

## Working with RAM disks (optional)

### 1. Verify the size of the RAM disks that the kernel uses.

执行 lsblk 命令可以查看系统上所有的块设备（仅截取部分）：

```
tux1@tux1-virtual-machine:~$ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
fd0          2:0    1    1.4M  0 disk
loop0        7:0     0     4K   1 loop /snap/bare/5
loop1        7:1     0   63.3M  1 loop /snap/core20/1879
loop2        7:2     0   63.5M  1 loop /snap/core20/1891
loop3        7:3     0    73M   1 loop /snap/core22/617
loop4        7:4     0   242M   1 loop /snap/firefox/2667
loop5        7:5     0  241.5M  1 loop /snap/firefox/2605
loop6        7:6     0   73.1M  1 loop /snap/core22/634
loop7        7:7     0    6.7M  1 loop /snap/gedit/664
loop8        7:8     0  349.7M  1 loop /snap/gnome-3-38-2004/137
loop9        7:9     0  349.7M  1 loop /snap/gnome-3-38-2004/140
loop10       7:10    0  460.6M  1 loop /snap/gnome-42-2204/102
loop11       7:11    0   91.7M  1 loop /snap/gtk-common-themes/1535
loop12       7:12    0  460.6M  1 loop /snap/gnome-42-2204/99
loop13       7:13    0   45.9M  1 loop /snap/snap-store/599
loop14       7:14    0   45.9M  1 loop /snap/snap-store/638
loop15       7:15    0   53.2M  1 loop /snap/snapd/18933
loop16       7:16    0   53.2M  1 loop /snap/snapd/19122
loop17       7:17    0   428K   1 loop /snap/snapd-desktop-integration/57
loop18       7:18    0   452K   1 loop /snap/snapd-desktop-integration/83
sda          8:0     0    20G   0 disk
├─sda1       8:1     0     1M   0 part
├─sda2       8:2     0   513M   0 part /boot/efi
└─sda3       8:3     0  19.5G   0 part /var/snap/firefox/common/host-hunspell/
sr0         11:0     1  141.3M  0 rom  /media/tux1/CDROM
```

后续查询资料知，内存盘的设备名以 ram 开头，随后跟一个数字，但此处 lsblk 列举的结果中并无这样的内存盘设备，因此内核使用的内存盘的大小为 0。

### 2. Format the ramdisk with mke2fs.

首先用 modprobe 命令创建一个 ramdisk 设备，modprobe 是 linux 的一个命令，可载入指定的个别模块，或载入一组相依的模块。brd 是一个 Linux 内核模块，该命令可以创建一个基于内存的块设备，即 ramdisk。此处创建一个大小为 100MB 的 ramdisk：

```
tux1@tux1-virtual-machine:~$ sudo modprobe brd rd_size=102400
```

使用 mke2fs 命令将 ramdisk 设备写入文件系统：

```
tux1@tux1-virtual-machine:~$ sudo mke2fs /dev/ram1
[sudo] password for tux1:
mke2fs 1.46.5 (30-Dec-2021)
/dev/ram1 contains a ext4 file system
    created on Sat May 20 23:25:10 2023
Proceed anyway? (y,N) y
Creating filesystem with 16384 4k blocks and 16384 inodes

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done
```

ramdisk 设备/dev/ram1 格式化完成。

### 3.Create a directory /mnt/ramdisk, and mount the RAM disk on that mountpoint.

首先执行 mkdir 命令创建目录/mnt/ramdisk:

```
tux1@tux1-virtual-machine:~$ sudo mkdir /mnt/ramdisk
```

接下来执行 mount 命令将/dev/ram1 挂载到目录/mnt/ramdisk 上:

```
tux1@tux1-virtual-machine:~$ sudo mount /dev/ram1 /mnt/ramdisk
```

### 4.Run the df command to see how much space is available on this RAM disk.

执行 df 命令:

```
tux1@tux1-virtual-machine:~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
tmpfs            398216      2148    396068   1% /run
/dev/sda3       19946096 15183092   3724464  81% /
tmpfs           1991068        0   1991068   0% /dev/shm
tmpfs            5120         4     5116   1% /run/lock
/dev/sda2       524252      5364   518888   2% /boot/efi
tmpfs           398212      2404   395808   1% /run/user/1000
/dev/sr1        3976468 3976468        0 100% /media/tux1/Ubuntu 22.10 amd64
/dev/sr0        144726   144726        0 100% /media/tux1/CDROM
/dev/ram1        61412        24     58112   1% /mnt/ramdisk
```

如上图所示, /dev/ram1 磁盘上的可用空间为 58112KB。

### 5.Copy some files onto this ramdisk, and verify that they are indeed there.

执行 cp 命令将主目录上的 vitest 文件和 getest 文件复制到该 ramdisk 上:

```
tux1@tux1-virtual-machine:~$ sudo cp vitest /mnt/ramdisk
[sudo] password for tux1:
tux1@tux1-virtual-machine:~$ sudo cp getest /mnt/ramdisk
```

利用 ls 命令验证这两个文件是否存在:

```
tux1@tux1-virtual-machine:~$ ls /mnt/ramdisk
getest  lost+found  vitest
```

两文件均在输出结果中列出, 证明这两个文件均存在。

### 6.Unmount the ramdisk and mount it again. Is the file still there?

使用 umount 命令可卸载 ramdisk, 可以通过设备名也可以通过挂载点卸载:

```
tux1@tux1-virtual-machine:~$ sudo umount /dev/ram1
```

随后执行 mount 命令将/dev/ram1 挂载到目录/mnt/ramdisk 上:

```
tux1@tux1-virtual-machine:~$ sudo mount /dev/ram1 /mnt/ramdisk
```

利用 ls 命令验证这两个文件是否存在:

```
tux1@tux1-virtual-machine:~$ ls /mnt/ramdisk
getest  lost+found  vitest
```

两文件均在输出结果中列出, 说明这两个文件仍然存在。

### 7.Reboot your system and then try to mount the RAM disk again. Does this work?

重新启动系统, 随后使用 mount 命令再次挂载 ramdisk:

```
tux1@tux1-virtual-machine:~$ sudo mount /dev/ram1 /mnt/ramdisk
[sudo] password for tux1:
mount: /mnt/ramdisk: special device /dev/ram1 does not exist.
dmesg(1) may have more information after failed mount system call.
```

命令报错: 内存盘/dev/ram1 不存在。



## Working with partitions

If you installed your system correctly, then you should have at least 450 MB in unpartitioned space left. We're going to create four partitions in here, 150 MB each, which are going to be used in the subsequent exercises.

8. Start the fdisk program and create four additional partitions. The partition size should be 150 MB each. Save the partition table to disk. Write down the partition numbers you created, because you will be using these in the rest of this exercise.

在/dev/sda3 中创建 4 个额外的分区，启动 fdisk 程序，该程序可以创建和维护分区表：

```
root@tux1-virtual-machine:/home/tux1# fdisk /dev/sda3

Welcome to fdisk (util-linux 2.38).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.

The device contains 'ext4' signature and it will be removed by a write command.
See fdisk(8) man page and --wipe option for more details.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xaa69da84.
```

输入 n，新建一个分区，再输入 p，选择分区类型为主分区（若想添加扩展分区，则输入 e），随后输入 1，选择分区号为 1（将扩展分区包含在内，主分区只有 4 个，因此只能选择 1-4，按顺序输入分区号）。

```
Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-40886271, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-40886271, default 40886271):
: +150M

Created a new partition 1 of type 'Linux' and of size 150 MiB.
```

随后按相同步骤创建剩余 3 个大小为 150MB 的分区，分区号分别为 2、3、4：

```
Command (m for help): n
Partition type
  p   primary (1 primary, 0 extended, 3 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (309248-40886271, default 309248):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (309248-40886271, default 40886271):
: +150M

Created a new partition 2 of type 'Linux' and of size 150 MiB.
```

```

Command (m for help): n
Partition type
  p   primary (2 primary, 0 extended, 2 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (3,4, default 3): 3
First sector (616448-40886271, default 616448):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (616448-40886271, default 40886271): +150M

Created a new partition 3 of type 'Linux' and of size 150 MiB.

```

```

Command (m for help): w
The partition table has been altered.
Failed to add partition 1 to system: Invalid argument
Failed to add partition 2 to system: Invalid argument
Failed to add partition 3 to system: Invalid argument
Failed to add partition 4 to system: Invalid argument

The kernel still uses the old partitions. The new table will be used at the next reboot.
Syncing disks.

```

9.Ensure the changes to the partition table have been modified in the running kernel by executing the `partprobe` command.

使用 `partprobe` 命令来通知内核重新加载分区表，以确保在运行的内核中执行了对分区表的更改：

```

root@tux1-virtual-machine:/home/tux1# partprobe /dev/sda3
Error: Partition(s) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 on /dev/sda3 have been written, but we have been unable to inform the kernel of the change, probably because it/they are in use. As a result, the old partition(s) will remain in use. You should reboot now before making further changes.

```

命令执行失败。

10.Format the first of the four partitions with the `mke2fs` command. Create a mountpoint `/mnt/partition` and mount this partition on this mountpoint.

使用 `mke2fs` 命令将第一个分区格式化：

```

root@tux1-virtual-machine:/home/tux1# mke2fs /dev/sda3p1
mke2fs 1.46.5 (30-Dec-2021)
The file /dev/sda3p1 does not exist and no size was specified.

```

命令执行失败。报错：/dev/sda3p1 不存在。

11.Run the `df` command to see how much space is available on these partitions.

执行 `df` 命令，输出结果如下：

```

root@tux1-virtual-machine:/home/tux1# df

```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
tmpfs	398216	2048	396168	1%	/run
/dev/sda3	19946096	15191816	3715740	81%	/
tmpfs	1991072	0	1991072	0%	/dev/shm
tmpfs	5120	4	5116	1%	/run/lock
/dev/sda2	524252	5364	518888	2%	/boot/efi
tmpfs	398212	2412	395800	1%	/run/user/1000
/dev/sr1	3976468	3976468	0	100%	/media/tux1/Ubuntu 22.10 amd64
/dev/sr0	144726	144726	0	100%	/media/tux1/CDROM

由于 4 个新分区为成功创建，故 `df` 命令未成功显示这几个分区。

## 12. Copy some files onto these partitions and verify that they are indeed there.

执行 cp 命令将主目录上的 vitest 文件复制到/dev/sda3p1 上，将 getset 文件复制到/dev/sda3p2 上：

```
root@tux1-virtual-machine:/home/tux1# cp vitest /dev/sda3p1
root@tux1-virtual-machine:/home/tux1# cp getest /dev/sda3p2
```

利用 ls 命令验证这些文件是否存在：

```
root@tux1-virtual-machine:/home/tux1# ls /dev/sda3p1
/dev/sda3p1
root@tux1-virtual-machine:/home/tux1# ls /dev/sda3p2
/dev/sda3p2
```

这些文件未列出，故不存在。

## 13. Reboot your system and then try to mount the partition again. Does this work?

重启系统后使用 mount 命令将/dev/sda3p1 挂载到/mnt/partition 上，命令执行失败，报错：/dev/sda3p1 不存在。

正常情况下分区应该能够重新挂载，由于没有在/etc/fstab 文件中配置自动挂载的选项，因此重启系统，之前挂载的分区可能会自动卸载。

## 14. Unmount the partition you just mounted.

使用 umount 命令可以卸载刚刚挂载的分区，可以通过设备名也可以通过挂载点卸载：

```
root@tux1-virtual-machine:/home/tux1# umount /mnt/partition
umount: /mnt/partition: not mounted.
```

命令报错：/mnt/partition 未挂载（不存在至少有 450MB 大小的未分区空间，/dev/sda3 已分区）。

## Preparing for RAID

15. With fdisk, change the partition types of the four partitions to 0xfd (Linux RAID autodetect). Use partprobe to re-read the partition table and, if necessary, reboot your system afterwards.

执行 fdisk 命令进入 fdisk 程序：

```
root@tux1-virtual-machine:/home/tux1# fdisk /dev/sda3

Welcome to fdisk (util-linux 2.38).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.


This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.


The device contains 'ext4' signature and it will be removed by a write command.
See fdisk(8) man page and --wipe option for more details.


Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x5ea662ce.
```



执行 `t` 命令将 4 个分区的分区类型更改为 `0xfd`（此处仅截取一个分区）：

```
Partition number (1-4, default 4):  
Command (m for help): t  
Partition number (1-4, default 4): 1  
Hex code or alias (type L to list all): 0xfd  
Changed type of partition 'Linux' to 'Linux raid autodetect'.
```

输入 `w` 保存并退出后，执行 `partprobe` 程序重新读取分区表：

```
root@tux1-virtual-machine:/home/tux1# partprobe /dev/sda3
```

## Working with RAID

**16. Create a RAID array using three partitions in a linear array. Use the `mdadm` command to initialize the array, and then create a file system on it. Mount the array on a mountpoint called `/mnt/raid`. Run the `df` command to find out how much space this partition has, and view the status of the RAID array. Then, unmount the partition and stop the RAID array.**

使用 `mdadm` 命令利用前三个分区创建一个名为 `/dev/md0` 的设备，阵列级别为 RAID0，选项 `-C` 表示创建模式，`-v` 表示创建过程中的详细信息，`-n*` 表示使用 \* 个块设备来创建此 RAID，`-l*` 表示要创建的 RAID 的级别，执行结果如下：

```
root@tux1-virtual-machine:/home/tux1# mdadm -Cv /dev/md0 -l0 -n3 /dev/sda3p1 /dev/sda3p2 /dev/sda3p3  
mdadm: cannot open /dev/sda3p1: No such file or directory
```

命令执行失败，报错：`/dev/sda3p1` 不存在（不存在至少有 450MB 大小的未分区空间，`/dev/sda3` 已分区）。

**Notice:** In order to watch what happens during the steps below, such as the `--create` or `--manage` options, open another terminal and type:

```
watch -n1 -d cat /proc/mdstat
```

Now adjust your other terminal window so you can type in it and see the results of the `watch` commands.

**17. Create a new RAID array using four partitions in a RAID-0 array. Use the `mdadm` command to initialize the array, and then create a file system on it. Mount the array on a mountpoint called `/mnt/raid`. Run the `df` command to find out how much space this partition has, and view the status of the RAID array. Then, unmount the partition and stop the RAID array.**

使用 `mdadm` 命令利用四个分区创建一个名为 `/dev/md1` 的设备，阵列级别为 RAID0，选项 `-C` 表示创建模式，`-v` 表示创建过程中的详细信息，`-n*` 表示使用 \* 个块设备来创建此 RAID，`-l*` 表示要创建的 RAID 的级别，执行结果如下：

```
root@tux1-virtual-machine:/home/tux1# mdadm -Cv /dev/md1 -l0 -n4 /dev/sda3p1 /dev/sda3p2 /dev/sda3p3 /dev/sda3p4  
mdadm: cannot open /dev/sda3p1: No such file or directory
```

命令执行失败，报错：`/dev/sda3p1` 不存在（不存在至少有 450MB 大小的未分区空间，`/dev/sda3` 已分区）。

**18. Create a new RAID array using three partitions in a RAID-1 array and one more used as a spare. Use the mdadm command to initialize the array, and then create a file system on it. Mount the array on a mountpoint called /mnt/raid. Run the df command to find out how much space this partition has, and view the status of the RAID array. Set one or two disks in the array to faulty and watch the array recover itself using the spare disk. Then, unmount the partition and stop the RAID array.**

使用 mdadm 命令利用三个分区和另一个分区作为备用分区 (/dev/sda3p4) 创建一个名为/dev/md2 的设备, 阵列级别为 RAID1, 选项-C 表示创建模式, -v 表示创建过程中的详细信息, -n\*表示使用\*个块设备来创建此 RAID, -l\*表示要创建的 RAID 的级别, -x\*表示备用分区的个数, 执行结果如下:

```
root@tux1-virtual-machine:/home/tux1# mdadm -Cv /dev/md2 -l1 -n3 /dev/sda3p1 /dev/sda3p2 /dev/sda3p3 -x1 /dev/sda3p4
mdadm: cannot open /dev/sda3p1: No such file or directory
```

命令执行失败, 报错: /dev/sda3p1 不存在。可以使用 mkfs 命令在 RAID 阵列上建立文件系统, 此处建立 ext4 文件系统:

```
root@tux1-virtual-machine:/home/tux1# mkfs.ext4 /dev/md2
mke2fs 1.46.5 (30-Dec-2021)
The file /dev/md2 does not exist and no size was specified.
```

命令执行失败, 报错: /dev/md2 不存在 (不存在至少有 450MB 大小的未分区空间, /dev/sda3 已分区)。

**19. Create a new RAID array using three partitions in a RAID-5 array and one more used as a spare. Use the mdadm command to initialize the array, and then create a file system on it. Mount the array on a mountpoint called /mnt/raid. Run the df command to find out how much space this partition has, and view the status of the RAID array. Set one or two disks in the array to faulty and watch the array recover itself using the spare disk. Then, unmount the partition and stop the RAID array.**

使用 mdadm 命令利用三个分区和另一个分区作为备用分区 (/dev/sda3p4) 创建一个名为/dev/md3 的设备, 阵列级别为 RAID1, 选项-C 表示创建模式, -v 表示创建过程中的详细信息, -n\*表示使用\*个块设备来创建此 RAID, -l\*表示要创建的 RAID 的级别, -x\*表示备用分区的个数, 执行结果如下:

```
root@tux1-virtual-machine:/home/tux1# mdadm -Cv /dev/md3 -l5 -n3 /dev/sda3p1 /dev/sda3p2 /dev/sda3p3 -x1 /dev/sda3p4
mdadm: layout defaults to left-symmetric
mdadm: cannot open /dev/sda3p1: No such file or directory
```

命令执行失败, 报错: /dev/sda3p1 不存在。可以使用 mkfs 命令在 RAID 阵列上建立文件系统, 此处建立 ext4 文件系统:

```
root@tux1-virtual-machine:/home/tux1# mkfs.ext4 /dev/md3
mke2fs 1.46.5 (30-Dec-2021)
The file /dev/md3 does not exist and no size was specified.
```

命令执行失败, 报错: /dev/md3 不存在, 使用 mount 命令将 RAID 设备挂

载到/mnt/raid 上:

```
root@tux1-virtual-machine:/home/tux1# mount /dev/md3 /mnt/raid
mount: /mnt/raid: special device /dev/md3 does not exist.
dmesg(1) may have more information after failed mount system call.
```

命令执行失败, 报错: /dev/md3 不存在。(不存在至少有 450MB 大小的未分区空间, /dev/sda3 已分区)。

### Working with LVM

**20. With fdisk, change the partition types of the four partitions to 0x8e (Linux LVM). Try using partprobe and, if necessary, reboot your system afterwards.**

使用 fdisk 命令进入 fdisk 程序:

```
root@tux1-virtual-machine:/home/tux1# fdisk /dev/sda3

Welcome to fdisk (util-linux 2.38).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

This disk is currently in use - repartitioning is probably a bad idea.
It's recommended to umount all file systems, and swapoff all swap
partitions on this disk.

The device contains 'ext4' signature and it will be removed by a write command.
See fdisk(8) man page and --wipe option for more details.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x5ea662ce.
```

执行 t 命令将四个分区的分区类型修改为 0xfd (此处仅截取一个分区的修改过程):

```
Command (m for help): t
Partition number (1-4, default 4): 1
Hex code or alias (type L to list all): 0x8e

Changed type of partition 'Linux' to 'Linux LVM'.
```

输入 w 保存并退出后, 执行 partprobe 程序重新读取分区表:

```
root@tux1-virtual-machine:/home/tux1# partprobe /dev/sda3
```

**21. Initialize each of the four partitions that you used earlier as physical volumes. If you get the error message /etc/lvmtab does not exist, then run the vgscan command to create it.**

使用 pvcreate 命令在四个分区上创建物理卷:

```
root@tux1-virtual-machine:/home/tux1# pvcreate /dev/sda3p1 /dev/sda3p2 /dev/sda3p3 /dev/sda3p4
No device found for /dev/sda3p1.
No device found for /dev/sda3p2.
No device found for /dev/sda3p3.
No device found for /dev/sda3p4.
```

命令执行失败, 报错: /dev/sda3p1, /dev/sda3p2, /dev/sda3p3, /dev/sda3p4 不存在。



**22.Create a volume group vg00 consisting of only the first two LVM partitions. Use a PE size of 4 MB.**

使用 `vgcreate` 命令创建卷组 `vg00`，并将上一步创建的两个物理卷加入到卷组中，`-s` 选项指定 PE 大小为 4MB：

```
root@tux1-virtual-machine:/home/tux1# vgcreate -s 4M vg00 /dev/sda3p1 /dev/sda3p2
No device found for /dev/sda3p1.
No device found for /dev/sda3p2.
```

命令执行失败，报错：`/dev/sda3p1`，`/dev/sda3p2` 不存在。

**23.Create a logical volume lv00 of 50 MB in the volume group vg00.**

使用 `lvcreate` 命令创建逻辑卷 `lv00`，并指定卷组 `vg00` 与逻辑卷大小为 50MB，其中 `-L` 选项表示逻辑卷大小为 50MB，`-n` 选项表示逻辑卷名称为 `lv00`：

```
root@tux1-virtual-machine:/home/tux1# lvcreate -L 50M -n lv00 vg00
Volume group "vg00" not found
Cannot process volume group vg00
```

命令执行失败，报错：卷组 `vg00` 不存在。

**24.Format the logical volume with the mke2fs command. Create a mount point /mnt/lv00 and mount the logical volume. Then, run the df command to see how much space is available.**

使用 `mke2fs` 命令格式化逻辑卷：

```
root@tux1-virtual-machine:/home/tux1# mke2fs /dev/vg00/lv00
mke2fs 1.46.5 (30-Dec-2021)
The file /dev/vg00/lv00 does not exist and no size was specified.
```

命令执行失败，报错：`/dev/vg00/lv00` 不存在。

**25.Use the various LVM commands to retrieve information about the physical volumes, the volume group, and the logical volumes.**

执行 `pydisplay`，`pvs`，`pvscan` 命令可以查看物理卷的详细信息、简要信息与扫描系统中的物理卷：

```
root@tux1-virtual-machine:/home/tux1# pvdisplay
root@tux1-virtual-machine:/home/tux1# pvs
root@tux1-virtual-machine:/home/tux1# pvscan
No matching physical volumes found
```

由于本次实验未真正创建物理卷，故无输出结果。

**26.Take a look at the files that were created in /etc/lvm. Can you read these files?**

使用 `ls` 命令查看在 `/etc/lvm` 中的文件：

```
root@tux1-virtual-machine:/home/tux1# ls /etc/lvm
lvml.conf  lvmllocal.conf  profile
```

其中有文件 `lvml.conf`，`lvmllocal.conf` 与 `profile`，这些是 LVM 的配置文件。

27. Add the third and fourth LVM partitions to the volume group vg00, and migrate all data onto these physical volumes. Then, reduce the volume group so that the volume group only contains the third and fourth partition. Do you need to unmount the /dev/vg00/lv00 logical volume first?

使用 vgextend 命令将第三、四个物理卷扩展到卷组 vg00 中：

```
root@tux1-virtual-machine:/home/tux1# vgextend vg00 /dev/sda3p3 /dev/sda3p4
No device found for /dev/sda3p3.
No device found for /dev/sda3p4.
```

命令执行失败，报错：/dev/sda3p3，/dev/sda3p4 不存在。

28. Add the first two LVM partitions back to your volume group, and create three more logical volumes in this volume group, called lv01 through lv03. Each logical volume needs to be 50 MB as well. We will need these in the next exercise.

使用 vgextend 命令将前两个物理卷重新扩展到卷组 vg00 中：

```
root@tux1-virtual-machine:/home/tux1# vgextend vg00 /dev/sda3p1 /dev/sda3p2
No device found for /dev/sda3p1.
No device found for /dev/sda3p2.
```

命令执行失败，报错：/dev/sda3p1，/dev/sda3p2 不存在。使用 lvcreate 命令创建逻辑卷 lv01-lv03，并指定卷组 vg00 与逻辑卷大小为 50MB，其中 -L 选项表示逻辑卷大小为 50MB，-n 选项表示逻辑卷名称为 lv01，lv02 与 lv03：

```
root@tux1-virtual-machine:/home/tux1# lvcreate -L 50M -n lv01 vg00
Volume group "vg00" not found
Cannot process volume group vg00
root@tux1-virtual-machine:/home/tux1# lvcreate -L 50M -n lv02 vg00
Volume group "vg00" not found
Cannot process volume group vg00
root@tux1-virtual-machine:/home/tux1# lvcreate -L 50M -n lv03 vg00
Volume group "vg00" not found
Cannot process volume group vg00
```

29. Take a look at the output of the pvscan command. Does this seem helpful?

执行 pvscan 命令：

```
root@tux1-virtual-machine:/home/tux1# pvscan
No matching physical volumes found
```

由于本次实验受虚拟环境影响，并未真正创建物理卷，因此命令执行失败，报错：未发现物理卷。

**注：**由于本次实验环境选择 VMware 虚拟机上的 Ubuntu 操作系统，并非实验要求的 Fedora、RHEL 或 SLES 操作系统，受虚拟环境限制，因此该命令执行结果存在不同（无内存盘），所有实验结果均为实验过程中实际显示。