

实验环境手工配置ceph集群中的osd主机添加磁盘

参考文档

[Ceph手动部署指南](#)

操作过程

1. 对磁盘进行gpt格式的分区

```
#parted -a optimal -s /dev/DISK mktable gpt          ## 对指定磁盘进行分区
```

2. 创建日志分区

```
#parted -a optimal -s /dev/DISK mkpart ceph 0% 15GB    ## 对指定磁盘进行ceph类型的分区，并指定分区空间为15GB，  
用于作为journal区域
```

3. 创建数据分区

```
#parted -a optimal -s /dev/DISK mkpart ceph 15GB 100%  ## 对指定磁盘进行ceph类型的分区，并指定分区空间为15GB，  
用于作为journal区域
```

4. 对磁盘的data区域进行文件系统格式化

```
#mkfs.xfs /dev/DISK2
```

5. 创建并生成一个osd id

```
# ceph osd create==> OSD_ID
```

6. 创建对应的osd目录

```
#mkdir /var/lib/ceph/osd/ceph-OSD_ID          ## 创建osd目录作为对应data区域的挂载点
```

7. 挂载文件系统：将data所在设备文件挂载至指定osd目录

```
# mount -t xfs -o rw,nodev,noexec,noatime,nodiratime,attr2,discard,inode64,logbsize=256k,noquota/dev/DISK2  
/var/lib/ceph/osd/ceph-OSD_ID
```

8. 查看磁盘本身的wwn序列号

```
#ls -la /dev/disk/by-id/ | grep DISK2 | grep wwn | awk '{print $9}'|awk -F- '{print $2}' ==> WWN
```

9. 编辑主机挂载分区表

```
#echo "/dev/disk/by-id/wwn-WWN-part2/var/lib/ceph/osd/ceph-OSD_IDxfsrw,noexec,nodev,noatime,nodiratime,barrier=0,discard,inode64  
,logbsize=256k,delaylog02" >> /etc/fstab          ## 编辑主机的挂载分区表，其中使用wwn-WWN-part2来标识数据分区的设备文件
```

10. 查看并记录数据分区的uuid

```
#lsblk -f | grep DISK2|awk '{printf $3}' ==> UUID
```

11. 向ceph的配置文件中添加如下配置

```
#vim /etc/ceph/ceph.conf
```

```
>>
```

```
[osd.OSD_ID]
ustack_osd_journal_name = wwn-WWN-part1
host = HOSTNAME
ustack_osd_device = /dev/DISK2
osd_data = /var/lib/ceph/osd/ceph-OSD-ID
ustack_osd_uuid = UUID
osd_journal_size = 14336
ustack_osd_mount_device = /dev/disk/by-id/wwn-WWN-part2
osd_journal = /dev/disk/by-id/wwn-WWN-part1
ustack_osd_name = wwn-WWN-part2
```

```
<<
```

12. 初始化指定osd实例

```
# ceph-osd -i OSD_ID --mkfs --mkkey --osd-uuid UUID      ## 使用格式化文件系统产生的uuid来作为指定osd的标识
```

13. 添加指定osd实例的认证信息

```
#ceph auth add osd.OSD_ID osd 'allow *' mon 'allow rwx' -i /var/lib/ceph/osd/ceph-OSD_ID/keyring      ##
对osd.OSD_ID这个osd实例的key赋予具有的权限：允许其他osd进行所有操作，允许mon进行rwx操作，使用-i指定key文件的路径
```

14. 启动osd实例进程

```
# service ceph start osd.OSD_ID
```

15. 查看osd进程是否正常启动

```
# ps -ef|grep osd
```