用**

//协议使用rbd，访问哪个**池和镜像**

<host name='192.168.4.51' port='6789'/>

//这里说明使用账户连接哪台ceph主机和端口

</source>

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

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

</disk>

6.使用第5步配置文件创建一个新的使用rbd集群存储系统的虚拟机

第一台虚拟机：virsh define /tmp/avpc1.xml

定义域 avpc（从 /tmp/avpc1.xml）

调试虚拟机引导顺序

第二台：

ceph管理主机（node51）上新创建镜像

rbd create vm2-image --image-feature layering --size 10G

rbd list

vm2-image

1)新建vim /root/secret1.xml

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

<usage type='ceph'>

<name>client**1**.admin secret</name>

</usage> //admin为ceph端提供的访问账户

</secret>

2)使用xml配置文件创建secret

virsh secret-define --file /root/secret.xml //生成访问ceph时的uuid

b7bd674d-d1eb-4e42-bc4b-56762dfb892n //这个是随即生成

3)获取密钥

ceph auth get-key client.admin

AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

4)设置secret，添加账户密钥

virsh secret-set-value --secret

e83a0a90-090a-4d19-b032-08bd81f9e30c --base64

AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

5)vim /root/xiao.xml

删除第三行**<uuid></uuid> //必须执行**

修改第二行**虚拟机的名字** <name>xiao</name>

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

<driver name='qemu' type='raw'/> //设备改为读写

<auth username='admin'>

<secret

type='ceph' uuid='e83a0a90-090a-4d19-b032-08bd81f9e30c'/>

</auth>

<source protocol='rbd' name=**'rbd/vm2-image**'> //**块存储应用**

<host name='192.168.4.51' port='6789'/>

</source>

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

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

</disk>

6)使用第5步配置文件创建一个新的使用rbd集群存储系统的虚拟机

第二台虚拟机执行命令：

virsh define /tmp/xiao.xml

定义域 xiao（从 /tmp/xiao.xml）

调试虚拟机引导顺序

**分布式文件系统**

文件系统存储需要元数据服务器，用来存储元数据

元数据

包括inode和block

ls -i /etc/passwd //-i显示inode

任何文件系统中的数据分为数据和元数据

数据是指普通文件中的实际数据

而元数据指用来描述一个文件的特征的系统数据

比如：访问权限、文件拥有者以及文件数据块的分布信息（inode）等

**cephFS**

**分布式文件系统**是只文件系统管理物理存储资源管理的物理存储资源比已定直接来年接在本地节点上，而是通过计算机网络与节点相连

CephFS使用ceph集群提供与POSIX兼容的文件系统

允许linux直接将ceph存储挂载（mount）到本地

准备：增加node54：

Ip地址：192.168.4.54

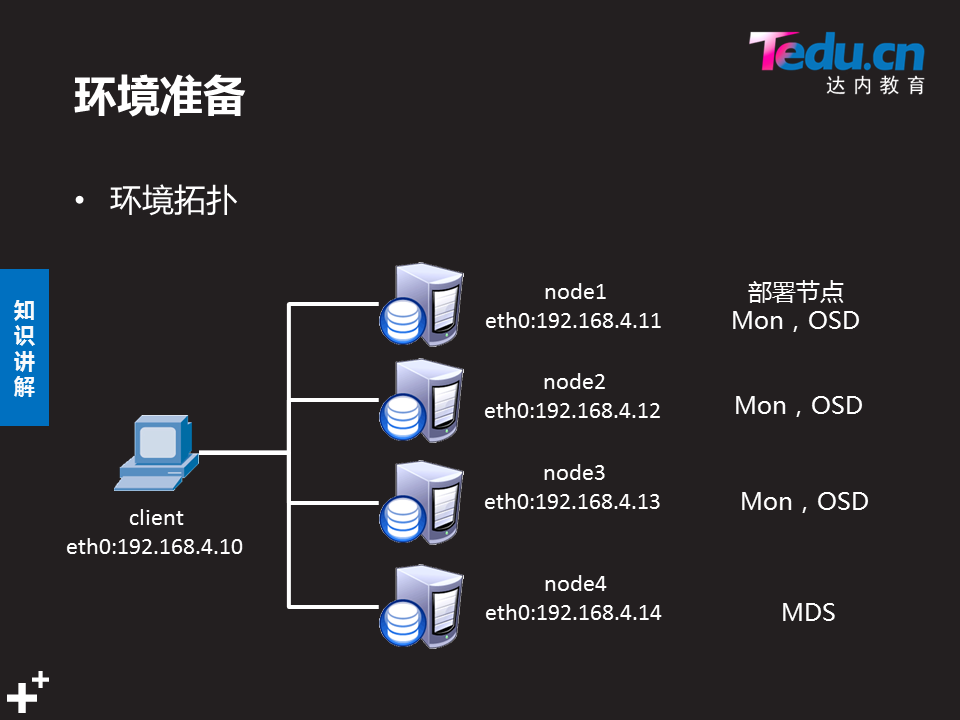
主机名：node54

配置yum源包括rhel、ceph源）

与client50同步时间

Node51无密码远程node54

修改node1的/etc/hosts，同步到所有的node主机



部署元数据服务器

登陆node54，安装ceph-mds软件包

Yum -y install ceph-mds

登陆node51

cd /root/ceph-cluster

ceph-deploy mds create node54

//给node54拷贝配置文件，启动mds服务

同步配置文件和key

ceph-deploy admin node54

登陆node54，查看服务启动状态，必须是启动的

systemctl status ceph-mds.target

创建存储池，在node54上操作

这个池用于存储数据，对应128个PG

ceph osd pool create cephfs\_data 128

>pool 'cephfs\_data' created

这个池用于存储元数据，对应128个PG

ceph osd pool create cephfs\_meatedata 128

>pool 'cephfs\_meatedata' created

创建文件系统

ceph mds stat //查看mds状态

e2:, 1 up:standby

利用上面的两个池创建一个文件系统，注意，先写mdatedate，再写data

ceph fs new myfs1 cephfs\_meatedata cephfs\_data

> new fs with metadata pool 2 and data pool 1 //myfs1 是文件系 统的名字

ceph mds stat

> e5: 1/1/1 up {0=node54=up:active}

客户端client50挂载,在client50上操作

yum -y install ceph-common.x86\_64

scp node1:/etc/ceph/ceph.client.admin.keyring \

node1:/etc/ceph/ceph.conf /etc/ceph/ //拷贝配置文件和密钥文件

mkdir /cephfs

cat /etc/ceph/ceph.client.admin.keyring //查看密钥文件

[client.admin] //admin，ceph访问帐号； key，ceph访问密码

key = AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

mount -t ceph 192.168.4.51:6789:/ /cephfs/ -o name=admin,

secret=AQDXvb5bWZrfHxAAM85iWlnV/UaCDyhBzgLwCg==

//挂载

查看挂载

df -h

192.168.4.51:6789:/ 60G 4.8G 56G 8% /cephfs

**对象存储；**默认端口为7480

准备实验环境，要求如下：

IP地址:192.168.4.55

主机名:node55

配置yum源（包括rhel、ceph的源）

与client主机同步时间

node51允许无密码远程node55

部署RGW软件包,在node51上操作

ceph-deploy install --rgw node55

ceph-deploy admin node55

ceph-deploy rgw create node55

ps aux | grep -v grep | grep radosgw

登陆node55，RGW默认web服务端口为7480修改为80方便客户端访问

vim /etc/ceph/ceph.conf //追加以下内容，访问不了就放在文件开始

[client.rgw.node55]

host=node55

rgw\_frontends = "civetweb port=80"

测试访问：

curl <http://192.168.4.55> //结果是一端代码

**使用第三方软件访问**

创建客户端访问时的账户和密码

[root@node55 ~]# radosgw-admin user create --uid="testuser" --display-name="First user"

显示用户密钥信息，在node55上操作 //testuser为用户，key是账户访问密钥

[root@node55 ~]# eradosgw-admin user info --uid=testuser

"user": "testuser",

"access\_key": "SXAK5K43KQHB8YIQO4H8",

"secret\_key": "8iy9I6p6onzbJtJgkC1qrZ9sndcrUZ2c1ZE9hZLy"

客户端安装访问网关主机node55的工具（命令行访问工具）

[root@node51 ~]# rpm -ivh s3cmd-2.0.1-1.el7.noarch.rpm

配置软件，Access Key和Secret Key的值是在node55上

s3cmd --configure

Access Key: SXAK5K43KQHB8YIQO4H8

Secret Key: 8iy9I6p6onzbJtJgkC1qrZ9sndcrUZ2c1ZE9hZLy

S3 Endpoint [s3.amazonaws.com]: 192.168.4.55:80

[%(bucket)s.s3.amazonaws.com]: %(bucket)s.192.168.4.55:80

Use HTTPS protocol [Yes]: No

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

Save settings? [y/N] y

创建存储数据的buckket（类似与存储数据的目录）

s3cmd mb s3://my\_bucket

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

上传文件/到某个指定的文件夹，上传不用创建文件夹，他会自动创建

s3cmd put /etc/passwd s3://my\_bucket/log/

查看文件夹

s3cmd ls s3://my\_bucket/log

查看文件夹的内容

s3cmd ls s3://my\_bucket/log/