3.5 实战案例——LVM 逻辑卷的使用

3.5.1 案例目标

（1）了解 LVM 逻辑卷的安装。

（2）了解 LVM 逻辑卷的配置与使用。

3.5.2 案例分析

1. 规划节点

Linux 操作系统的单节点规划，见表 3-5-1。

表 3-5-1 节点规划

IP	主机名	节点
192.168.200.10	localhost	Linux 服务器节点

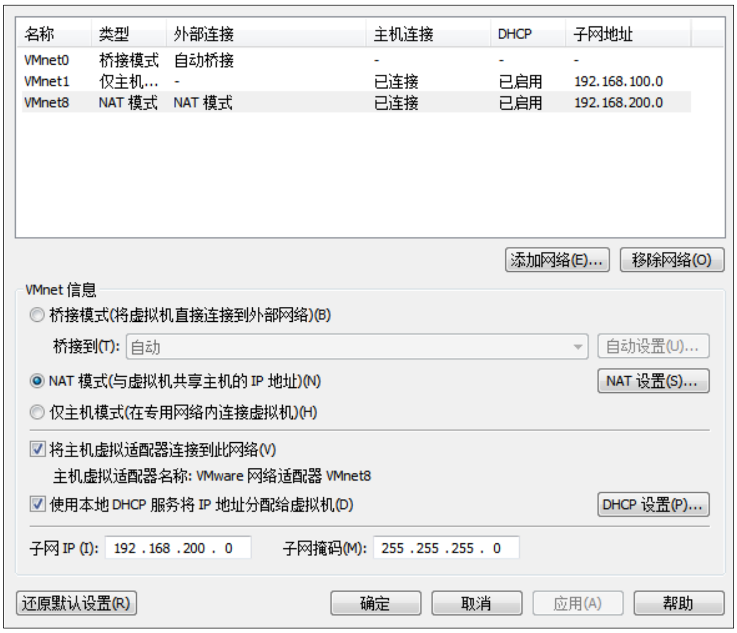
2. 基础准备

使用实战案例 3.3 安装的 Linux 系统进行下述实验。

3.5.3 案例实施

1. 配置 IP 地址

查看虚拟网络编辑器，查看本机 NAT 模式的网络信息，如图 3-5-1、图 3-5-2 所示。



名称	类型	外部连接	主机连接	DHCP	子网地址
VMnet0	桥接模式	自动桥接	-	-	-
VMnet1	仅主机...	-	已连接	已启用	192.168.100.0
VMnet8	NAT 模式	NAT 模式	已连接	已启用	192.168.200.0

VMnet 信息

○ 桥接模式(将虚拟机直接连接到外部网络)(B)

桥接到(T): 自动

● NAT 模式(与虚拟机共享主机的 IP 地址)(N)

○ 仅主机模式(在专用网络内连接虚拟机)(H)

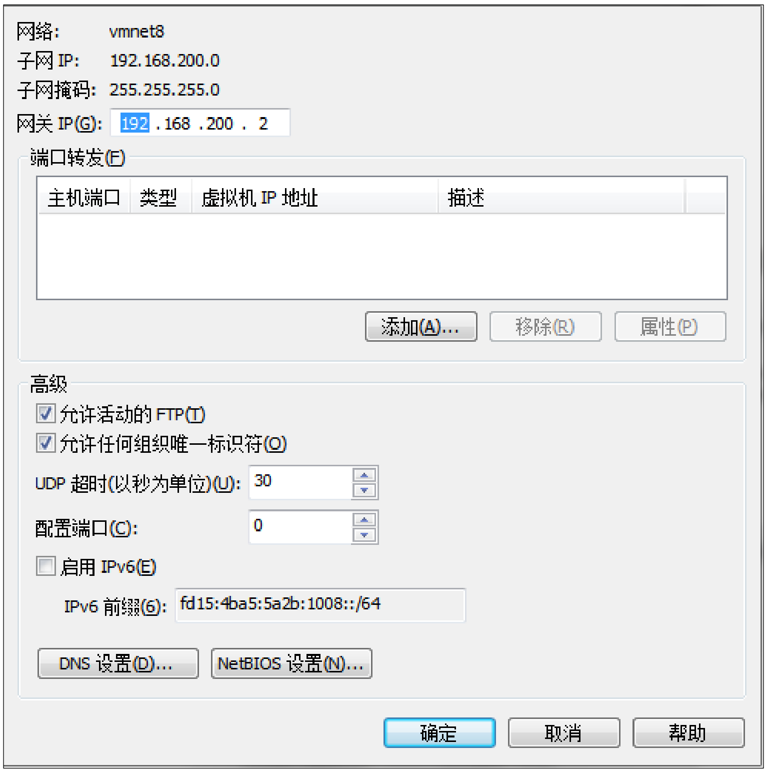
☑ 将主机虚拟适配器连接到此网络(V)

主机虚拟适配器名称: VMware 网络适配器 VMnet8

☑ 使用本地 DHCP 服务将 IP 地址分配给虚拟机(D)

子网 IP (I): 192 .168 .200 . 0　　子网掩码(M): 255 .255 .255 . 0

图 3-5-1 虚拟网络编辑器



网络: vmnet8
子网 IP: 192.168.200.0
子网掩码: 255.255.255.0
网关 IP(G): 192 . 168 . 200 . 2

端口转发(F)

主机端口	类型	虚拟机 IP 地址	描述

高级
☑ 允许活动的 FTP(T)
☑ 允许任何组织唯一标识符(O)
UDP 超时(以秒为单位)(U): 30
配置端口(C): 0
□ 启用 IPv6(E)
IPv6 前缀(6): fd15:4ba5:5a2b:1008::/64

图 3-5-2 NAT 设置详情

回到虚拟机界面，编辑网卡配置文件，将网络配置成 192.168.200.10，命令如下：

```
[root@localhost ~]# vi /etc/sysconfig/network-scripts/ifcfg-eno16777736

[root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-eno16777736

TYPE=Ethernet

BOOTPROTO=static

DEFROUTE=yes

PEERDNS=yes

PEERROUTES=yes

IPV4_FAILURE_FATAL=no

IPV6INIT=yes

IPV6_AUTOCONF=yes

IPV6_DEFROUTE=yes

IPV6_PEERDNS=yes

IPV6_PEERROUTES=yes

IPV6_FAILURE_FATAL=no
```

```
NAME=eno16777736
UUID=25acd229-1851-4454-9219-8dcee56b798c
DEVICE=eno16777736
ONBOOT=yes
IPADDR=192.168.200.10
NETMASK=255.255.255.0
GATEWAY=192.168.200.2
```

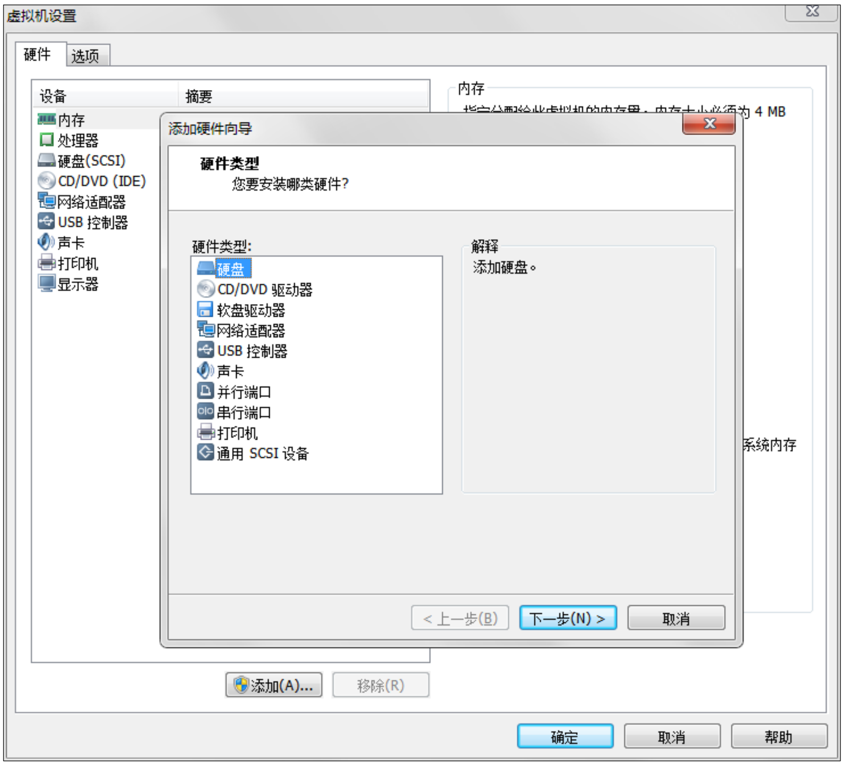
配置完成后，重启网络并查看 IP，命令如下：

```
[root@localhost ~]# systemctl restart network
[root@localhost ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eno16777736: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:0c:29:d6:48:b7 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.10/24 brd 192.168.200.255 scope global eno16777736
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fed6:48b7/64 scope link
       valid_lft forever preferred_lft forever
```

配置完 IP 后，可以通过 PC 机的远程连接工具 SecureCRT 连接虚拟机。

2. 添加硬盘

在 VMwareWorkstation 中的虚拟机设置界面，单击下方“添加”按钮，选择“硬盘”，然后单击右下角“下一步”按钮，如图 3-5-3 所示。


虚拟机设置

内存

指定分配给此虚拟机的内存量。内存大小必须为 4 MB

系统内存

添加硬件向导

硬件类型

您要安装哪类硬件?

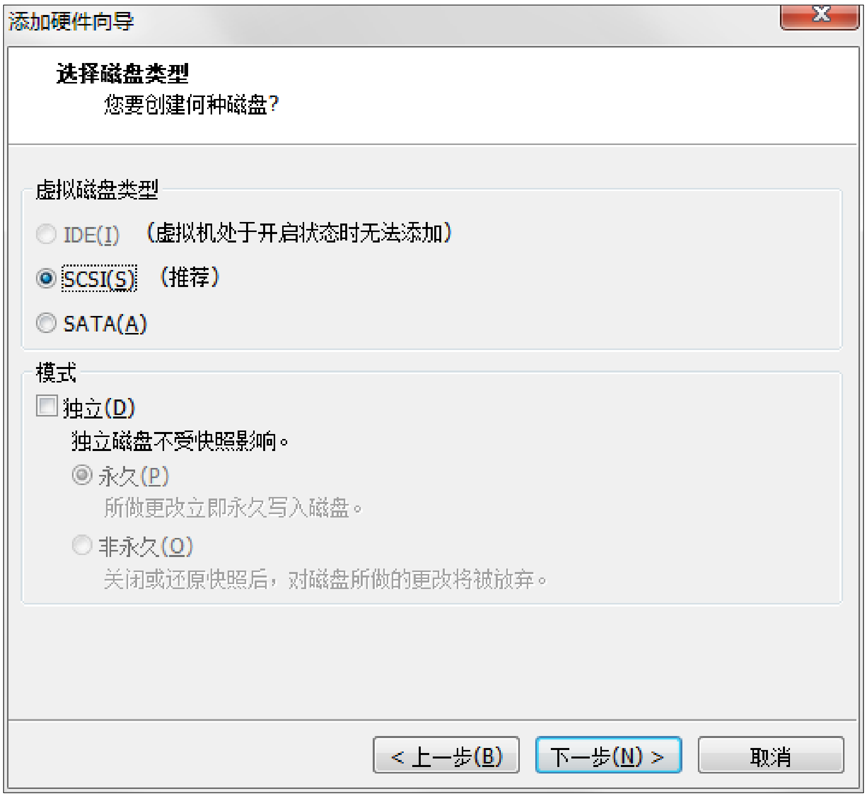
硬件类型:

解释

添加硬盘。

图 3-5-3 添加硬盘

选择 SCSI（S）磁盘，单击右下角“下一步”按钮，如图 3-5-4 所示。


添加硬件向导

选择磁盘类型

您要创建何种磁盘?

虚拟磁盘类型

○ IDE(I) （虚拟机处于开启状态时无法添加）

● SCSI(S) （推荐）

○ SATA(A)

模式

□ 独立(D)

独立磁盘不受快照影响。

● 永久(P)

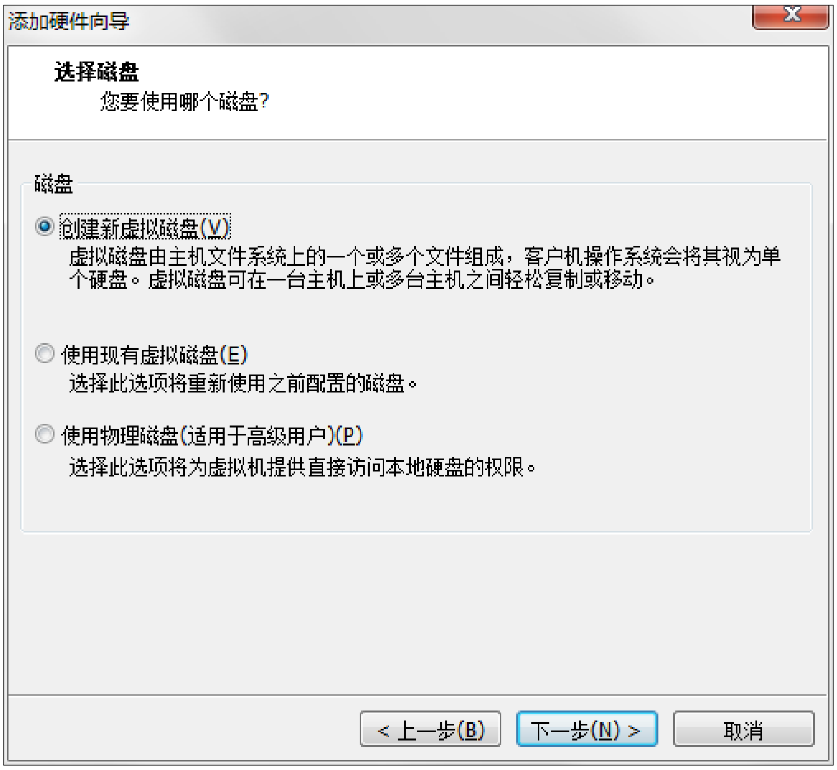
所做更改立即永久写入磁盘。

○ 非永久(O)

关闭或还原快照后，对磁盘所做的更改将被放弃。

图 3-5-4 选择磁盘类型

选择“创建新虚拟磁盘（V）”选项，然后单击右下角“下一步”按钮。如图 3-5-5 所示。



添加硬件向导

选择磁盘

您要使用哪个磁盘?

磁盘

● 创建新虚拟磁盘(V)

虚拟磁盘由主机文件系统上的一个或多个文件组成，客户机操作系统会将其视为单个硬盘。虚拟磁盘可在一台主机上或多台主机之间轻松复制或移动。

○ 使用现有虚拟磁盘(E)

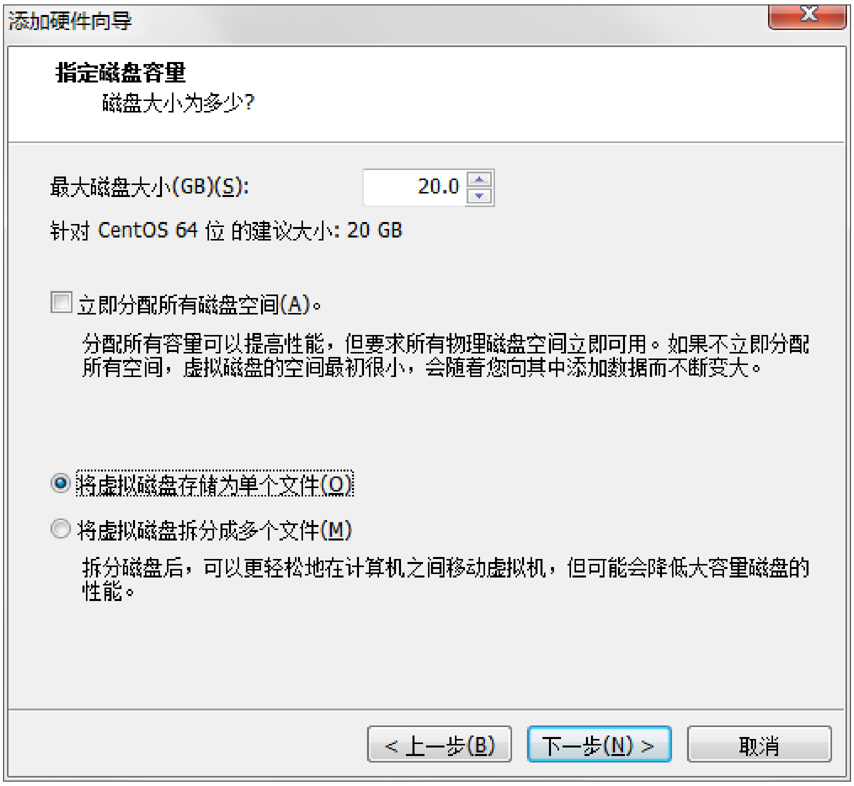
选择此选项将重新使用之前配置的磁盘。

○ 使用物理磁盘(适用于高级用户)(P)

选择此选项将为虚拟机提供直接访问本地硬盘的权限。

图 3-5-5 选择磁盘

指定磁盘大小为 20 GB，选择“将虚拟磁盘存储为单个文件（O）”选项，如图 3-5-6 所示。


添加硬件向导

指定磁盘容量

磁盘大小为多少?

最大磁盘大小(GB)(S): 20.0

针对 CentOS 64 位 的建议大小: 20 GB

□ 立即分配所有磁盘空间(A)。

分配所有容量可以提高性能，但要求所有物理磁盘空间立即可用。如果不立即分配所有空间，虚拟磁盘的空间最初很小，会随着您向其中添加数据而不断变大。

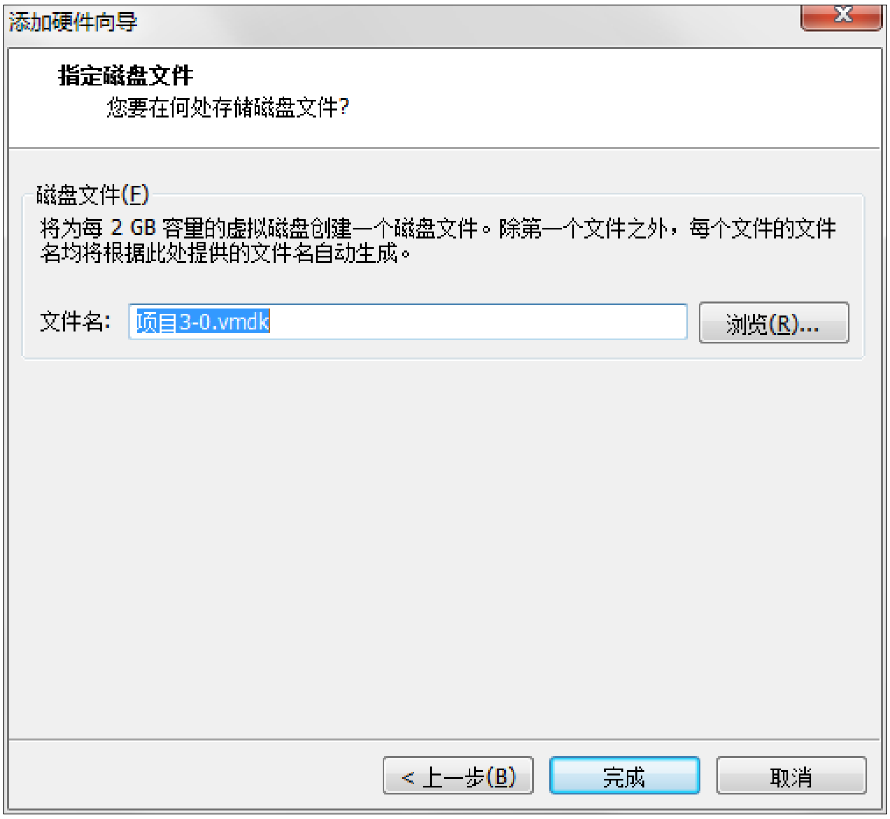
● 将虚拟磁盘存储为单个文件(O)

○ 将虚拟磁盘拆分成多个文件(M)

拆分磁盘后，可以更轻松地在计算机之间移动虚拟机，但可能会降低大容量磁盘的性能。

图 3-5-6 指定磁盘容量

文件名不做修改，使用默认名称，然后单击右下角“完成”按钮，如图 3-5-7 所示。


添加硬件向导

指定磁盘文件

您要在何处存储磁盘文件?

磁盘文件(F)

将为每 2 GB 容量的虚拟磁盘创建一个磁盘文件。除第一个文件之外，每个文件的文件名均将根据此处提供的文件名自动生成。

文件名: 项目3-0.vmdk

图 3-5-7 指定磁盘文件

添加完磁盘后，重启虚拟机。重启过后，使用命令查看磁盘，命令如下：

```
[root@localhost ~]# lsblk
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                       8:0      0     20G    0 disk
 ├─sda1                    8:1      0    500M    0 part /boot
 └─sda2                    8:2      0 19.5G    0 part
 ├─centos-root 253:0      0 17.5G    0 lvm    /
 └─centos-swap 253:1      0      2G    0 lvm    [SWAP]
sdb                       8:16     0     20G    0 disk
sr0                      11:0      1      4G    0 rom
```

可以看到存在一块名叫 sdb 的块设备，大小为 20 GB。

3. LVM 逻辑卷的使用

（1）创建物理卷

在创建物理卷之前，需要对磁盘进行分区。首先使用 fdisk 命令对 sdb 进行分区操作，分出两个大小为 5 GB 的分区，命令如下：

```
[root@localhost ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x9e46a7c2.
Command (m for help): p
Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x9e46a7c2
   Device Boot        Start           End         Blocks     Id    System
Command (m for help): n
Partition type:
    p    primary (0 primary, 0 extended, 4 free)

    e    extended

Select (default p): p

Partition number (1-4, default 1):

First sector (2048-41943039, default 2048):

Using default value 2048

Last sector, +sectors or +size{K,M,G} (2048-41943039, default 41943039): +5G

Partition 1 of type Linux and of size 5 GiB is set

Command (m for help): n

Partition type:

    p    primary (1 primary, 0 extended, 3 free)

    e    extended

Select (default p): p

Partition number (2-4, default 2):

First sector (10487808-41943039, default 10487808):

Using default value 10487808

Last sector, +sectors or +size{K,M,G} (10487808-41943039, default 41943039): +5G

Partition 2 of type Linux and of size 5 GiB is set

Command (m for help): p

Disk /dev/sdb: 21.5 GB, 21474836480 bytes, 41943040 sectors

Units = sectors of 1 * 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x9e46a7c2

    Device Boot      Start         End      Blocks   Id  System

/dev/sdb1             2048    10487807     5242880   83  Linux

/dev/sdb2         10487808    20973567     5242880   83  Linux

Command (m for help): w

The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.

Syncing disks.

[root@localhost ~]# lsblk

NAME                    MAJ:MIN RM    SIZE RO TYPE MOUNTPOINT

sda                       8:0       0      20G    0 disk

 ├─sda1                   8:1       0    500M    0 part /boot

 └─sda2                   8:2       0 19.5G    0 part

 ├─centos-root 253:0       0 17.5G    0 lvm    /

 └─centos-swap 253:1       0       2G    0 lvm    [SWAP]

sdb                       8:16      0      20G    0 disk

 ├─sdb1                   8:17      0       5G    0 part

 └─sdb2                   8:18      0       5G    0 part

sr0                      11:0       1       4G    0 rom
```

分完分区后，对这两个分区进行创建物理卷操作，命令如下：

```
[root@localhost ~]# pvcreate /dev/sdb1 /dev/sdb2

Physical volume "/dev/sdb1" successfully created

Physical volume "/dev/sdb2" successfully created
```

创建完毕后，可以查看物理卷的简单信息与详细信息，命令如下：

物理卷简单信息：

```
[root@localhost ~]# pvs

PV              VG        Fmt    Attr PSize    PFree

/dev/sda2    centos lvm2 a--    19.51g 40.00m

/dev/sdb1              lvm2 ---      5.00g    5.00g

/dev/sdb2              lvm2 ---      5.00g    5.00g
```

物理卷详细信息：

```
[root@localhost ~]# pvdisplay

--- Physical volume ---

PV Name                             /dev/sda2

VG Name                              centos

PV Size                           19.51 GiB / not usable 3.00 MiB
```

```
Allocatable           yes
PE Size               4.00 MiB
Total PE              4994
Free PE               10
Allocated PE          4984
PV UUID               78lNjC-ofC2-YQIH-O2RA-3ZHG-N8dM-d4Hve2
"/dev/sdb1" is a new physical volume of "5.00 GiB"
--- NEW Physical volume ---
PV Name               /dev/sdb1
VG Name
PV Size               5.00 GiB
Allocatable           NO
PE Size               0
Total PE              0
Free PE               0
Allocated PE          0
PV UUID               73POMD-3fHz-k0Gj-vG64-KAA0-qnzO-ZqdvnB
"/dev/sdb2" is a new physical volume of "5.00 GiB"
--- NEW Physical volume ---
PV Name               /dev/sdb2
VG Name
PV Size               5.00 GiB
Allocatable           NO
PE Size               0
Total PE              0
Free PE               0
Allocated PE          0
PV UUID               ImEUkD-dEb9-zvE3-gyO4-9kpN-MPCb-NchoSf
```

（2）创建卷组

使用刚才创建好的两个物理卷，创建名为 myvg 的卷组，命令如下：

```
[root@localhost ~]# vgcreate myvg /dev/sdb[1-2]
Volume group "myvg" successfully created
```

查看卷组信息，（可以查看到创建的 myvg 卷组，名字为 centos 的卷组是系统卷组，因为在安装系统的时候，是使用 LVM 模式安装的）命令如下：

```
[root@localhost ~]# vgs
VG       #PV #LV #SN Attr   VSize  VFree
centos     1   2   0 wz--n- 19.51g 40.00m
myvg       2   0   0 wz--n-  9.99g  9.99g
```

查看卷组详细信息，命令如下：

```
[root@localhost ~]# vgdisplay
--- Volume group ---
VG Name               centos
System ID
Format                lvm2
Metadata Areas        1
Metadata Sequence No  3
VG Access             read/write
VG Status             resizable
MAX LV                0
Cur LV                2
Open LV               2
Max PV                0
Cur PV                1
Act PV                1
VG Size               19.51 GiB
PE Size               4.00 MiB
Total PE              4994
Alloc PE / Size       4984 / 19.47 GiB
Free  PE / Size       10 / 40.00 MiB
VG UUID               2H21hv-L20X-mqhJ-SvzR-crT2-ln9v-jj2gTY
```

```
--- Volume group ---
VG Name                     myvg
System ID
Format                      lvm2
Metadata Areas              2
Metadata Sequence No        1
VG Access                   read/write
VG Status                   resizable
MAX LV                      0
Cur LV                      0
Open LV                     0
Max PV                      0
Cur PV                      2
Act PV                      2
VG Size                     9.99 GiB
PE Size                     4.00 MiB
Total PE                    2558
Alloc PE / Size             0 / 0
Free  PE / Size             2558 / 9.99 GiB
VG UUID                     PYGJuQ-s1Ix-ZwGf-kFaV-4Lfh-ooHl-QXcy6a
```

当多个物理卷组合成一个卷组后时，LVM 会在所有的物理卷上做类似格式化的工作，将每个物理卷切成一块一块的空间，这一块一块的空间就称为 PE（Physical Extent），它的默认大小是 4 MB。

由于受内核限制的原因，一个逻辑卷（Logic Volume）最多只能包含 65536 个 PE（Physical Extent），所以一个 PE 的大小就决定了逻辑卷的最大容量，4 MB 的 PE 决定了单个逻辑卷最大容量为 256 GB，若希望使用大于 256 Gb 的逻辑卷，则创建卷组时需要指定更大的 PE。

删除卷组，重新创建卷组，并指定 PE 大小为 16 MB，命令如下：

```
[root@localhost ~]# vgremove myvg
Volume group "myvg" successfully removed
[root@localhost ~]# vgcreate -s 16m myvg /dev/sdb[1-2]
```

```
Volume group "myvg" successfully created
[root@localhost ~]# vgdisplay
--- Volume group ---
VG Name               centos
System ID
Format                lvm2
Metadata Areas        1
Metadata Sequence No  3
VG Access             read/write
VG Status             resizable
MAX LV                0
Cur LV                2
Open LV               2
Max PV                0
Cur PV                1
Act PV                1
VG Size               19.51 GiB
PE Size               4.00 MiB
Total PE              4994
Alloc PE / Size       4984 / 19.47 GiB
Free  PE / Size       10 / 40.00 MiB
VG UUID               2H21hv-L20X-mqhJ-SvzR-crT2-ln9v-jj2gTY
--- Volume group ---
VG Name               myvg
System ID
Format                lvm2
Metadata Areas        2
Metadata Sequence No  1
VG Access             read/write
VG Status             resizable
```

```
MAX LV                    0
Cur LV                    0
Open LV                   0
Max PV                    0
Cur PV                    2
Act PV                    2
VG Size                   9.97 GiB
PE Size                   16.00 MiB
Total PE                  638
Alloc PE / Size           0 / 0
Free  PE / Size           638 / 9.97 GiB
VG UUID                   dU0pP2-EW9d-6c0h-8tgQ-t1bN-tBIo-FDqfdR
```

可以查看到现在 myvg 卷组的 PE 大小为 16 MB。

向卷组 myvg 中添加一个物理卷，在/dev/sdb 上再分一个/dev/sdb3 分区，把该分区加到卷组 myvg 中。命令如下：

```
[root@localhost ~]# lsblk
NAME                  MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda                     8:0    0    20G  0 disk
├─sda1                  8:1    0   500M  0 part /boot
└─sda2                  8:2    0  19.5G  0 part
├─centos-root 253:0    0 17.5G  0 lvm   /
└─centos-swap 253:1    0    2G  0 lvm   [SWAP]
sdb                     8:16   0    20G  0 disk
├─sdb1                  8:17   0     5G  0 part
├─sdb2                  8:18   0     5G  0 part
└─sdb3                  8:19   0     5G  0 part
sr0                    11:0    1     4G  0 rom
```

将创建的/dev/sdb3 添加到 myvg 卷组中，在添加的过程中，会自动将/dev/sdb3 创建为物理卷，命令如下：

```
[root@localhost ~]# vgextend myvg /dev/sdb3
```

可以查看到现在卷组中存在 3 个物理卷设备。

```
Physical volume "/dev/sdb3" successfully created
Volume group "myvg" successfully extended
[root@localhost ~]# vgs
VG       #PV #LV #SN Attr   VSize   VFree
centos     1   2   0 wz--n- 19.51g 40.00m
myvg       3   0   0 wz--n- 14.95g 14.95g
[root@localhost ~]# vgdisplay myvg
--- Volume group ---
VG Name               myvg
System ID
Format                lvm2
Metadata Areas        3
Metadata Sequence No  2
VG Access             read/write
VG Status             resizable
MAX LV                0
Cur LV                0
Open LV               0
Max PV                0
Cur PV                3
Act PV                3
VG Size               14.95 GiB
PE Size               16.00 MiB
Total PE              957
Alloc PE / Size       0 / 0
Free  PE / Size       957 / 14.95 GiB
VG UUID               dU0pP2-EW9d-6c0h-8tgQ-t1bN-tBIo-FDqfdR
```

可以查看到现在卷组中存在 3 个物理卷设备。

（3）创建逻辑卷

创建逻辑卷，名称为 mylv，大小为 5 GB。命令如下：

```
[root@localhost ~]# lvcreate -L +5G -n mylv myvg
Logical volume "mylv" created.
```

- -L：创建逻辑卷的大小 large。
- -n：创建的逻辑卷名称 name。

查看逻辑卷，命令如下：

```
[root@localhost ~]# lvs
LV      VG        Attr            LSize    Pool Origin Data%    Meta%    Move Log Cpy%Sync
Convert
root centos -wi-ao---- 17.47g
swap centos -wi-ao----    2.00g
mylv myvg      -wi-a-----    5.00g
```

扫描上一步创建的 lv 逻辑卷。命令如下：

```
[root@localhost ~]# lvscan
ACTIVE                    '/dev/centos/root' [17.47 GiB] inherit
ACTIVE                    '/dev/centos/swap' [2.00 GiB] inherit
ACTIVE                    '/dev/myvg/mylv' [5.00 GiB] inherit
```

使用 ext4 文件系统格式化逻辑卷 mylv。命令如下：

```
[root@localhost ~]# mkfs.ext4 /dev/mapper/myvg-mylv
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
327680 inodes, 1310720 blocks
65536 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1342177280
40 block groups
```

```
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736
Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

把逻辑卷 mylv 挂载到/mnt 下并验证。命令如下：

```
[root@localhost ~]# mount /dev/mapper/myvg-mylv /mnt/
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   18G  872M   17G   5% /
devtmpfs                 1.9G     0  1.9G   0% /dev
tmpfs                    1.9G     0  1.9G   0% /dev/shm
tmpfs                    1.9G  8.6M  1.9G   1% /run
tmpfs                    1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1                497M  125M  373M  25% /boot
tmpfs                    378M     0  378M   0% /run/user/0
/dev/mapper/myvg-mylv    4.8G   20M  4.6G   1% /mnt
```

然后对创建的 LVM 卷扩容至 1 GB。

```
[root@localhost ~]# lvextend -L +1G /dev/mapper/myvg-mylv
Size of logical volume myvg/mylv changed from 5.00 GiB (320 extents) to 6.00 GiB (384
extents).
Logical volume mylv successfully resized.
[root@localhost ~]# lvs
LV   VG     Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync
Convert
root centos -wi-ao---- 17.47g
```

```
swap centos -wi-ao----    2.00g
mylv myvg     -wi-ao----    6.00g
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   18G  872M   17G   5% /
devtmpfs                 1.9G     0  1.9G   0% /dev
tmpfs                    1.9G     0  1.9G   0% /dev/shm
tmpfs                    1.9G  8.6M  1.9G   1% /run
tmpfs                    1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1                497M  125M  373M  25% /boot
tmpfs                    378M     0  378M   0% /run/user/0
/dev/mapper/myvg-mylv    4.8G   20M  4.6G   1% /mnt
```

可以查看到 LVM 卷的大小变成了 6 GB，但是挂载信息中没有发生变化，这时系统还识别不了新添加的磁盘文件系统，所以还需要对文件系统进行扩容。

```
[root@localhost ~]# resize2fs /dev/mapper/myvg-mylv
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/myvg-mylv is mounted on /mnt; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/mapper/myvg-mylv is now 1572864 blocks long.
[root@localhost ~]# df -h
Filesystem               Size  Used Avail Use% Mounted on
/dev/mapper/centos-root   18G  872M   17G   5% /
devtmpfs                 1.9G     0  1.9G   0% /dev
tmpfs                    1.9G     0  1.9G   0% /dev/shm
tmpfs                    1.9G  8.6M  1.9G   1% /run
tmpfs                    1.9G     0  1.9G   0% /sys/fs/cgroup
/dev/sda1                497M  125M  373M  25% /boot
tmpfs                    378M     0  378M   0% /run/user/0
/dev/mapper/myvg-mylv    5.8G   20M  5.5G   1% /mnt
```

扩容逻辑卷成功。