一、环境准备

还原快照，开启CentOS7虚拟机

昨日的习题：

案例1：chmod权限设置

1）以root用户新建/nsddir/目录，在该目录下新建文件readme.txt

2）使用户zhangsan能够在/nsddir/目录下创建/删除子目录

3）使用户zhangsan能够修改/nsddir/readme.txt文件的容

[root@A ~]# mdkir /nsddir

[root@A ~]# echo haha >> /nsddir/readme.txt

[root@A ~]# chmod o+w /nsddir

[root@A ~]# su - zhangsan

[zhangsan@A ~]$ mkdir /nsddir/zhangsan

[zhangsan@A ~]$ ls /nsddir

[zhangsan@A ~]$ exit

[root@A ~]# chmod o+w /nsddir/readme.txt

[root@A ~]# su - zhangsan

[zhangsan@A ~]$ echo xixi >> /nsddir/readme.txt

[zhangsan@A ~]$ cat /nsddir/readme.txt

[zhangsan@A ~]$ exit

案例2：chown归属设置

1）新建/tarena1目录

a）将属主设为gelin01，属组设为tarena组

b）使用户gelin01对此目录具有rwx权限，其他人对此目录无任何权限

[root@A ~]# mkdir /tarena1

[root@A ~]# useradd gelin01

[root@A ~]# groupadd tarena

[root@A ~]# chown gelin01:tarena /tarena1

[root@A ~]# ls -ld /tarena1

[root@A ~]# chmod o=--- /tarena1

[root@A ~]# ls -ld /tarena1

2）使用户gelin02能进入、查看/tarena1文件夹（提示：将gelin02加入所属组）

[root@A ~]# useradd gelin02

[root@A ~]# gpasswd -a gelin02 tarena

[root@A ~]# id gelin02

[root@A ~]# su - gelin02

[gelin02@A ~]$ cd /tarena1

[gelin02@A tarena1]$ cd /tarena1

[gelin02@A tarena1]$ ls

[gelin02@A tarena1]$ exit

3）新建/tarena2目录

a）将属组设为tarena

b）使tarena组的任何用户都能在此目录下创建、删除文件

[root@A ~]# mkdir /tarena2

[root@A ~]# chown :tarena /tarena2

[root@A ~]# chmod g+w /tarena2

[root@A ~]# ls -ld /tarena2

[root@A ~]# useradd ceshi

[root@A ~]# gpasswd -a ceshi tarena

[root@A ~]# id ceshi

[root@A ~]# su - ceshi

[ceshi@A ~]$ mkdir /tarena2/ceshi

[ceshi@A ~]$ ls /tarena2

[ceshi@A ~]$ exit

4）新建/tarena/public目录

a）使任何用户对此目录都有rwx权限

b）拒绝zhangsan进入此目录，对此目录无任何权限

[root@A ~]# mkdir /tarena/public

[root@A ~]# chmod 777 /tarena/public

[root@A ~]# ls -ld /tarena/public

[root@A ~]# setfacl -m u:zhangsan:--- /tarena/public

[root@A ~]# su - zhangsan

[zhangsan@A ~]$ ls /tarena/public

[zhangsan@A ~]$ cd /tarena/public

[zhangsan@A ~]$ exit

案例3:权限设置

1、创建文件夹/data/test,设置目录的访问权限，使所有者和所属组具备读写执行的权限；其他人无任何权限。

2、递归修改文件夹/data/test的归属使所有者为zhangsan，所属组为tarena。

3、请实现在test目录下，新建的所有子文件或目录的所属组都会是tarena。

4、为lisi创建ACL访问权限，使得lisi可以查看/etc/shadow文件

[root@A ~]# mkdir /data/test

[root@A ~]# chmod u=rwx,g=rwx,o=--- /data/test 或者 chmod 770 /data/test

[root@A ~]# ls -ld /data/test

[root@A ~]# chown -R zhangsan:tarena /data/test

[root@A ~]# ls -ld /data/test

[root@A ~]# chmod g+s /data/test

[root@A ~]# mkdir /data/test/abc

[root@A ~]# ls -ld /data/test/abc

[root@A ~]# setfacl -m u:lisi:r /etc/shadow

[root@A ~]# getfacl /etc/shadow

[root@A ~]# su - lisi

[lisi@A ~]$ cat /etc/shadow

[lisi@A ~]$ exit

案例4:虚拟机 server0上操作

将文件 /etc/fstab 拷贝为 /var/tmp/fstab，并调整文件 /var/tmp/fstab权限

满足以下要求：

– 此文件的拥有者是 root

– 此文件对任何人都不可执行

– 用户 natasha 能够对此文件执行读和写操作

– 用户 harry 对此文件既不能读，也不能写

[root@A ~]# cp /etc/fstab /var/tmp/fstab

[root@A ~]# ls -l /var/tmp/fstab

[root@A ~]# setfacl -m u:natasha:rw /var/tmp/fstab

[root@A ~]# getfacl /var/tmp/fstab

[root@A ~]# su - natasha

[natasha@A ~]$ cat /var/tmp/fstab

[natasha@A ~]$ echo ceshi >> /var/tmp/fstab

[natasha@A ~]$ cat /var/tmp/fstab

[natasha@A ~]$ exit

[root@A ~]# setfacl -m u:harry:--- /var/tmp/fstab

[root@A ~]# getfacl /var/tmp/fstab

[root@A ~]# su - harry

[harry@A ~]$ cat /var/tmp/fstab

[harry@A ~]$ echo ceshi >> /var/tmp/fstab

[harry@A ~]$ exit

案例5:虚拟机操作

创建一个共用目录 /home/admins，要求如下：

– 此目录的所属组是 adminuser

– adminuser 组的成员对此目录有读写和执行的权限，并且其他用户没有任何权限

– 在此目录中创建的文件，其所属组会自动设置为 属于 adminuser 组

[root@A ~]# mkdir /home/admins

[root@A ~]# groupadd adminuser

[root@A ~]# chown :adminuser /home/admins

[root@A ~]# ls -ld /home/admins

[root@A ~]# chown g+s /home/admins

[root@A ~]# ls -ld /home/admins

[root@A ~]# mkdir /home/admins/ceshi

[root@A ~]# ls -ld /home/admins/ceshi

设置权限使得lisi用户,可以查看/etc/shadow文件,您有几种办法？

1.利用基本权限，其他人直接设置

chmod o+r /etc/shadow

2.利用基本权限，所属组直接设置

chown :lisi /etc/shadow

chmod g+r /etc/shadow

3.利用基本权限，所有者直接设置

chown lisi /etc/shadow

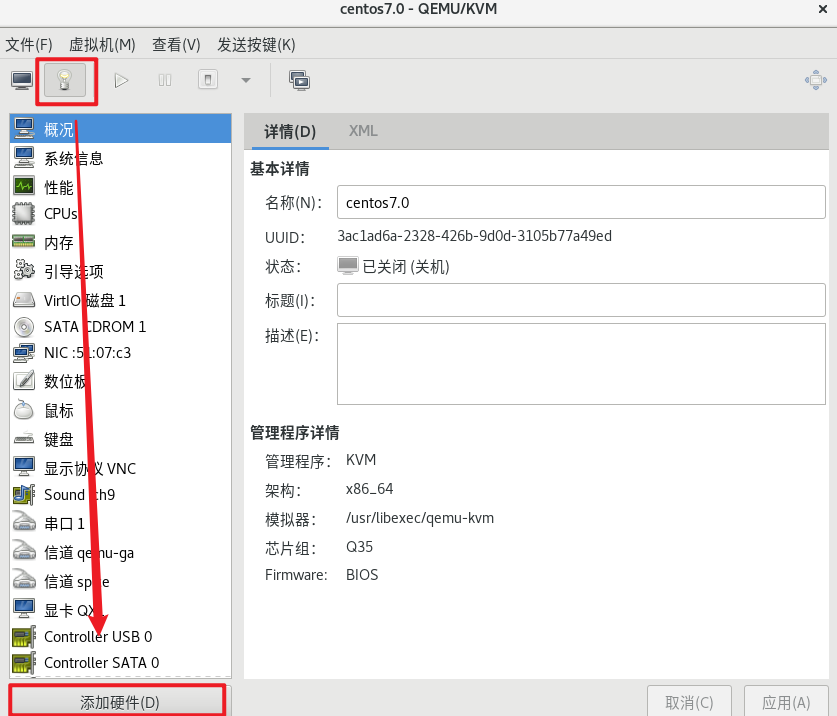
chmod u+r /etc/shadow

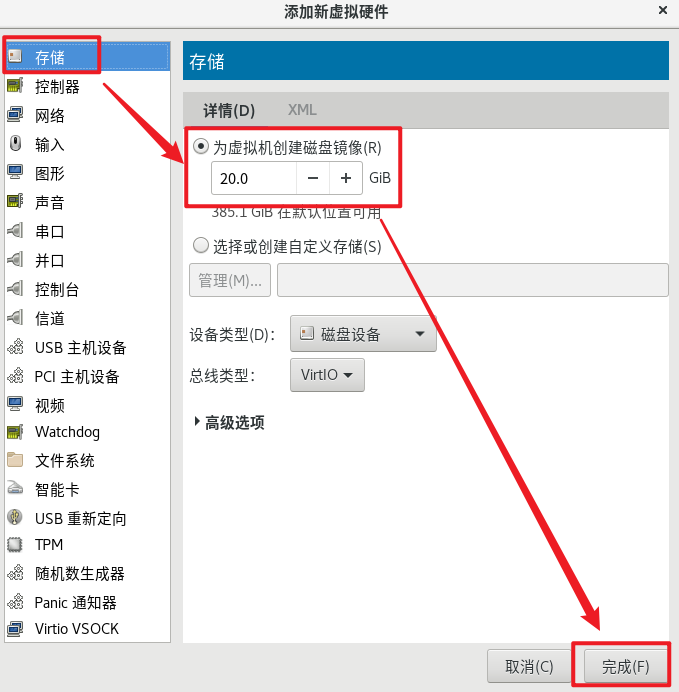
4.利用ACL策略

setfacl -m u:lisi:r /etc/shadow

二、磁盘空间的管理

真机是Linux：KVM虚拟机



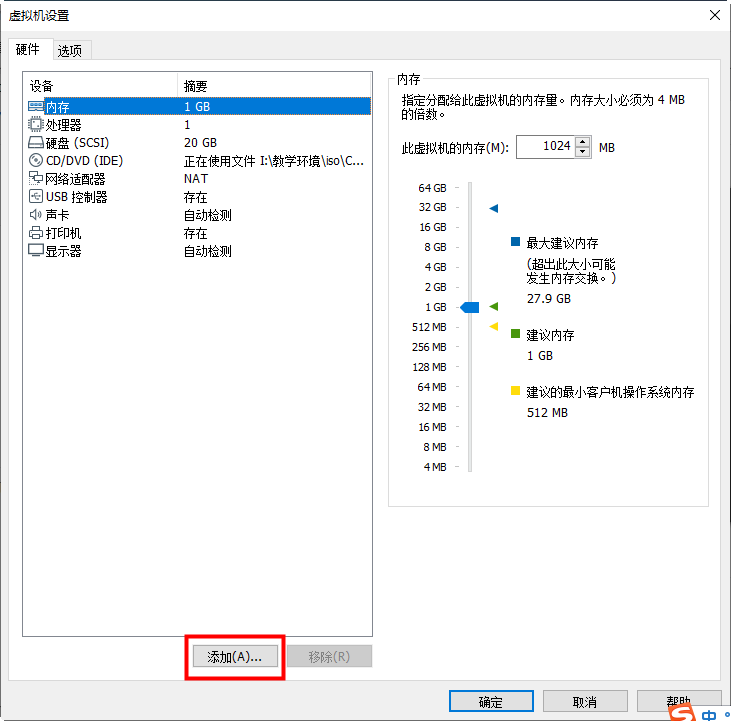


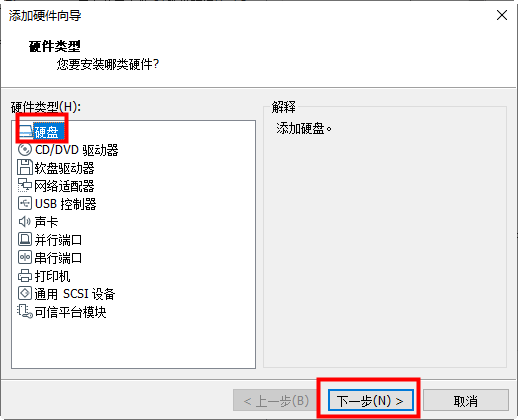
真机是windows：添加一块新的硬盘（磁盘）

关闭虚拟机CentOS7

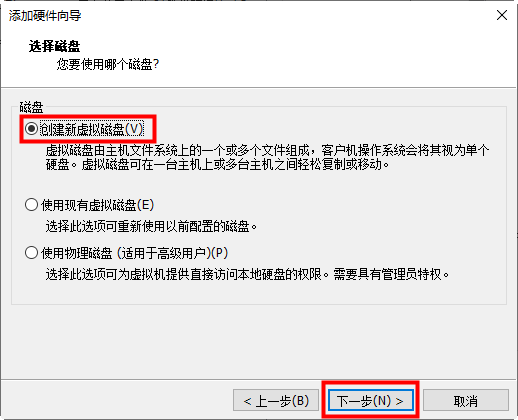
[root@localhost ~]# poweroff

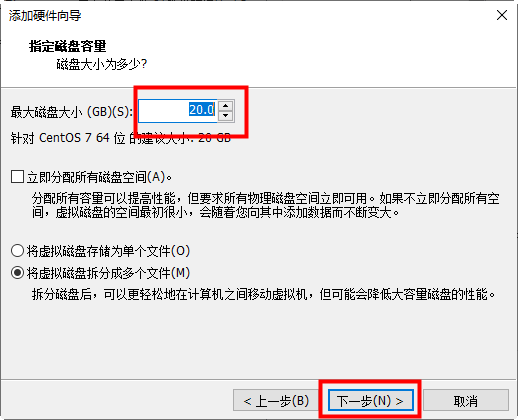


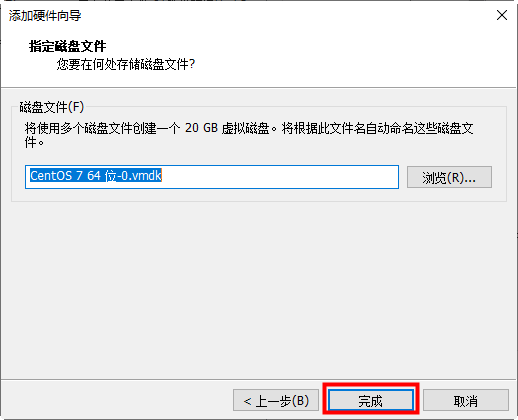


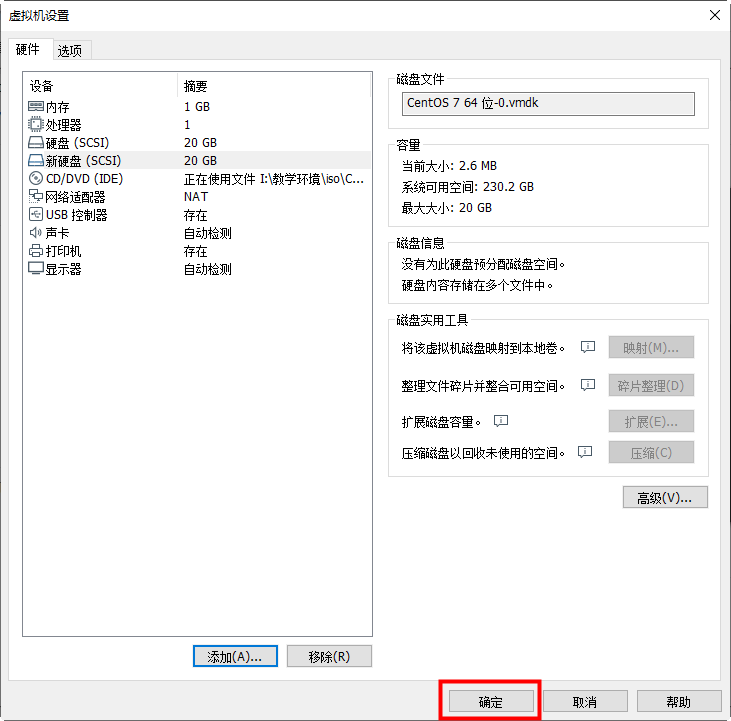














2.查看本机识别的新的硬盘

**真机为Windows机器，VMware虚拟机**

[root@localhost ~]# lsblk #列出当前系统识别的硬盘

[root@localhost ~]# ls /dev/sda #SCSI接口第一块

[root@localhost ~]# ls /dev/sdb #SCSI接口第二块

**真机为Linux机器，KVM虚拟机**

[root@localhost ~]# lsblk #列出当前系统识别的硬盘

[root@localhost ~]# ls /dev/vda #虚拟接口第一块

[root@localhost ~]# ls /dev/vdb #虚拟接口第二块

**扇区默认的大小：512字节**

**计算机容量单位：**

一般用B，KB，MB，GB，TB，PB，EB，ZB，YB，BB来表示，它们之间的关系是：

**1KB (Kilobyte 千字节)=1024B,**

**1MB (Megabyte 兆字节 简称“兆”)=1024KB，**

**1GB (Gigabyte 吉字节 又称“千兆”)=1024MB,**

**1TB (Terabyte 万亿字节 太字节)=1024GB,**

**1PB (Petabyte 千万亿字节 拍字节)=1024TB,**

**1EB (Exabyte 百亿亿字节 艾字节)=1024PB,**

1ZB (Zettabyte 十万亿亿字节 泽字节)= 1024 EB,

1YB (Yottabyte 一亿亿亿字节 尧字节)= 1024 ZB,

1BB (Brontobyte 一千亿亿亿字节)= 1024 YB.

**一块硬盘的“艺术”之旅（硬盘空间使用，经历的步骤）**

•识别硬盘 => 分区规划 => 格式化 => 挂载使用



**三、识别硬盘**

[root@localhost ~]# **lsblk** #列出当前系统识别的硬盘

**四、分区规划**

•分区方案（分区模式）： **MBR与GPT**

•**MBR/msdos**分区模式

–分区类型：**主分区、扩展分区(占用所有剩余空间)、逻辑分区**

–最多只能有**4个主分区**

–**扩展分区可以没有，至多有一个**

–1~4个主分区，或者 **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，空间不能直接存储数据**

–可以用于存储数据的分区：**主分区与逻辑分区**

fdisk常用交互指令：

m 列出指令帮助

p 查看现有的分区表（存放分区信息的表格）

n 新建分区

d 删除分区

q 放弃更改并退出

w 保存更改并退出

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

[root@localhost ~]# **fdisk /dev/sdb**

**n 创建新的分区----->分区类型 回车----->分区编号 回车---->起始扇区 回车----->在last结束时 +2G**

**p 查看分区表**

**n 创建新的分区----->分区类型 回车----->分区编号 回车---->起始扇区 回车----->在last结束时 +1G**

**w 保存并退出**

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

[root@localhost ~]# **ls /dev/sdb[1-2]**

**五、格式化：赋予空间文件系统的过程**

文件系统：数据在空间中存放的规则

Windows常见的文件系统：NTFS FAT(兼容性强)

Linux常见的文件系统：ext4(RHEL6) xfs(RHEL7) FAT

]# **ls /dev/sdb[1-2]**

/dev/sdb1 /dev/sdb2

]# **mkfs.(tab)(tab)**  #连续按两次tab键

]# **mkfs.ext4 /dev/sdb1**  #格式化文件系统ext4

]# **mkfs.xfs /dev/sdb2**  #格式化文件系统xfs

]# **blkid /dev/sdb1**  #查看文件系统类型

]# **blkid /dev/sdb2**  #查看文件系统类型

**六、挂载使用**

[root@localhost ~]# **blkid /dev/sdb1**  #查看文件系统类型

[root@localhost ~]# **mkdir /mypart1**

[root@localhost ~]# **mount /dev/sdb1 /mypart1**

[root@localhost ~]# **df -h** #显示正在挂载的设备信息

[root@localhost ~]# **mkdir /mypart2**

[root@localhost ~]# **mount /dev/sdb2 /mypart2**

[root@localhost ~]# **df -h**  #显示正在挂载的设备信息

[root@localhost ~]# **df -h | grep sdb**

**总结：**

**1.识别硬盘 lsblk**

**2.划分分区 MBR分区模式 fdisk**

**3.格式化文件系统** **mkfs.xfs mkfs.ext4 blkid**

**4.挂载使用 mount df -h**

**七、开机自动挂载/etc/fstab**

设备路径 挂载点   文件系统类型  参数 备份标记   检测顺序

[root@localhost ~]# **vim /etc/fstab** #vmware虚拟机

**/dev/sdb1 /mypart1 ext4 defaults 0 0**

**/dev/sdb2 /mypart2 xfs defaults 0 0**

[root@localhost ~]# **vim /etc/fstab**  #KVM虚拟机

**/dev/vdb1 /mypart1 ext4 defaults 0 0**

**/dev/vdb2 /mypart2 xfs defaults 0 0**

[root@localhost ~]# **umount /mypart1**  #卸载

[root@localhost ~]# **umount /mypart2**  #卸载

[root@localhost ~]# **df -h | grep mypart**

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

检测/etc/fstab开机自动挂载配置文件,格式是否正确

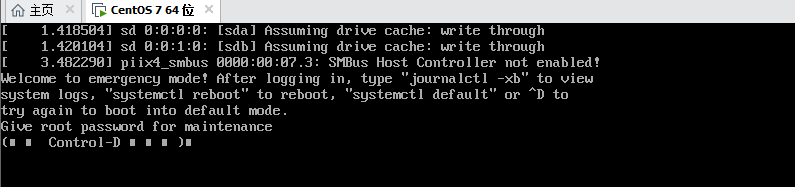
检测/etc/fstab中,书写完成,但当前没有挂载的设备,进行挂载

[root@localhost ~]# **df -h | grep mypart**

[root@localhost ~]# reboot #不是必须的

如果/etc/fstab文件有误：修复办法

1.输入root的密码



2.修改/etc/fstab文件内容

C:\Users\Python\AppData\Local\YNote\data\niubenlinux@163.com\09244c1a8bf449adaa13909084ae7e78\clipboard.png

**八、综合分区**

[root@localhost ~]# **fdisk /dev/sdb**

**p** 查看分区表

**n** 创建主分区--->**回车**--->**回车**--->**回车**--->在last结束时 **+2G**

**p** 查看分区表

**n** 创建扩展分区 --->**回车**--->起始**回车**--->结束**回车** **将所有剩余空间给扩展分区**

**p** 查看分区表

**n** 创建逻辑分区----->起始**回车**------>结束**+2G**

**n** 创建逻辑分区----->起始**回车**------>结束**+2G**

**p** 查看分区表

**w** 保存并退出

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

[root@localhost ~]# **partprobe** #刷新分区表

Warning: 无法以读写方式打开 /dev/sr0 (只读文件系统)。/dev/sr0 已按照只读方式打开。

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

九、总结

1.识别硬盘 lsblk

2.分区规划 fdisk MBR分区模式

3.刷新分区表 partprobe

4.格式化文件系统 mkfs.ext4 mkfs.xfs blkid

5.挂载 mount /etc/fstab mount -a df -h

**/dev/sda5表示含义**：SCSI接口设备第一块的第一个逻辑分区

**综合分区练习：针对/dev/sdc第三块硬盘**

划分**三个主分区**分别为5G 4G 2G

划分**二个逻辑分区**分别为 5G 2G

[root@localhost ~]# **fdisk /dev/sdc**

**p** 查看分区表

**n** 创建主分区--->**回车**--->**回车**--->**回车**--->在last结束时 **+5G**

**n** 创建主分区--->**回车**--->**回车**--->**回车**--->在last结束时 **+4G**

**n** 创建主分区--->**回车**--->**回车**--->**回车**--->在last结束时 **+2G**

**p** 查看分区表

**n** 创建扩展分区 --->**回车**--->起始**回车**--->结束**回车** **将所有剩余空间给扩展分区**

**p** 查看分区表

**n** 创建逻辑分区----->起始**回车**------>结束**+5G**

**n** 创建逻辑分区----->起始**回车**------>结束**+2G**

**p** 查看分区表

**w** 保存并退出

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

Gib（进率为1024）

Gb（进率为1000）

十、分区模式GPT

1.关闭计算机添加新的硬盘

[root@localhost ~]# poweroff

[root@localhost ~]# lsblk

2.分区模式GPT

•GPT，GUID Partition Table

–全局唯一标识分区表

–突破固定大小64字节的分区表限制

–最多可支持**128个主分区**，最大支持18EB容量

\*\* 1 EB = 1024 PB = 1024 x 1024 TB

•parted常用分区指令

–help    //查看指令帮助

–mktable  gpt    //建立指定模式分区表

–mkpart  分区的名称  文件系统类型  start  end

    //指定大小或百分比%作为起始、结束位置

–print   //查看分区表

–rm  序号    //删除指定的分区

–quit   //退出交互环境

•parted进行分区

[root@localhost ~]# **parted /dev/sdd**  #划分2个分区

(parted) **mktable gpt**  #指定分区模式

(parted) **mkpart** #划分新的分区

分区名称？ []? **haha** #随意写

文件系统类型？ [ext2]? **ext4**  #随意写，不会进行格式化

起始点？ **0**

结束点？ **5G**

忽略/Ignore/放弃/Cancel? **Ignore** #选择忽略，输入**i(tab)**补全

(parted) **print**  #查看分区表信息

(parted) **unit GB**  #使用GB作为单位

(parted) **print**

(parted) **mkpart**  #划分新的分区

分区名称？ []? **haha**

文件系统类型？ [ext2]? **ext4**

起始点？ **5G** #为上一个分区的结束

结束点？ **10G**

(parted) **print**

(parted) **quit**

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

**•分区的使用**

[root@nb ~]# **ls /dev/sdd1**

/dev/sdd1

[root@nb ~]# **mkfs.xfs /dev/sdd1**

[root@nb ~]# **blkid /dev/sdd1**

/dev/sdd1: UUID="6ee70a2d-26e5-4ed2-90d9-9a173bd8a9a8" **TYPE="xfs"** PARTLABEL="haha" PARTUUID="1e940194-adb4-4f33-921e-7c8c25f4c127"

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

**/dev/sdd1 /test xfs defaults 0 0**

[root@nb ~]# **mkdir /test**

[root@nb ~]# **mount -a #测试是否书写正确**

[root@nb ~]# **df -h | grep sdd1**

**十一、交换空间（虚拟内存**）

利用硬盘的空间，充当内存的空间

CPU----->内存----->硬盘

**当物理内存占满了，CPU可以将内存的中数据，暂时放入交换空间中，缓解真实物理内存的压力**

交换空间最好为内存的2倍，交换空间最大不会超过16G

**方式一：利用硬盘分区制作交换空间**

]# **ls /dev/sdd2**

]# **mkswap /dev/sdd2**  #格式化交换文件系统

]# **blkid /dev/sdd2**  #查看文件系统类型

]# **swapon**  #查看交换空间组成的成员信息

]# **swapon /dev/sdd2** #启用交换分区

]# **swapon**  #查看交换空间组成的成员信息

]# **free -m**  #查看交换空间的大小

]# **swapoff /dev/sdd2**  #停用交换分区

]# **swapon**  #查看交换空间组成的成员信息

]# **free -m** #查看交换空间的大小

* **开机自动启用交换分区**

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

**/dev/sdd2 swap swap defaults 0 0**

[root@localhost ~]# **swapoff /dev/sdd2** #停用

[root@localhost ~]# **swapon** #查看交换空间组成的成员信息

[root@localhost ~]# **swapon -a** #专门检测交换分区的书写

[root@localhost ~]# **swapon**  #查看交换空间组成的成员信息

方式二：利用文件方式

1.生成较大的文件

dd if=数据的源头 of=生成的文件 bs=每次读写数据的大小 count=次数

/dev/zero:拥有无限的数据

]# dd if=/dev/zero of=/opt/swap.txt bs=1M count=2048

]# ls -lh /opt/sw.txt

]# **mkswap /opt/swap.txt**

]# **swapon /opt/swap.txt**

swapon: /opt/swap.txt：不安全的权限 0644，建议使用 0600。

]# **swapon**

]# **vim /etc/fstab**

**/opt/swap.txt swap swap defaults 0 0**

]# **swapoff /opt/swap.txt #停用**

]# **swapon -a #检测**

]# **swapon #查看组成的成员**

课后的习题：

案例1：复制、粘贴、移动

以root用户新建/example/目录，在此目录下新建nsd.txt文件，并进一步完成下列操作

1）将“I love Study”写入到文件nsd.txt

2）将nsd.txt重命名为mylove.txt

3）将/etc/passwd、/boot、/etc/group同时拷贝到/example/目录下

4）将ifconfig命令的前两行内容，追加写入mylove.txt

5）将主机名永久配置文件（/etc/hostname），拷贝到/example/目录下

6）将DNS永久配置文件（/etc/resolv.conf），拷贝到/example/目录下

7）将开机自动挂载配置文件（/etc/fstab），拷贝到/example/目录下

案例2:虚拟机上操作,（MBR分区模式）规划分区

添加一块60G的硬盘并规划分区：

划分2个10G的主分区；1个12G的主分区;1个20G的逻辑分区。

案例3:虚拟机上操作,分区使用

1、案例2中新添加60G硬盘的第一个逻辑分区

– 格式化成xfs文件系统，实现该分区开机自动挂载，挂载点为/mnt/xfs

2、案例2中新添加60G硬盘的第一个主分区

– 完成开机自动挂载，挂载点/mnt/mypart,文件系统为ext4

案例4:虚拟机上操作,采用GPT分区模式，利用parted规划分区

添加一块20G的硬盘并规划分区：

划分2个2G的主分区；1个5G的主分区;

案例5:虚拟机上操作,交换分区使用

1、案例4中新添加20G硬盘的第一个主分区

– 格式化成交换文件系统，实现该分区开机自动启用

2、案例4中新添加20G硬盘的第二个主分区

– 格式化成交换文件系统，实现该分区开机自动启用