Ceph 基础篇

1. Ceph 介绍

1.1 Ceph 介绍

在过去几年中，数据存储需求急剧增长。研究表明，大型组织中的数据正以每年 40%到 60%的速度增长，许多公司每 年的数据都翻了一番。国际数据公司(IDC)的分析师估计，到 2000 年，全球共有 54.4 exabytes 的数据。到 2007 年，这 一数字达到 295 艾字节，到 2020 年，全球预计将达到 44 zettabytes。传统的存储系统无法管理这样的数据增长;我们需要一个像 Ceph 这样的系统，它是分布式的，可扩展的，最重要的是，在经济上是可行的。Ceph 是专门为处理当今和未来的数据存储需求而设计的。

1ZB=1024EB 1EB=1024PB 1PB=1024TB

(1) 软件定义存储 -SDS

SDS 是减少存储基础设施的 TCO(总体成本)所需要的。除了降低存储成本外，SDS 还可以提供灵活性、可伸缩性和可靠 性。Ceph 是一种真正的 SDS;它运行在没有厂商锁定的普通硬件上。与传统的存储系统(硬件与软件结合在一起)不同， 在 SDS 中，您可以从任何制造商中自由选择硬件，也可以根据自己的需要自由设计异构硬件解决方案。Ceph 在此硬件 之上的软件定义存储提供了您需要的所有，并将负责所有事情，从软件层提供了所有企业存储特性。

（2）云存储

目前已经和开源云架构 OpenStack 结合起来，成为 Openstack 后端存储的标配，并且又同时支持用于 kubernetes 动态 存储。

（3）下一代统一存储体系架构

统一存储的定义最近发生了变化。几年前，术语“统一存储”指从单个系统提供文件和块存储。如今，由于近年来的技术进步，如云计算、大数据和物联网，一种新的存储方式正在进化，即对象存储。因此，所有不支持对象存储的存储系 统都不是真正的统一存储解决方案。真正的统一存储就像 Ceph;它支持来自单个系统的块、文件和对象存储。

Ceph 是目前最热门的软件定义存储(SDS)技术，正在撼动整个存储行业。它是一个开源项目，为块、文件和对象存储提供统一的软件定义的解决方案。Ceph 的核心思想是提供一个分布式存储系统，该系统具有大规模的可伸缩性和高性能，并且没有单点故障。从根本上说，它被设计成在通用硬件上运行时具有高度的可伸缩性(可达艾字节（ExaByte）级别甚至更高)。

Ceph 提供了出色的性能、巨大的可伸缩性、强大的功能和灵活性。它摆脱昂贵的专有存储。Ceph 确实是一个企业级的 存储解决方案，可以在普通硬件上运行;这是一个低成本但功能丰富的存储系统。Ceph 的通用存储系统提供块存储、文 件存储和对象存储，使客户可以随心所欲地使用存储。

Ceph 正在快速发展和改进，目前发布了十三个版本，每个长期版本都有一个名称,该名称遵循字母顺序发行。

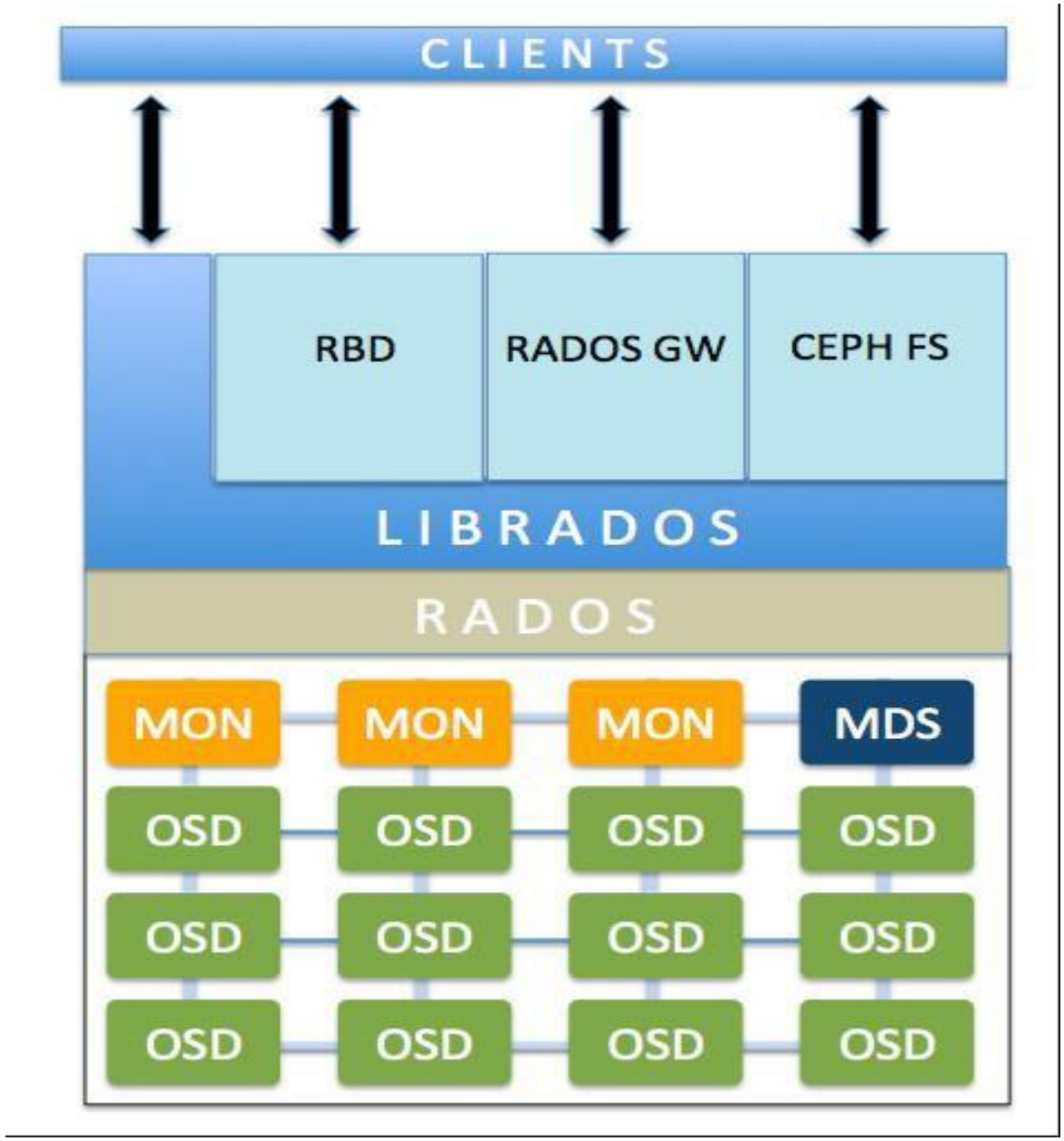
Ceph 的吉祥物是章鱼。

Ceph 版本名称	Ceph 版本号	Ceph 发行日期
Argonaut	V0.48(LTS)	2012/7/3
Bobtail	V0.56(LTS)	2013/1/1
Cuttlefish	V0.61	2013/5/7
Dumpling	V0.67(LTS)	2013/8/14
Emperor	V0.72	2013/11/9
Firefly	V0.80(LTS)	2013/5/7
Giant	V0.87.1	2015/2/26
Hammer	V0.94(LTS)	2015/4/7
Infernalis	V9.0.0	2015/5/5
Jewel	V10.0.0(LTS)	2015/11
Kraken	V11.0.0	2016/6
Luminous	V12.0.0(LTS)	2017/8
Mimic	V13.2.0	2018/6

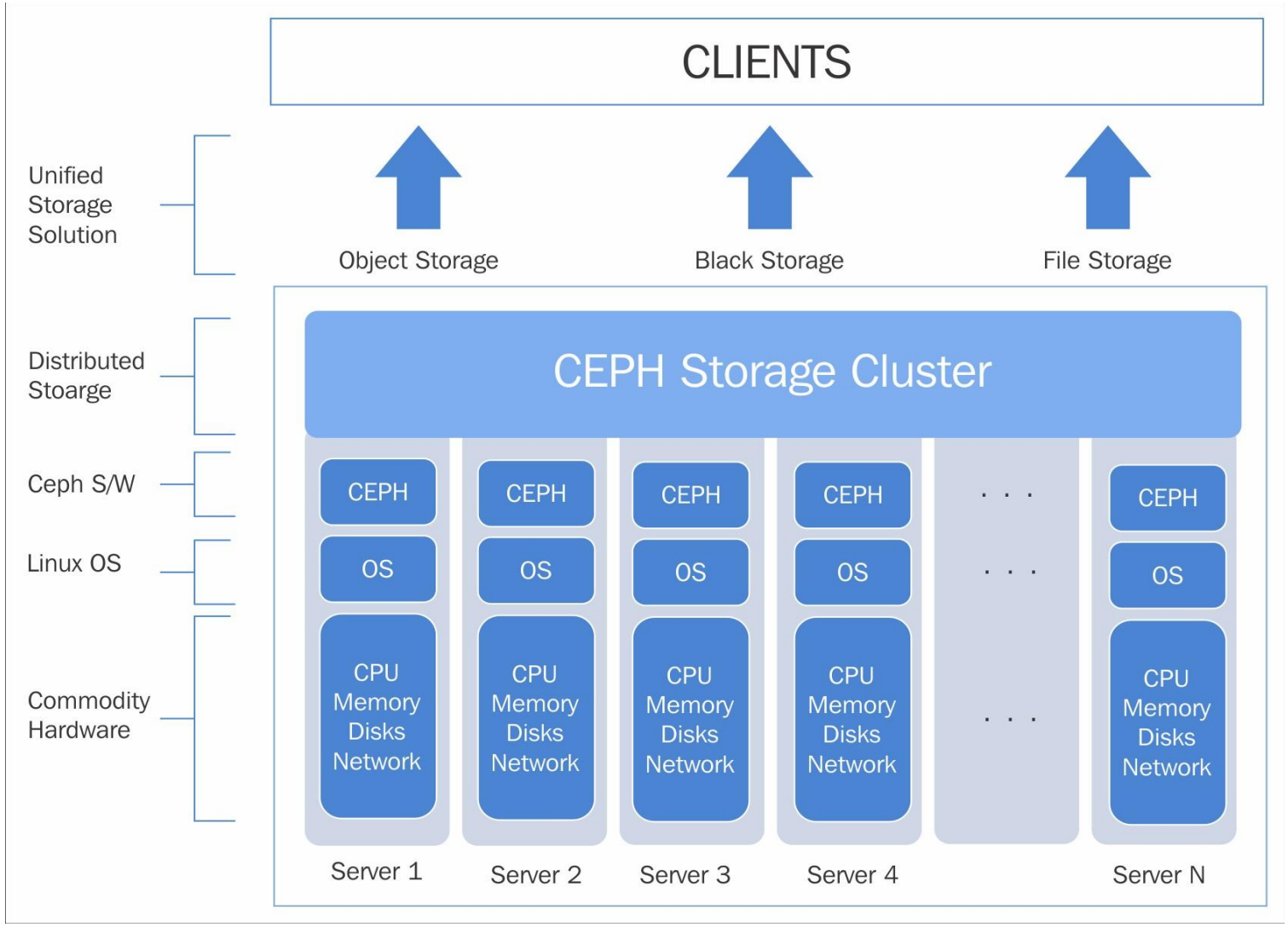
1.2 Ceph 架构

（1）Ceph 组件

- Ceph monitors(MON)：Ceph 监视器通过保存集群状态的映射来跟踪整个集群的健康状况
- Ceph 对象存储设备(OSD)： 一旦应用程序向 Ceph 集群发出写操作，数据就以对象的形式存储在 OSD 中。
  - 这是 Ceph 集群中存储实际用户数据的惟一组件，通常，一个 OSD 守护进程绑定到集群中的一个物理磁 盘。因此，通常来说，Ceph 集群中物理磁盘的总数与在每个物理磁盘上存储用户数据的 OSD 守护进程 的总数相同。
- Ceph metadata server (MDS): MDS 跟踪文件层次结构，仅为 Ceph FS 文件系统存储元数据
- RADOS: RADOS 对象存储负责存储这些对象，而不管它们的数据类型如何。RADOS 层确保数据始终保持一致。为此，它执行数据复制、故障检测和恢复，以及跨集群节点的数据迁移和再平衡。
- Librados: librados 库是一种访问 RADOS 的方便方法，支持 PHP、Ruby、Java、Python、C 和 c++编程语言。它为 Ceph 存储集群(RADOS)提供了本机接口，并为其他服务提供了基础，如 RBD、RGW 和 CephFS，这些服务构建 在 librados 之上。librados 还支持从应用程序直接访问 RADOS，没有 HTTP 开销。
- RBD：提供持久块存储，它是瘦配置的、可调整大小的，并在多个 osd 上存储数据条带。RBD 服务被构建为 一个在 librados 之上的本机接口。
- RGW：RGW 提供对象存储服务。它使用 librgw (Rados 网关库)和 librados，允许应用程序与 Ceph 对象存储建立 连接。RGW 提供了与 Amazon S3 和 OpenStack Swift 兼容的 RESTful api 接口。
- CephFS: Ceph 文件系统提供了一个符合 posix 标准的文件系统，它使用 Ceph 存储集群在文件系统上存储用户数 据。与 RBD 和 RGW 一样，CephFS 服务也作为 librados 的本机接口实现。
- Ceph manager: Ceph manager 守护进程(Ceph -mgr)是在 Kraken 版本中引入的，它与 monitor 守护进程一起运行，为外部监视和管理系统提供额外的监视和接口。

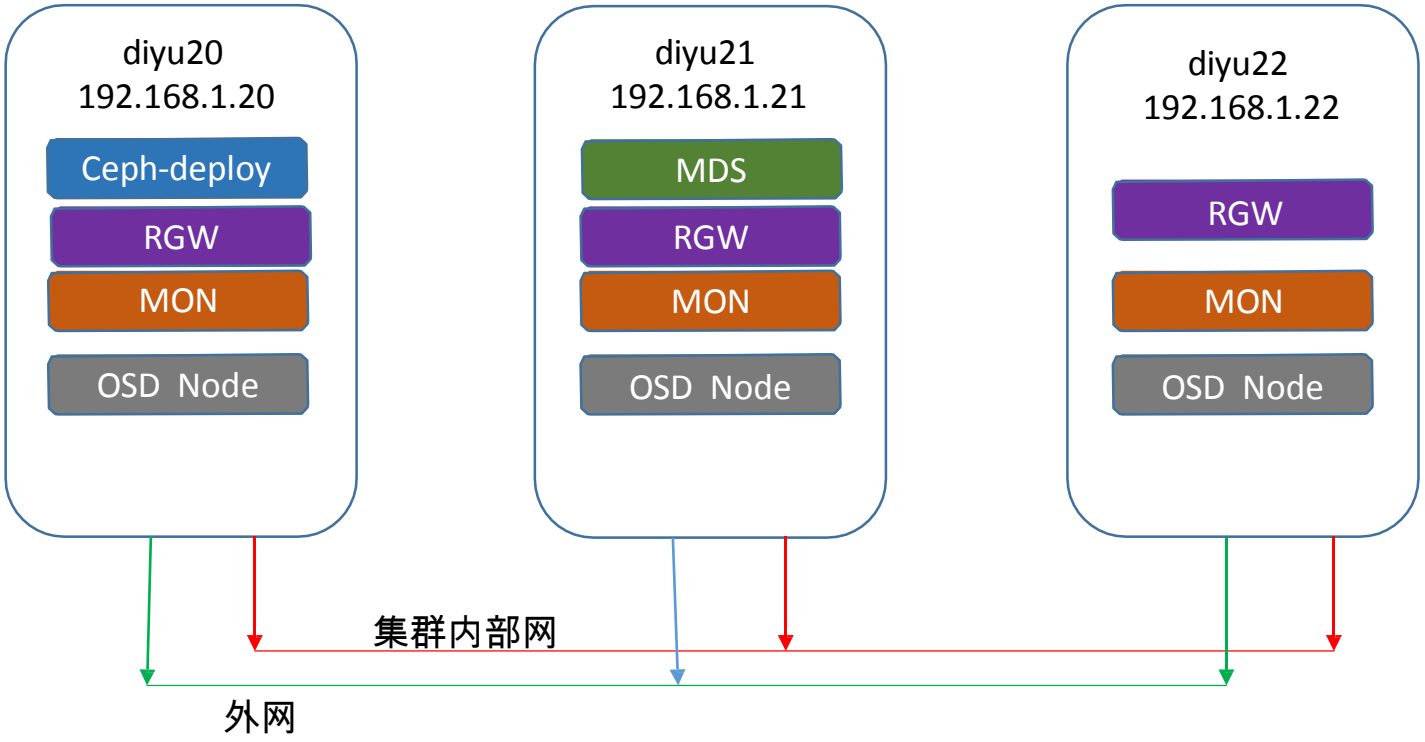

CLIENTS
RBD
RADOS GW
CEPH FS
LIBRADOS
RADOS
MON
MON
MON
MDS
OSD
OSD
OSD
OSD
OSD
OSD
OSD
OSD
OSD
OSD
OSD
OSD

（2）Ceph 部署


CLIENTS
Unified Storage Solution
Object Storage
Black Storage
File Storage
Distributed Stoarge
CEPH Storage Cluster
Ceph S/W
Linux OS
Commodity Hardware
CEPH
OS
CPU
Memory
Disks
Network
Server 1
Server 2
Server 3
Server 4
Server N

2. Ceph 部署

2.1 Ceph 安装前准备



```
export username="ceph-admin"
export passwd="ceph-admin"
export  node1="diyu20"
export  node2="diyu21"
export node3="diyu22"


# 配置 rpm
vi /etc/yum.repos.d/ceph.repo

[ceph]
name=ceph
baseurl=http://mirrors.aliyun.com/ceph/rpm-luminous/el7/x86_64/
gpgcheck=0
priority=1

[ceph-noarch]
name=cephnoarch
baseurl=http://mirrors.aliyun.com/ceph/rpm-luminous/el7/noarch/
gpgcheck=0
priority=1

[ceph-source]
name=Ceph source packages
baseurl=http://mirrors.aliyun.com/ceph/rpm-luminous/el7/SRPMS
enabled=0
gpgcheck=1
type=rpm-md
```

```
gpgkey=http://mirrors.aliyun.com/ceph/keys/release.asc
priority=1
```

配置 NTP

```
yum -y install ntpdate ntp
ntpdate  cn.ntp.org.cn
systemctl restart ntpd ntpdate && systemctl enable ntpd ntpdate
```

创建部署用户和 ssh 免密码登录

```
useradd ${username}
echo "${passwd}" | passwd --stdin ${username}
echo "${username} ALL = (root) NOPASSWD:ALL" | sudo tee /etc/sudoers.d/${username}
chmod 0440 /etc/sudoers.d/${username}
```

配置防火墙，或者关闭
#systemctl disable firewalld
#systemctl stop firewalld
SELINUX 设置为 disable
vi /etc/selinux/config

SELINUX=disabled
setenforce 0# 关闭 selinux
sed -i "/^SELINUX/s/enforcing/disabled/" /etc/selinux/config setenforce 0

配置主机名解析，使用 /etc/hosts,或者 dns
vi /etc/hosts
192.168.1.20 diyu20
192.168.1.21 diyu21
192.168.1.22 diyu22

配置 sudo 不需要 tty
sed -i 's/Default requiretty/ #Default requiretty/' /etc/sudoers

2.2 使用 ceph-deploy 部署集群

 # 配置免密钥登录

```
su - ceph-admin
export username=ceph-admin
ssh-keygen
su - ceph-admin
ssh-keygen
ssh-copy-id ceph-admin@diyu20
ssh-copy-id ceph-admin@diyu21
ssh-copy-id ceph-admin@diyu22
```

 # 安装 ceph-deploy

```
sudo yum install -y  ceph-deploy python-pip
```

#建立目录

```
mkdir  my-cluster
cd my-cluster
```

 # 部署节点

```
ceph-deploy  new  diyu20 diyu21 diyu22
```

ls

编辑 ceph.conf 配置文件

cat ceph.conf

[global]

.....

public network = 192.168.20.0/24

cluster network = 192.168.20.0/24

#

ceph-deploy install diyu20 diyu21 diyu22

或者

安装 ceph 包，替代 ceph-deploy install node1 node2 ,不过下面的命令需要在每台 node 上安装（推荐 快）

sudo yum install -y yum-utils

sudo yum install --nogpgcheck -y epel-release

sudo yum install -y ceph ceph-radosgw

配置初始 monitor(s)、并收集所有密钥：

ceph-deploy mon create-initial

ls -l *.keyring

把配置信息拷贝到各节点

ceph-deploy admin diyu20 diyu21 diyu22

配置 osd

lsblk

ceph-deploy disk zap diyu20 /dev/sdb

```
ceph-deploy osd create  diyu20 --data /dev/sdb
ceph-deploy osd create  diyu20 --data /dev/sdc
ceph-deploy osd create  diyu20 --data /dev/sdd

ceph-deploy osd create  diyu21 --data /dev/sdb
ceph-deploy osd create  diyu21 --data /dev/sdc
ceph-deploy osd create  diyu21 --data /dev/sdd

ceph-deploy osd create  diyu22 --data /dev/sdb
ceph-deploy osd create  diyu22 --data /dev/sdc
ceph-deploy osd create  diyu22 --data /dev/sdd

ceph-deploy mgr create diyu20 diyu21 diyu22

sudo chmod 755 /etc/ceph/ceph.client.admin.keyring

ceph mgr module enable dashboard
```

部署 mgr ，L 版以后才需要部署
ceph-deploy mgr create diyu20 diyu21 diyu22

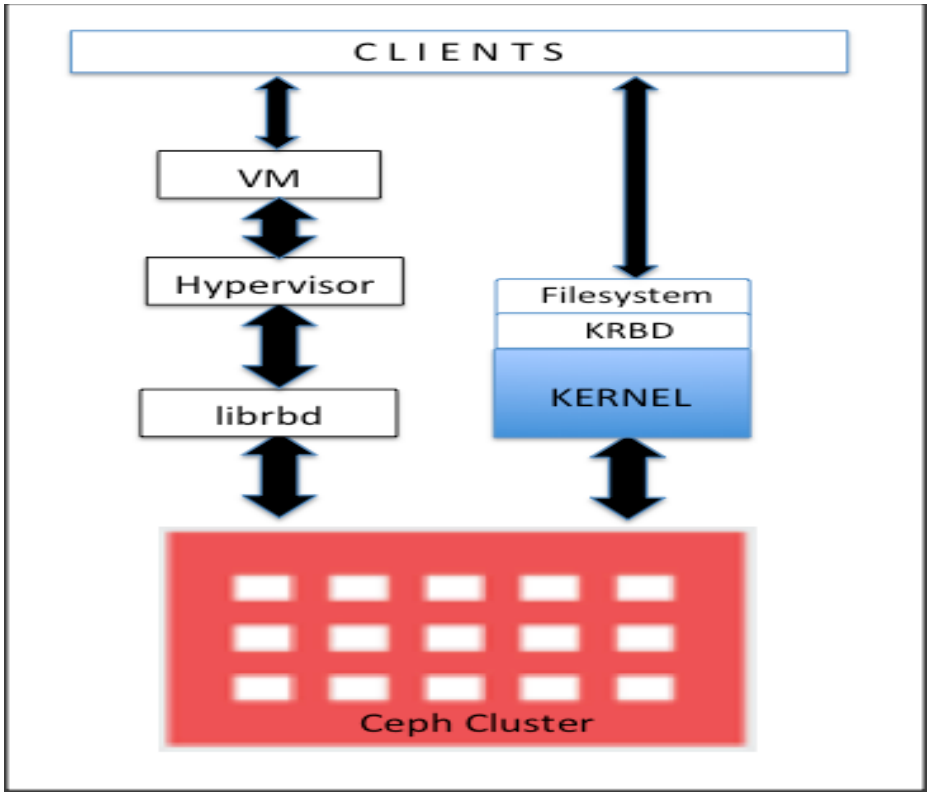
开启 dashboard 模块，用于 UI 查看
ceph mgr module enable dashboard

curl http://192.168.1.20:7000

3. Ceph 块存储

3.1 安装 Ceph 块存储客户端

Ceph 块设备，以前称为 RADOS 块设备，为客户机提供可靠的、分布式的和高性能的块存储磁盘。RADOS 块设备利用 librbd 库并以顺序的形式在 Ceph 集群中的多个 osd 上存储数据块。RBD 是由 Ceph 的 RADOS 层支持的，因此每个块设备 都分布在多个 Ceph 节点上，提供了高性能和优异的可靠性。RBD 有 Linux 内核的本地支持，这意味着 RBD 驱动程序从 过去几年就与 Linux 内核集成得很好。除了可靠性和性能之外，RBD 还提供了企业特性，例如完整和增量快照、瘦配 置、写时复制克隆、动态调整大小等等。RBD 还支持内存缓存，这大大提高了其性能:



任何普通的 Linux 主机(RHEL 或基于 debian 的)都可以充当 Ceph 客户机。客户端通过网络与 Ceph 存储集群交互以存储或检 索用户数据。Ceph RBD 支持已经添加到 Linux 主线内核中，从 2.6.34 和以后的版本开始。

****

参考
https://blog.csdn.net/litianze99/article/details/44624451

列出所有 ceph 用户：
ceph auth list

获取指定用户
ceph auth get {TYPE.ID}

****

创建 ceph 块客户端用户名和认证密钥
ceph auth get-or-create client.rbd mon 'allow r' osd 'allow class-read object_prefix rbd_children, allow rwx pool=rbd'|tee ./ceph.client.rbd.keyring

客户端(设置成 1.12)
mkdir /etc/ceph -p

scp ceph.client.rbd.keyring /etc/ceph/ceph.conf root@192.168.1.12:/etc/ceph/
scp ceph.client.rbd.keyring /etc/ceph/ceph.conf root@192.168.1.12:/etc/ceph/

手工把密钥文件拷贝到客户端

检查是否符合块设备环境要求
uname -r
modprobe rbd
echo $?

需要注意：
1、linux 内核从
2.6.32 版本开始
支持 ceph
2、建议使用
2.6.34 以及以上
的内核版本

安装 ceph 客户端
vi /etc/yum.repos.d/ceph.repo
同上

yum -y install ceph
cat /etc/ceph/ceph.client.rbd.keyring
ceph -s --name client.rbd 测试联通

3.2 客户端创建块设备及映射

(1) 创建块设备

默认创建块设备，会直接创建在 rbd 池中，但使用 deploy安装后，该 rbd 池并没有创建。

```
# 创建池和块
ceph osd lspools                    # 查看集群存储池
ceph osd pool create rbd 50         # 50 为 place group 数量，由于我们后续测试，也需要更多的 pg,所以这里设置为 50
```

确定 pg_num 取值是强制性的，因为不能自动计算。下面是几个常用的值：

- 少于 5 个 OSD 时可把 pg_num 设置为 128
- OSD 数量在 5 到 10 个时，可把 pg_num 设置为 512
- OSD 数量在 10 到 50 个时，可把 pg_num 设置为 4096
- OSD 数量大于 50 时，你得理解权衡方法、以及如何自己计算 pg_num取值

```
#  客户端创建 块设备
rbd  create rbd1  --size  10240  --name client.rbd

rbd ls  --name  client.rbd
rbd ls -p rbd --name client.rbd
rbd list  --name  client.rbd

rbd --image  rbd1 info  --name client.rbd
```

##########
映射到客户端
rbd map --image rbd1 --name client.rbd
报错后
rbd: sysfs write failed
RBD image feature set mismatch. You can disable features unsupported by the kernel with "rbd feature disable rbd1 object-map fast-diff deep-flatten".
layering: 分层支持
exclusive-lock: 排它锁定支持对
object-map: 对象映射支持(需要排它锁定(exclusive-lock))
deep-flatten: 快照平支持(snapshot flatten support)
fast-diff: 在 client-node1 上使用 krbd(内核 rbd)客户机进行快速 diff 计算(需要对象映射)，我们将无法在 CentOS 内核 3.10 上映射块设备映像，因为该内核不支持对象映射(object-map)、深平(deep-flatten)和快速 diff(fast-diff)(在内核 4.9 中引入了支持)。为了解决这个问题，我们将禁用不支持的特性，有几个选项可以做到这一点:
1）动态禁用
rbd feature disable rbd1 exclusive-lock object-map deep-flatten fast-diff --name client.rbd
2）创建 RBD 镜像时，只启用分层特性。
rbdcreate rbd2 --size 10240 --image-feature layering --name client.rbd
3）ceph 配置文件中禁用
rbd_default_features= 1
##########

（2）映射块设备

映射到客户端，应该会报错

rbd map --image rbd1 --name client.rbd

layering: 分层支持

exclusive-lock: 排它锁定支持对

object-map: 对象映射支持(需要排它锁定(exclusive-lock))

deep-flatten: 快照平支持(snapshot flatten support)

• fast-diff: 在 client-node1 上使用 krbd(内核 rbd)客户机进行快速 diff 计算(需要对象映射)，我们将无法在 CentOS 内核 3.10 上映射块设备映像，因为该内核不支持对象映射(object-map)、深平(deep-flatten)和快速 diff(fast-diff)(在内核 4.9 中引 入了支持)。为了解决这个问题，我们将禁用不支持的特性，有几个选项可以做到这一点:

1）动态禁用（推荐）

rbd feature disable rbd1 exclusive-lock object-map deep-flatten fast-diff --name client.rbd

2） 创建 RBD 镜像时，只启用 分层特性。

rbd create rbd2 --size 10240 --image-feature layering --name client.rbd

3）ceph 配置文件中禁用

rbd_default_features = 1

我们这里动态禁用
rbd feature disable rbd1 exclusive-lock object-map deep-flatten fast-diff --name client.rbd
rbd map --image rbd1 --name client.rbd rbd
showmapped --name client.rbd

创建文件系统，并挂载
fdisk -l /dev/rbd0
mkfs.xfs /dev/rbd0
mkdir /mnt/ceph-disk1
mount /dev/rbd0 /mnt/ceph-disk1
df -h /mnt/ceph-disk1

写入数据测试
dd if=/dev/zero of=/mnt/ceph-disk1/file1 count=100 bs=1M

做成服务，开机自动挂载
vi /usr/local/bin/rbd-mount
见附件

vi /etc/systemd/system/rbd-mount.service
见附件

systemctl daemon-reload
systemctl enable rbd-mount.service

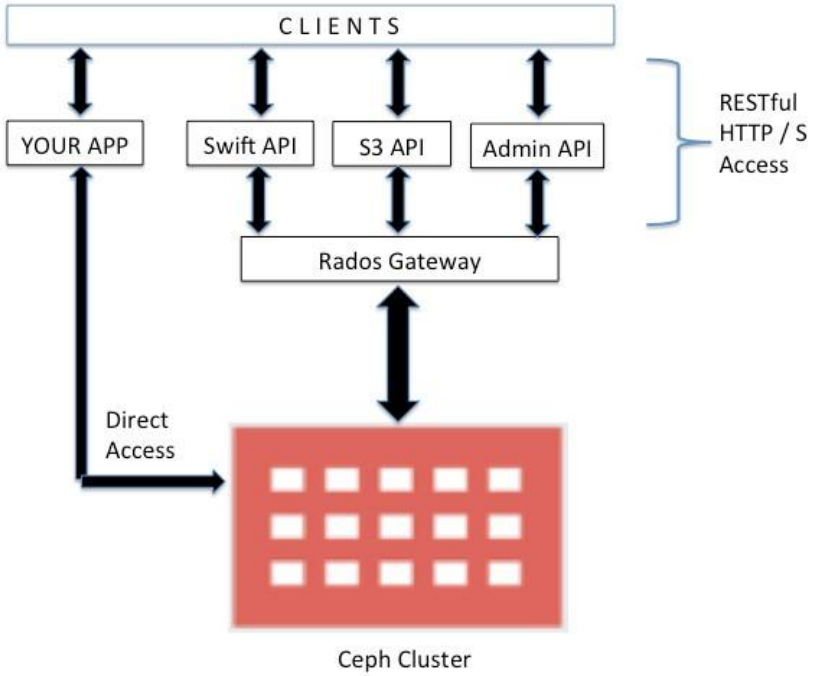
reboot -f df -h

4. Ceph 对象存储

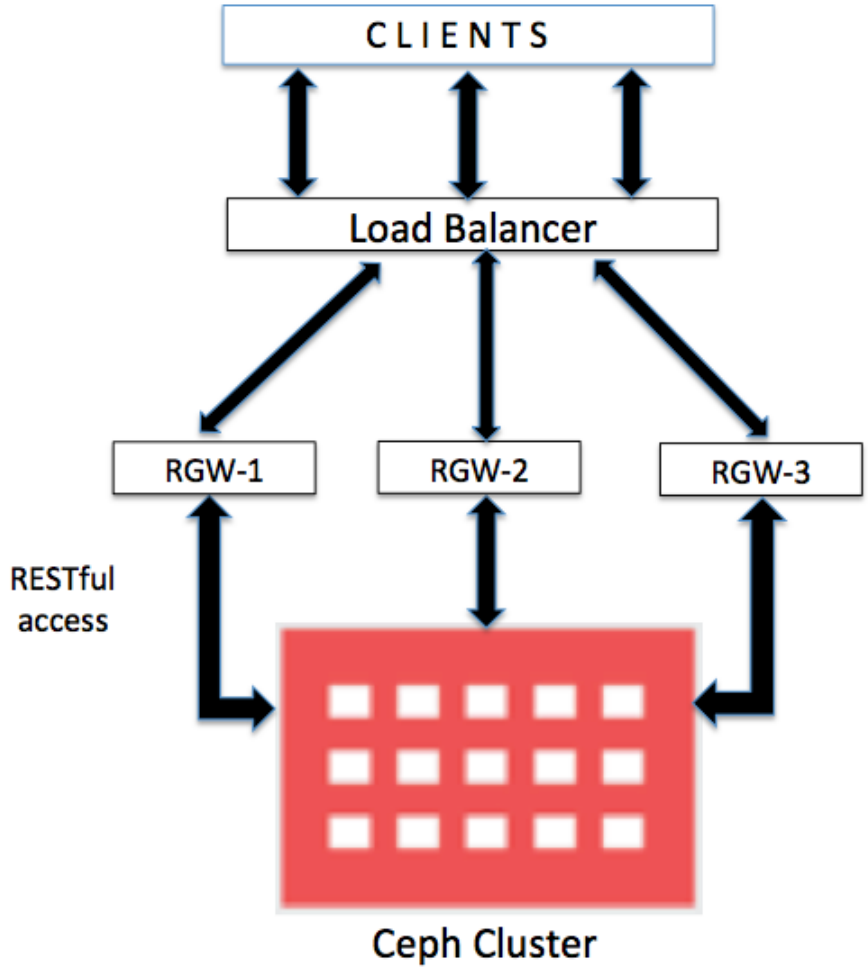
4.1 部署 Ceph 对象存储

作为文件系统的磁盘，操作系统不能直接访问对象存储。相反，它只能通过应用程序级别的 API 访问。Ceph 是一种分 布式对象存储系统，通过 Ceph 对象网关提供对象存储接口，也称为 RADOS 网关(RGW)接口，它构建在 Ceph RADOS 层之 上。RGW 使用 librgw (RADOS Gateway Library)和 librados，允许应用程序与 Ceph 对象存储建立连接。RGW 为应用程序提 供了一个 RESTful S3 / swift 兼容的 API 接口，用于在 Ceph 集群中以对象的形式存储数据。Ceph 还支持多租户对象存储，可以通过 RESTful API 访问。此外，RGW 还支持 Ceph 管理 API，可以使用本机 API 调用来管理 Ceph 存储集群。

librados 软件库非常灵活，允许用户应用程序通过 C、c++、Java、Python 和 PHP 绑定直接访问 Ceph 存储集群。Ceph 对象 存储还具有多站点功能，即为灾难恢复提供解决方案。


CLIENTS
YOUR APP
Swift API
S3 API
Admin API
RESTful
HTTP / S
Access
Rados Gateway
Direct
Access
Ceph Cluster

对于生产环境，建议您在物理专用机器上配置 RGW。但是，如果您的对象存储工作负载不太大，您可以考虑将任何 监视器机器作为 RGW 节点使用。RGW 是一个独立的服务，它从外部连接到 Ceph 集群，并向客户端提供对象存储访问。在生产环境中，建议您运行多个 RGW 实例，由负载均衡器屏蔽，如下图所示:



 # 安装 ceph-radosgw，前面已经安装
yum -y install ceph-radosgw

 # 部署
ceph-deploy rgw create diyu20 diyu21 diyu22

 # 配置 80 **端口**
vi /etc/ceph/ceph.conf
……. [client.rgw.diyu20]
rgw_frontends = "civetweb port=80"

sudo systemctl restart ceph-radosgw@rgw.diyu22.service

实战中前置会加负载均衡器，无需修改默认端口 7480

https://www.cnblogs.com/druex/articles/7018752.html
https://www.cnblogs.com/sisimi/p/7753310.html
https://blog.csdn.net/xiongwenwu/article/details/53942164

 # 创建池
pool，
create_pool.sh
见附件
chmod +x create_pool.sh

//命令仅供参考
ceph osd pool delete rbd rbd --yes-i-really-really-mean-it
调整 rbd 大小

```
ceph osd pool create rbd 30


# 测试是否能够访问 ceph 集群
sudo cp /var/lib/ceph/radosgw/ceph-rgw.diyu20/keyring ./
ceph -s -k  ./keyring  --name client.rgw.diyu20
```

4.2 使用 S3 API 访问 Ceph 对象存储

创建 radosgw 用户

radosgw-admin user create --uid=radosgw --display-name="radosgw"

会显示

```
  "keys": [
    {
      "user": "radosgw",
      "access_key": "GSDTVXLLRN1F8SPE989J",
      "secret_key": "8BX4ALNAfpdZQEe9OOAigTn718HPVard7XuktWO0"
    }
  ],
```

注意：请把 access_key 和 secret_key 保存下来，如果忘记可使用：radosgw-admin user info --uid … -k … --name …

安装 s3cmd 客户端

yum install s3cmd -y

将会在家目录下创建 .s3cfg 文件 , location 必须使用 US , 不使用 https, s3cmd --configure

复制 radosgw 的访问 key

"access_key": "GSDTVXLLRN1F8SPE989J",

"secret_key": "8BX4ALNAfpdZQEe9OOAigTn718HPVard7XuktWO0"

S3 Endpoint [s3.amazonaws.com]: 空

Encryption password: 空

Use HTTPS protocol [Yes]: no

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

将会在家目录下创建.s3cfg 文件, location 必须使用 US , 不使用 https,

编辑.s3cfg 文件，修改 host_base 和 host_bucket
vi /root/.s3cfg
可以任何一台，原则上前置会加负载均衡器
……
host_base = diyu22.diyu.com
host_bucket = %(bucket).diyu22.diyu.com:7480
……
注意这里一定要使用域名，可以在 host 中加入域名，如果使用 ip 或者机器名均会创建失败

创建桶并放入文件
s3cmd mb s3://first-bucket
s3cmd ls
s3cmd put /etc/hosts s3://first-bucket
s3cmd ls s3://first-bucket

4.3 使用 Swift API 访问 对象存储

#建立子帐号

```
radosgw-admin subuser create --uid=radosgw --subuser=radosgw:swift --access=full
```

如果子账号有问题，可先删除再加入

```
radosgw-admin subuser rm --uid=radosgw --subuser=radosgw:swift
```

查看帐号

```
radosgw-admin user info --uid=radosgw
```

记录下 swift_key

```
"swift_keys": [
{
  "user": "radosgw:swift",
  "secret_key": "qDWUGPKNvCDxzL7cjBWHO3tNoTX6lWhayyroHwdQ"
}
```

安装软件

```
yum -y install python-pip
pip install --upgrade pip
pip  install  --upgrade  python-swiftclient
```

客户端访问

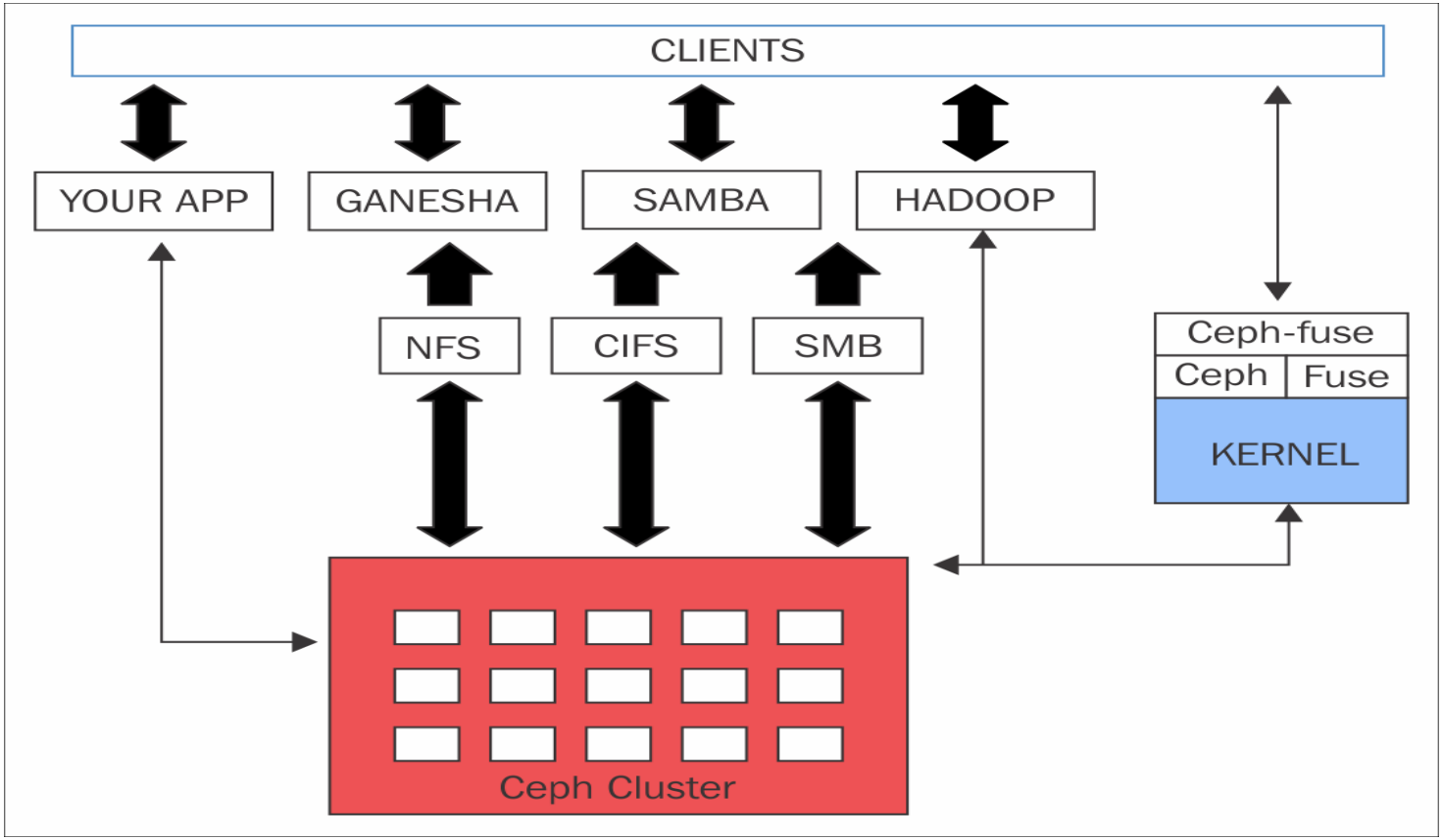
可不用域名访问，机器名和 ip 均可以

```
swift -A  http://diyu22:7480/auth/1.0 -U radosgw:swift -K   rXMTDPwcglA3ceNOo53Y11O5dnyXPKLSCTH5kK6z  list
swift -A  http://diyu22:7480/auth/1.0 -U radosgw:swift -K   rXMTDPwcglA3ceNOo53Y11O5dnyXPKLSCTH5kK6z  post second-bucket
swift -A  http://diyu22:7480/auth/1.0 -U radosgw:swift -K   rXMTDPwcglA3ceNOo53Y11O5dnyXPKLSCTH5kK6z  list
```

5. Ceph 文件存储

5.1 部署 Ceph 文件存储

Ceph 文件系统提供了任何大小的符合 posix 标准的分布式文件系统，它使用 Ceph RADOS 存储数据。要实现 Ceph 文件系统，您需要一个正在运行的 Ceph 存储集群和至少一个 Ceph 元数据服务器(MDS)来管理其元数据并使其与数据分离，这有助于降低复杂性和提高可靠性。下图描述了 Ceph FS 的架构视图及其接口:



libcephfs 库在支持其多个客户机实现方面发挥着重要作用。它具有本机 Linux 内核驱动程序支持，因此客户机可以使 用本机文件系统安装，例如，使用 mount 命令。它与 SAMBA 紧密集成，支持 CIFS 和 SMB。Ceph FS 使用 cephfuse 模块扩 展到用户空间(FUSE)中的文件系统。它还允许使用 libcephfs 库与 RADOS 集群进行直接的应用程序交互。作为 Hadoop HDFS 的替代品，Ceph FS 越来越受欢迎。

只有 Ceph FS 才需要 Ceph MDS;其他存储方法的块和基于对象的存储不需要 MDS 服务。Ceph MDS 作为一个守护进程运行，它允许客户机挂载任意大小的 POSIX 文件系统。MDS 不直接向客户端提供任何数据;数据服务仅由 OSD 完成。

部署 cephfs

只装一个服务
ceph-deploy mds create diyu21

ceph health detail
ceph -s

######################
https://blog.csdn.net/zahurqf/article/details/83145424
要先停止服务才能删除，注意是启动 mds 服务的那台机器
systemctl stop ceph-mds.target
ceph fs rm cephfs --yes-i-really-mean-it
ceph osd pool delete cephfs_data cephfs_data --yes-i-really-really-mean-it
ceph osd pool delete cephfs_metadata cephfs_metadata --yes-i-really-really-mean-it
systemctl start ceph-mds.target

Error EINVAL: key for client.cephfs exists but cap osd does not match
可以删除

ceph auth del client.cephfs
######################
注意：查看输出，应该能看到执行了哪些命令，以及生成的 keyring

ceph osd pool create cephfs_data 128
ceph osd pool create cephfs_metadata 64
ceph fs new cephfs cephfs_metadata cephfs_data

ceph mds stat
ceph osd pool ls
ceph fs ls

 # 创建用户(可选，因为部署时，已经生成)
#这里注意写法，如容易写错，会造成后面无法挂载
ceph auth get-or-create client.cephfs mon 'allow r' mds 'allow r, allow rw path=/' osd 'allow rw pool=cephfs_data' -o ceph.client.cephfs.keyring

copy 到客户端
scp ceph.client.cephfs.keyring root@192.168.1.12:/etc/ceph/

5.2 通过内核驱动和 FUSE 客户端挂载 Ceph FS

在 Linux 内核 2.6.34 和以后的版本中添加了对 Ceph 的本机支持。

 # 创建挂载目录
mkdir /mnt/cephfs

挂载
ceph auth get-key client.cephfs
AQBRJ2ZcObg7BhAAE8wM5jAJRkhTPj9Nb6Mgug==

#在启动 mds 那台机器上查看端口 diyu21
netstat -tunlp | grep 6789
证明服务已经启动

客户端挂载
mount -t ceph 192.168.1.21:6789:/ /mnt/cephfs -o name=cephfs,secret=AQBRJ2ZcObg7BhAAE8wM5jAJRkhTPj9Nb6Mgug==
df -h /mnt/cephfs/
umount /mnt/cephfs/

删除后用文件挂载
mount -t ceph diyu21:6789:/ /mnt/cephfs -o name=cephfs,secretfile=/etc/ceph/cephfs
df -h /mnt/cephfs/

开机自启动
vi /etc/fstab
添加

diyu21:6789:/ /mnt/cephfs ceph name=cephfs,secretfile=/etc/ceph/cephfs,_netdev,noatime 0 0

校验

umount /mnt/cephfs

mount /mnt/cephfs

dd if=/dev/zero of=/mnt/cephfs/file1 bs=1M count=1024

Ceph 文件系统由 LINUX 内核本地支持;但是，如果您的主机在较低的内核版本上运行，或者您有任何应用程序依赖项，您总是可以使用 FUSE 客户端让 Ceph 挂载 Ceph FS。

安装软件包
rpm -qa|grep -i ceph-fuse
yum -y install ceph-fuse

挂载
ceph-fuse --keyring /etc/ceph/ceph.client.cephfs.keyring --name client.cephfs -m diyu21:6789 /mnt/cephfs
umount /mnt/cephfs/

开机自启动
vi /etc/fstab
id=cephfs,keyring=/etc/ceph/ceph.client.cephfs.keyring /mnt/cephfs fuse.ceph defaults 0 0 _netdev

注：因为 keyring 文件包含了用户名，所以 fstab 不需要指定用了

5.3 将 Ceph FS 导出为 NFS 服务器

网络文件系统(Network Filesystem, NFS)是最流行的可共享文件系统协议之一，每个基于 unix 的系统都可以使用它。不理解 Ceph FS 类型的基于 unix 的客户机仍然可以使用 NFS 访问 Ceph 文件系统。要做到这一点，我们需要一个 NFS 服 务器，它可以作为 NFS 共享重新导出 Ceph FS。NFS-ganesha 是一个在用户空间中运行的 NFS 服务器，使用 libcephfs 支 持 Ceph FS 文件系统抽象层(FSAL)。

安装软件
yum install -y nfs-utils nfs-ganesha

启动 NFS 所需的 rpc 服务
systemctl enable rpcbind
systemctl start rpcbind
systemctl status rpcbind

修改配置文件
vi /etc/ganesha/ganesha.conf

```
EXPORT
{
     Export_ID = 1;
     Path = "/";
     Pseudo = "/";
     Access_Type = RW;
     SecType = "none";
     NFS_Protocols = "3";
     Squash = No_Root_Squash;
     Transport_Protocols = TCP;

     FSAL {
             Name = CEPH;
     }
}
```

ps -aux|grep ganesha

#通过提供 Ganesha.conf 启动 NFS Ganesha 守护进程

ganesha.nfsd -f /etc/ganesha/ganesha.conf -L /var/log/ganesha.log -N NIV_DEBUG

showmount -e

客户端挂载
yum install -y nfs-utils
mkdir/mnt/cephnfs
mount -o rw,noatime diyu21:/ /mnt/cephnfs

https://www.zstack.io/help/?name=a_print_ZStack_Ceph_Deploy_Tutorial&page=topic/ZStack_2.1_Ceph_Deploy_Tutorial_0014.html&level=2.2.6
yum install net-tools