**分区规划及使用 LVM逻辑卷 管理交换空间**

磁道：track

扇区：sector

磁头：head

柱面：cylinder

每个扇区，512字节

• 识别硬盘 => 分区规划 => 格式化 => 挂载使用

毛坯楼层---------》打隔断---------》装修---------》入驻

• MBR/msdos 分区模式

– 1~4个主分区,或者 0~3个主分区+1个扩展分区(n

个逻辑分区)

– 最大支持容量为 2.2TB 的磁盘

– 扩展分区不能格式化

**1. 查看硬盘**

**[root@server0 /]# lsblk**

**2.修改硬盘的分区表**

**[root@server0 /]# fdisk /dev/vdb**

**n 创建新的分区----->回车----->回车---->回车----->在last结束时 +2G**

**p 查看分区表**

**n 创建新的分区----->回车----->回车---->回车----->在last结束时 +2G**

**d 删除分区**

**w 保存并退出**

**[root@server0 /]# lsblk**

**[root@server0 /]# ls /dev/vdb[1-2]**

**3. 格式化文件系统**

**• mkfs 工具集**

– mkfs.ext3 分区设备路径

– mkfs.ext4 分区设备路径

– mkfs.xfs 分区设备路径

– mkfs.vfat -F 32 分区设备路径

**[root@server0 /]# mkfs.ext4 /dev/vdb1**

**[root@server0 /]# blkid /dev/vdb1 #查看分区文件系统命令**

**[root@server0 /]# mkfs.xfs /dev/vdb2**

**[root@server0 /]# blkid /dev/vdb2 #查看分区文件系统的命令**

**4. 挂载使用**

**[root@server0 ~]# mount /dev/vdb1 /part1**

**[root@server0 ~]# mount /dev/vdb2 /part2**

**[root@server0 ~]# df -h #查看正在挂载的分区使用情况**

**5.开机自动挂载**

**• 配置文件 /etc/fstab 的记录格式**

**– 设备路径 挂载点 类型 参数 备份标记 检测顺序**

**[root@server0 ~]# vim /etc/fstab**

**/dev/vdc5 /mnt/mypart ext4 defaults 0 0**

**[root@server0 ~]# tail -2 /etc/fstab**

**/dev/vdb1 /part1 ext4 defaults 0 0**

**/dev/vdb2 /part2 xfs defaults 0 0**

**[root@server0 ~]# df -h**

**验证：**

**[root@server0 ~]# mount -a**

**检测/etc/fstab开机自动挂载配置文件,格式是否正确**

**检测/etc/fstab中,书写完成,但当前没有挂载的设备,进行挂载**

**[root@server0 ~]# df -h**

**[root@server0 ~]# fdisk /dev/vdb #划分第三个分区**

**[root@server0 ~]# partprobe 刷新 新的分区**

**总结：**

1.lsblk 查看新的磁盘

2.fdisk 划分新的分区

3.partprobe 刷新

4.mkfs 格式化文件系统

5.mount 挂载使用

6./etc/fstab 开机自动

**综合分区：**

1.添加一个80G的虚拟硬盘

[root@server0 ~]# lsblk

2.划分区如下：

划分3个主分区，每个10G

划分3个逻辑分区，每个5G

**[root@server0 ~]# fdisk /dev/vdc**

**p 查看分区表**

**n 创建新的分区----->回车----->回车---->回车----->在last结束时 +10G**

**n 创建新的分区----->回车----->回车---->回车----->在last结束时 +10G**

**n 创建新的分区----->回车----->回车---->回车----->在last结束时 +10G**

**p 查看分区表**

**n 创建新的分区**

**----->回车---->起始回车----->结束回车 将所有空间给扩展分区**

**p 查看分区表**

**n 创建新的分区----->起始回车------>结束+5G**

**n 创建新的分区----->起始回车------>结束+5G**

**n 创建新的分区----->起始回车------>结束+5G**

**d 删除分区**

**w 保存并退出**

**[root@server0 ~]# ls /dev/vdc[1-7]**

**LVM逻辑卷管理**

作用： 1.整合分散的空间

2.LVM逻辑卷空间可以动态的扩大或者缩减

零散空闲存储 ---- 整合的虚拟磁盘 ---- 虚拟的分区

将众多的物理卷，组成卷组，在从卷组中还分出逻辑卷

面粉 ------------>大面团----------->小面团-------->蒸------->吃

砖------------>房子------------>打隔断------> 装修------>入住

**建立LVM逻辑卷**

**1.创建卷组**

**命令格式：vgcreate 卷组名 分区路径**

**[root@server0 ~]# vgcreate myvg /dev/vdc[1-2]**

**[root@server0 ~]# vgs #查看卷组基本信息**

**[root@server0 ~]# pvs #查看物理卷基本信息**

**2.创建逻辑卷**

**命令格式：lvcreate -L 大小 -n 逻辑卷名称 基于卷组名**

**[root@server0 ~]# lvcreate -L 16G -n mylv myvg**

**Logical volume "mylv" created**

**[root@server0 ~]# lvs**

**[root@server0 ~]# mkfs.ext4 /dev/myvg/mylv**

**[root@server0 ~]# vim /etc/fstab**

**/dev/myvg/mylv /lvm ext4 defaults 0 0**

**[root@server0 ~]# mkdir /lvm**

**[root@server0 ~]# mount -a**

**[root@server0 ~]# df -h**

**LVM逻辑卷的扩展**

**一、卷组有足够的剩余空间**

**1.直接扩展逻辑卷的空间**

[root@server0 ~]# vgs

[root@server0 ~]# lvs

[root@server0 ~]# lvextend -L 18G /dev/myvg/mylv

[root@server0 ~]# lvs

**2.扩展逻辑卷文件系统**

扩展ext4: resize2fs

扩展xfs: xfs\_growfs

[root@server0 ~]# blkid /dev/myvg/mylv

[root@server0 ~]# df -h

[root@server0 ~]# resize2fs /dev/myvg/mylv

[root@server0 ~]# df -h

**二、卷组没有足够的剩余空间**

**1.扩展卷组**

[root@server0 ~]# vgs

[root@server0 ~]# vgextend myvg /dev/vdc3

[root@server0 ~]# vgs

**2.直接扩展逻辑卷的空间**

[root@server0 ~]# lvs

[root@server0 ~]# lvextend -L 25G /dev/myvg/mylv

[root@server0 ~]# lvs

**3.扩展逻辑卷文件系统**

扩展ext4: resize2fs

扩展xfs: xfs\_growfs

[root@server0 ~]# blkid /dev/myvg/mylv

[root@server0 ~]# df -h

[root@server0 ~]# resize2fs /dev/myvg/mylv

[root@server0 ~]# df -h

**LVM其他内容**

1. LVM的缩减（了解）

2. PE单位：卷组划分空间单位 4M

[root@server0 ~]# vgdisplay myvg

[root@server0 ~]# vgchange -s 1M myvg #修改PE的大小

[root@server0 ~]# vgdisplay myvg

[root@server0 ~]# lvcreate -L 250M -n lvtest01 myvg

[root@server0 ~]# lvs

命令格式：lvcreate -l PE的个数 -n 逻辑卷的名字 基于的卷组

[root@server0 ~]# lvcreate -l 86 -n lvtest03 myvg

Logical volume "lvtest03" created

[root@server0 ~]# lvs

**交换分区的管理**

**一、分区**

**[root@server0 ~]# lsblk**

parted 专用于 划分大空间 用于划分GPT分区方案

GPT分区模式 支持 128个主分区

支持 18EB空间

**[root@server0 ~]# parted /dev/vdb**

**(parted) mktable gpt #指定分区的模式**

**(parted) print #输出分区表**

**(parted) mkpart #划分新的分区**

分区名称？ []? haha #分区的名称随意指定

文件系统类型？ [ext2]? #文件系统直接回车

起始点？ 0

结束点？ 2G

警告: The resulting partition is not properly aligned

for best performance.

忽略/Ignore/放弃/Cancel? Ignore #选择忽略

(parted) unit GB #用GB作为显示单位

(parted) quit #退出

**二、交换分区的使用**

• 相当于虚拟内存,

– 当物理内存不够用时,使用磁盘空间来模拟内存

– 在一定程度上缓解内存不足的问题

– 交换分区:以空闲分区充当的交换空间

**1.查看交换空间信息**

[root@server0 ~]# swapon -s

**2.制作交换空间**

**[root@server0 ~]# mkswap /dev/vdb1 #格式化交换文件系统**

**[root@server0 ~]# mkswap /dev/vdb2 #格式化交换文件系统**

**[root@server0 ~]# swapon /dev/vdb1 #启用交换分区**

**[root@server0 ~]# swapon -s**

**[root@server0 ~]# swapon /dev/vdb2**

**[root@server0 ~]# swapon -s**

**[root@server0 ~]# swapoff /dev/vdb1 #停用交换分区**

**[root@server0 ~]# swapon -s**

**[root@server0 ~]# swapoff /dev/vdb2 #停用交换分区**

**[root@server0 ~]# swapon -s**

**3.完成开机自动启用交换空间**

[root@server0 ~]# blkid /dev/vdb1

/dev/vdb1: UUID="07de15a7-baa9-48ee-856a-dec6f2b2c1d4" TYPE="swap" PARTLABEL="haha" PARTUUID="ca01c66b-1a2f-443f-bc45-ceeb6f65a312"

[root@server0 ~]# vim /etc/fstab

UUID="07de15a7-baa9-48ee-856a-dec6f2b2c1d4" swap swap defaults 0 0

[root@server0 ~]# swapon -a

[root@server0 ~]# swapon -s

**附加权限**

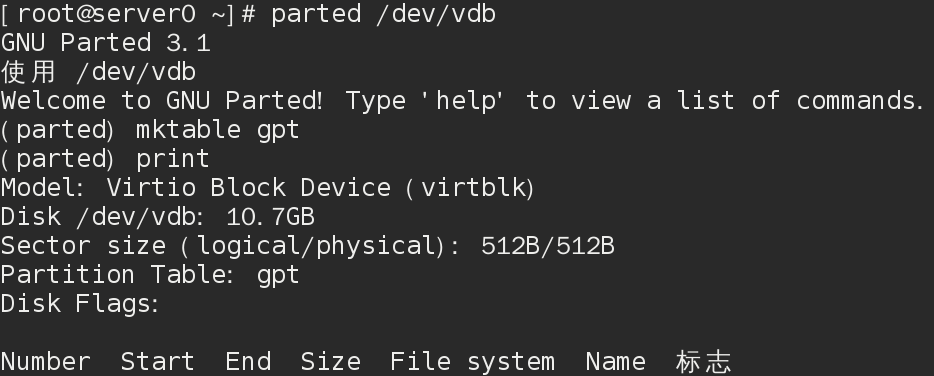
Set UID

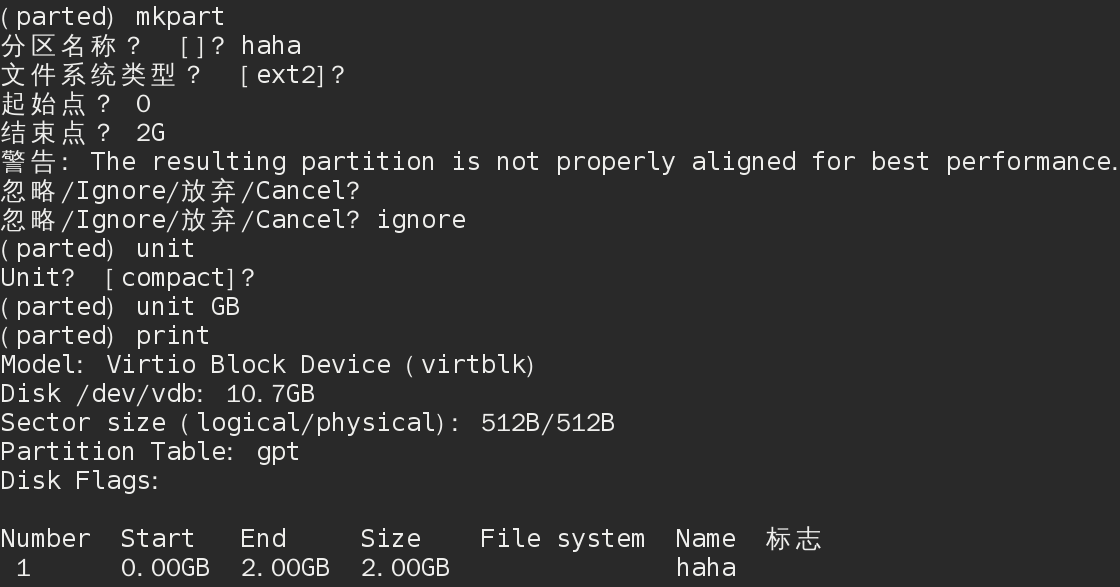
• 附加在属主的 x 位上

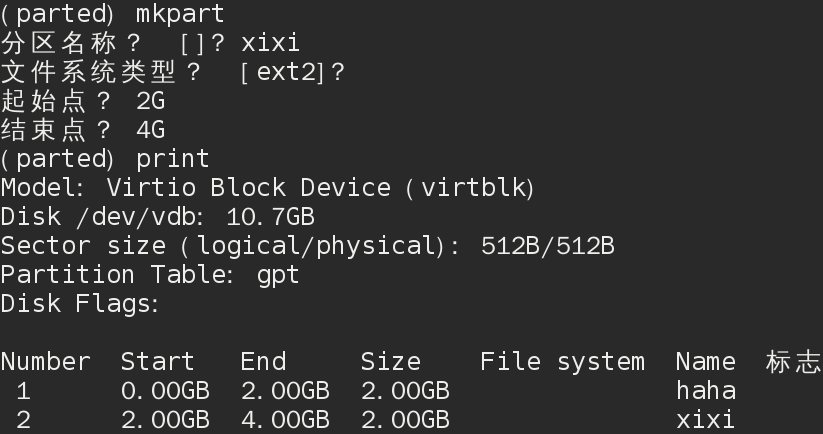
– 属主的权限标识会变为 s

– 适用于可执行文件,Set UID可以让使用者具有文件属主的身份及部分权限

– 传递所有者身份





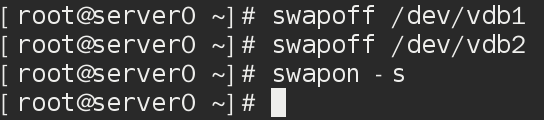




格式化交换文件系统、启用交换分区



停用交换分区



完成开机自动启用交换空间

