Hands-on Projects

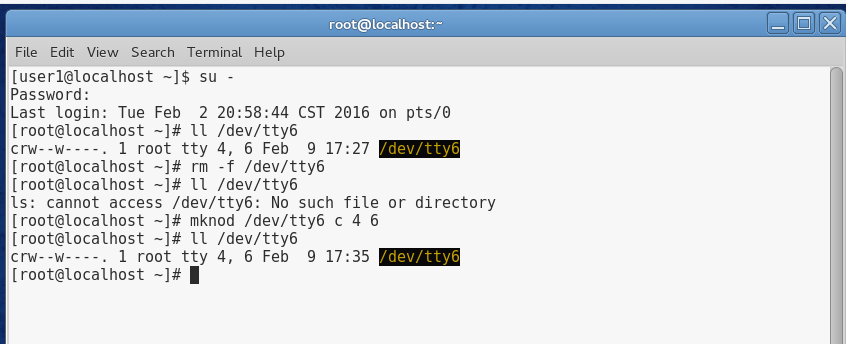
These projects should be completed in the order given. The hands-on projects presented in this chapter should take a total of three hours to complete. The requirements for this lab include:

* A computer with Fedora Linux installed according to Hands-on Project 2-1.

Project 5-1

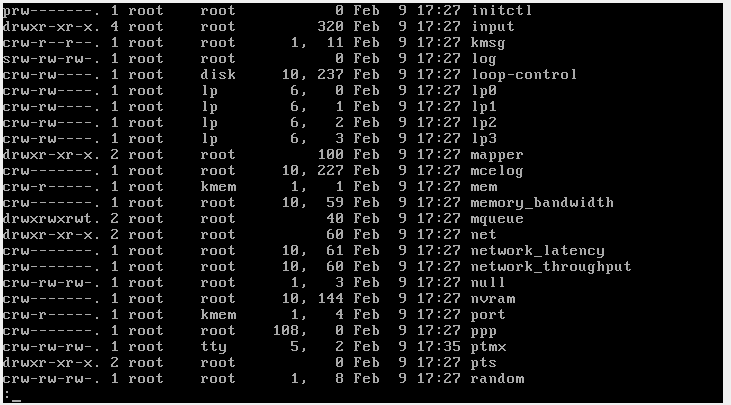
In this hands-on project, you view and create device files.

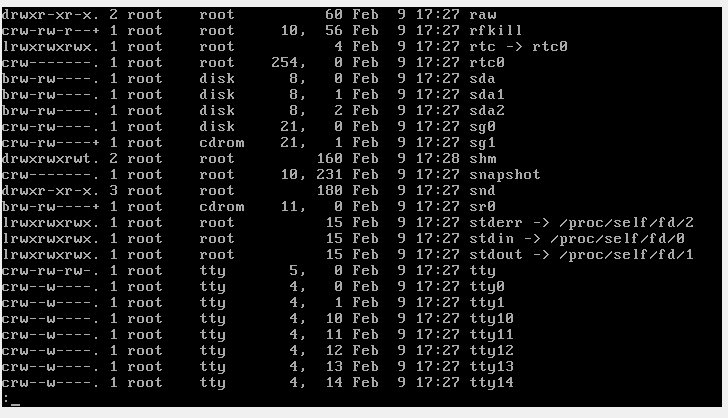
1. Boot your Fedora Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
3. At the command prompt, type **ll /dev/tty6** and press Enter. Recall that tty are the different terminals that you can access using **Ctrl+Alt+F#** keys. You should notice that tty is a character device. This makes sense since we input data to the terminal one character at a time. Observe the major and minor numbers.
4. At the command prompt, type **rm -f /dev/tty6** and press Enter. Next, type **ll /dev/tty6** at the command prompt and press Enter. The file should have been removed.
5. Switch to the command-line terminal (tty6) by pressing **Ctrl+Alt+F6** and attempt to log in to the terminal. Switch back to the graphical display (tty0) by pressing **Ctrl+Alt+F1**, type the command **mknod /dev/tty6 c 4 6** at the command prompt, and press Enter. What did this command do? Next, type **ll /dev/tty6** at the command prompt and press Enter. You should see the recreated device file.
6. **Provide screenshot(s) of steps 3 through 5.**

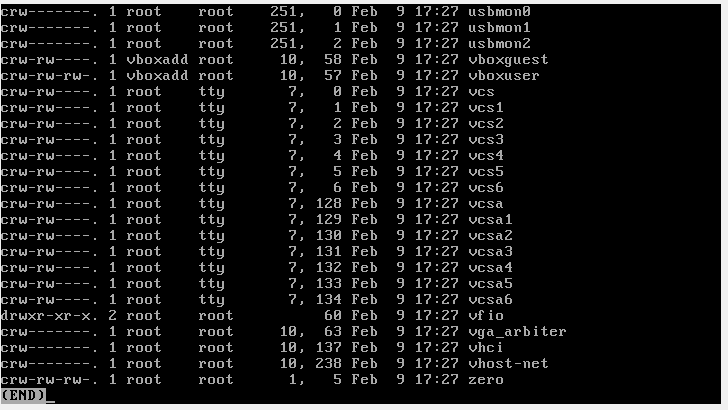


1. Switch to a command-line terminal (tty6) by pressing **Ctrl+Alt+F6** and log in to the terminal using the user name of **root** and the password of **LNXrocks!**. You should be able to use this terminal after recreating the file.
2. At the command prompt, type **ll /dev/tty?** and press Enter. Review the different terminal devices and their associated major and minor numbers.
3. At the command prompt, type **ll /dev | less** and press Enter to list all of the filenames underneath the /dev directory. Next, type **du -sh /dev** at the command prompt and press Enter. You will see that the device directory is not large at all.
4. At the command prompt, type **cat /proc/devices | less** and press Enter. You can determine major numbers from this file and correlate this to the information provided in step 8.
5. **Provide screenshot(s) of steps 7 through 10.**







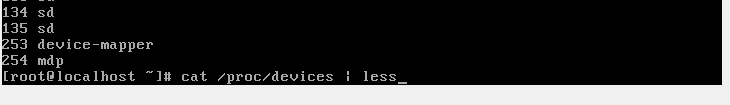








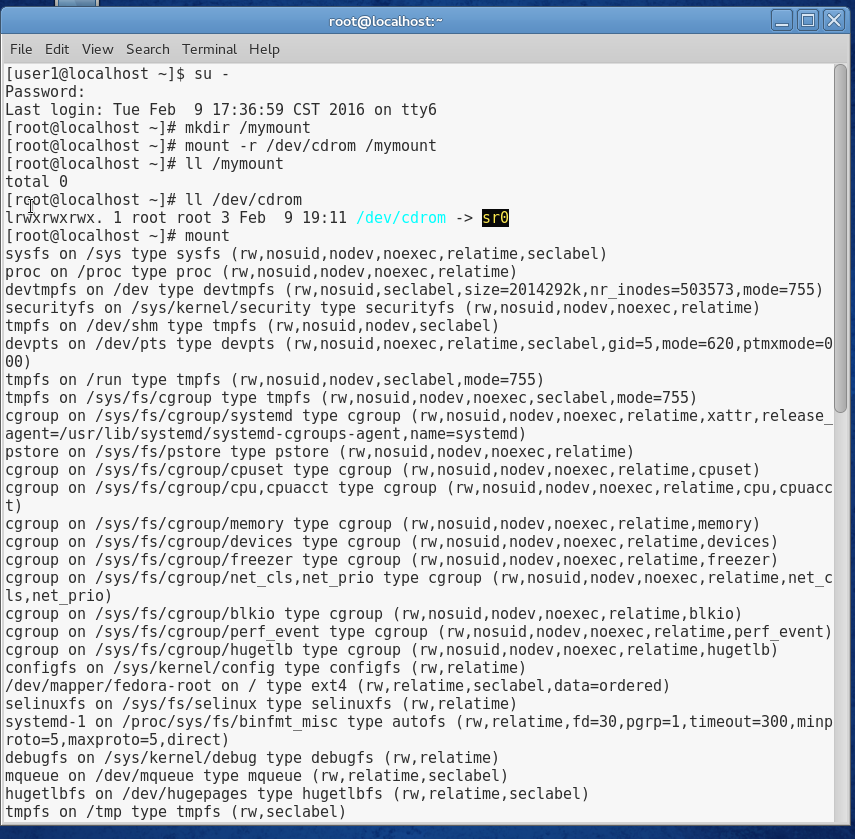


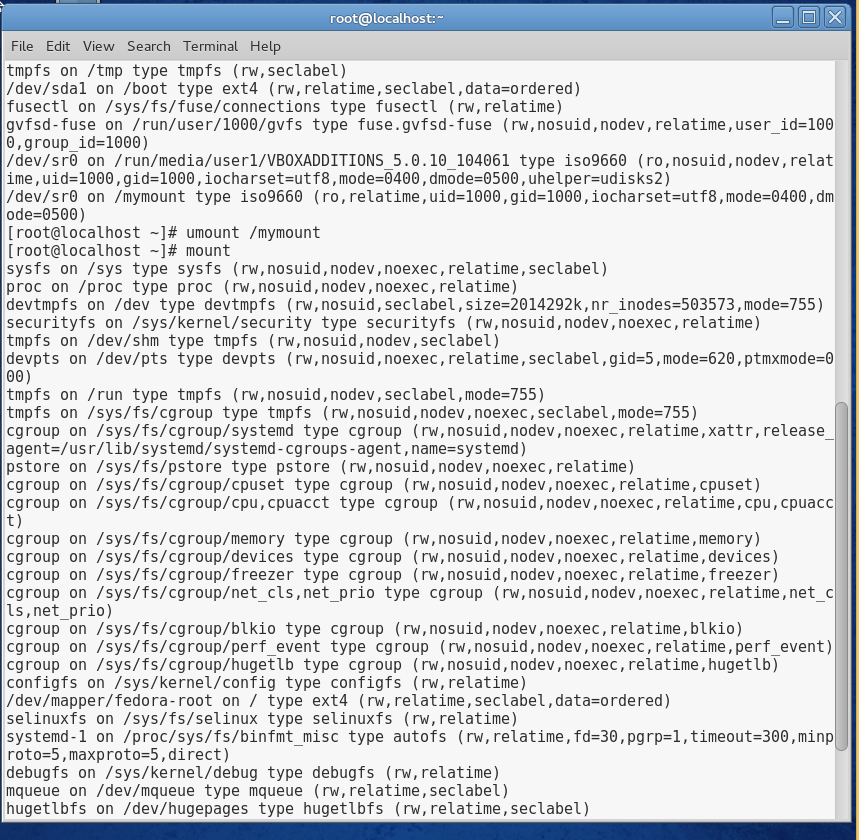


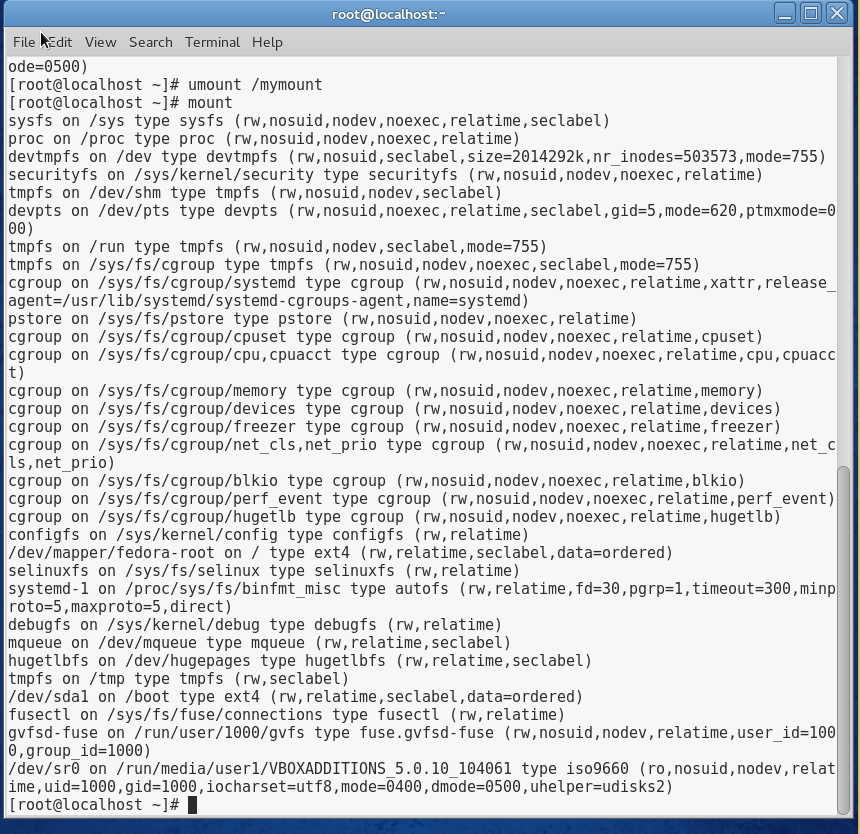
## Project 5-2

In this hands-on project, you mount DVDs to the directory tree and view their contents.

1. In VirtualBox settings for your VM, attach the ISO image for Fedora Linux (Fedora-Live-Desktop-x86\_64-20-1.iso) to the virtual CDROM drive as you did when we first installed Fedora. You can do this while your Fedora system is running.
2. Your desktop environment should prompt you that the ISO was inserted and you can open this up with the Files application. If you are not prompted, go to your activities screen and search for Files.
3. Take a few moments to explore the contents of the DVD within the Files application. When finished, click the eject button next to the Fedora Live Desktop DVD icon to eject the DVD.
4. Open up a terminal and change to the **root** user with password of **LNXrocks!**.
5. At the command prompt, type **mkdir /mymount** and press Enter to create a new mount point directory.
6. Next, type **mount -r /dev/cdrom /mymount** at the command prompt, and press Enter. We have mounted the ISO file as read only using the provided option.
7. At the command prompt, type **ll /mymount** and press Enter. You should see the directories seen in step 2.
8. At the command prompt, type **ll /dev/cdrom** and press Enter. You will see that this is a symlink to the actual CDROM device /dev/sr0.
9. At the command prompt, type **mount** and press Enter. Note the line at the bottom that lists your mounted DVD.
10. Next, type **umount /mymount** at the command prompt and press Enter. Type the **mount** command at a command prompt and press Enter to verify the ISO was unmounted.
11. In your virtualization software, detach the DVD ISO image for Fedora Linux (Fedora-Live-Desktop-x86\_64-20-1.iso) to the DVD drive for the virtual machine. This performs the same action as ejecting the DVD from the physical DVD drive.
12. **Provide screenshot(s) of steps 2 through 10.**



mk



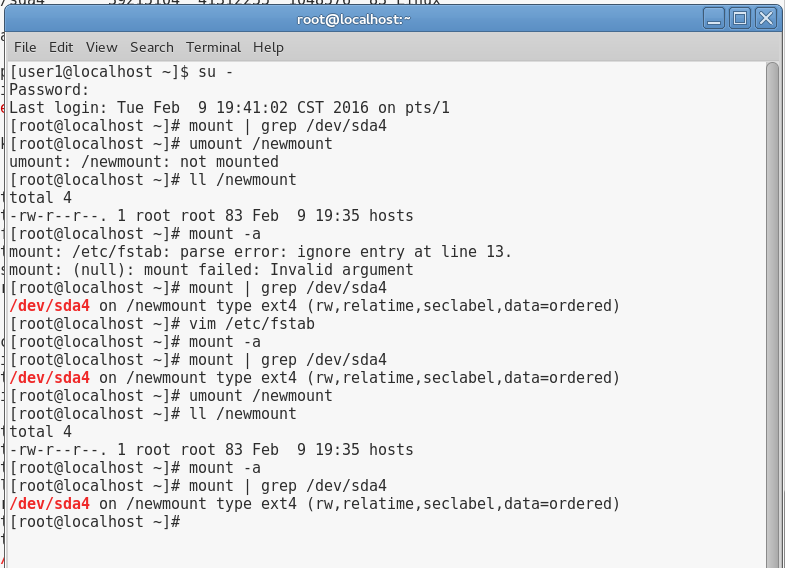
## Project 5-3

In this hands-on project, you work with standard hard disk partitions. You will first create a hard disk partition using the fdisk utility. Next, you create an ext4 filesystem on the partition and mount it to the directory tree. Finally, you use the /etc/fstab file to automatically mount the partition at boot time.

1. Boot your Fedora Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
3. Your hard drive should be referenced by **/dev/sda** but to double-check type **fdisk –l** and press Enter in a terminal. You should see /dev/sda as the disk and 2 partitions /dev/sda1 and /dev/sda2. /dev/sda2 should be Linux LVM.
4. At the command prompt, type **fdisk /dev/sda** and press Enter. You are now in the fdisk utility. Type **m** and press Enter to view the options of the fdisk utility. We should have at least 5GB of disk space free so we want to create 2 new partitions with this free space.
5. At the command prompt, type **n** and press Enter. You should have 2 free primary partitions to use. The default should say **p** so hit Enter to take the default. Otherwise, type **p** and press Enter. Your partition number should default to **3** press Enter to proceed. Press Enter to take the default first sector. Type **+4G** for the last sector and press Enter. You should see that output that you created your 3rd partition.
6. Now we will create the last partition we need. At the command prompt, type **n** and press Enter. You should have 1 free primary partitions to use. Type **p** and press Enter. Your partition number should default to **4** press Enter to proceed. Press Enter to take the default first sector. Type **+1G** for the last sector and press Enter. You should see that output that you created your 4th partition.
7. Type p to print the partitions. You should have sda1 through 4 created. Type **w** to write the changes we just made and to exit the fdisk utility.
8. At the command prompt, type **partprobe** and press Enter. At the command prompt, type **mkfs –t ext4 /dev/sda4** and press Enter.
9. At the command prompt, type **mkdir /newmount** and press Enter to create a mount point directory underneath the / directory for mounting the new filesystem we created.
10. At the command prompt, type **cp /etc/hosts /newmount** and press Enter. At the command prompt type **ll /newmount** and you will see the file we just copied to the directory.
11. At the command prompt, type **mount –t ext4 /dev/sda4 /newmount** and press Enter. At the command prompt, type **mount | grep /dev/sda4** and press Enter. You will see that the new filesystem is mounted to /newmount.
12. At the command prompt, type **ll /newmount** and observe that you no longer see the hosts file. This is because we mounted /dev/sda4 over the /newmount directory and now see the contents of /dev/sda4. The hosts file still exists and we would have to unmount this directory to access it again.
13. In order to have this filesystem mount to the /newmount directory upon reboot of the system, we need to add this to the /etc/fstab file. At the command prompt, type **vim /etc/fstab** and press Enter. Add the following entry to the bottom of the file. Once added write the contents and exit vim.

/dev/sda4 /newmount ext4 defaults 0 0

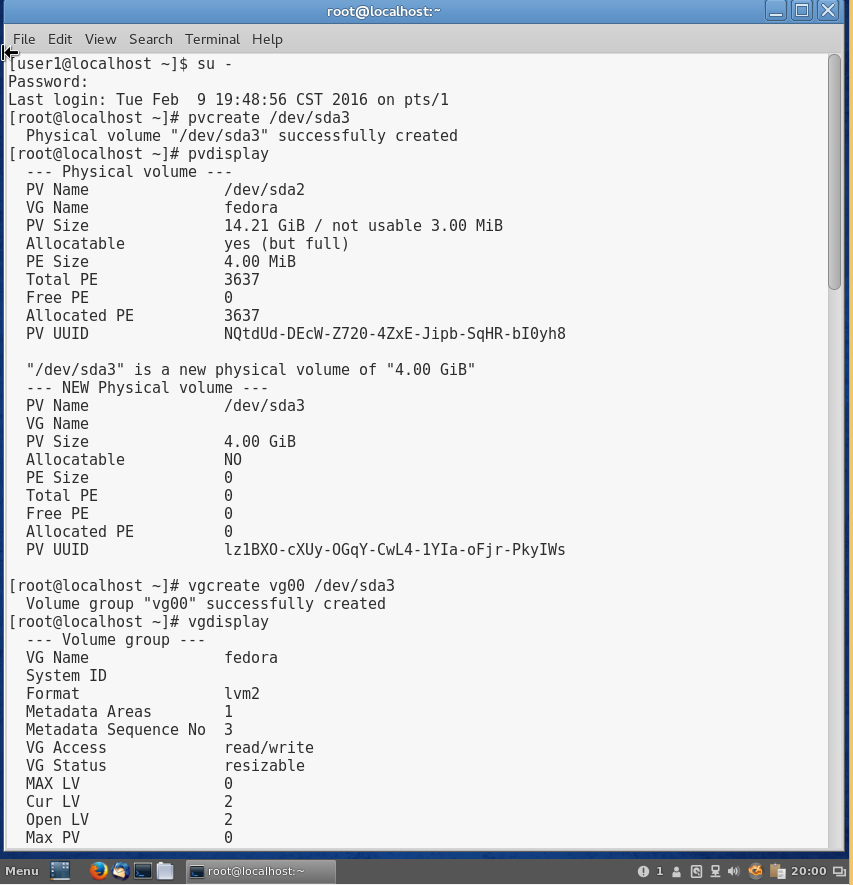
1. Log into your system and open a terminal and become the **root** user.
2. At the command prompt, type **mount | grep /dev/sda4** and press Enter. You should see that /dev/sda4 is mounted. If it is not, review the previous steps to determine why.
3. At the command prompt, type **umount /newmount** and press Enter. Next, type the **ll /newmount** command and you should see the hosts file.
4. At the command prompt, type **mount -a** and press Enter. Next, type the **mount | grep /dev/sda4** command and press Enter. The partition should be mounted since the **-a** option reads the /etc/fstab file and mounts those directories.
5. **Provide screenshot(s) of steps 15 through 17.**

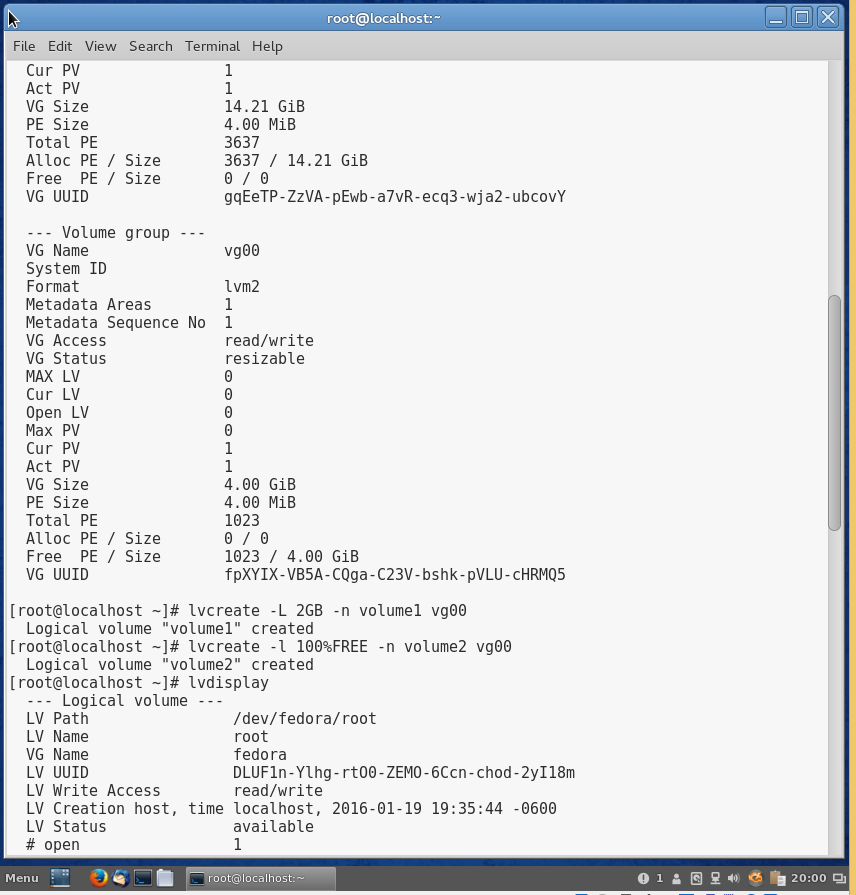
****

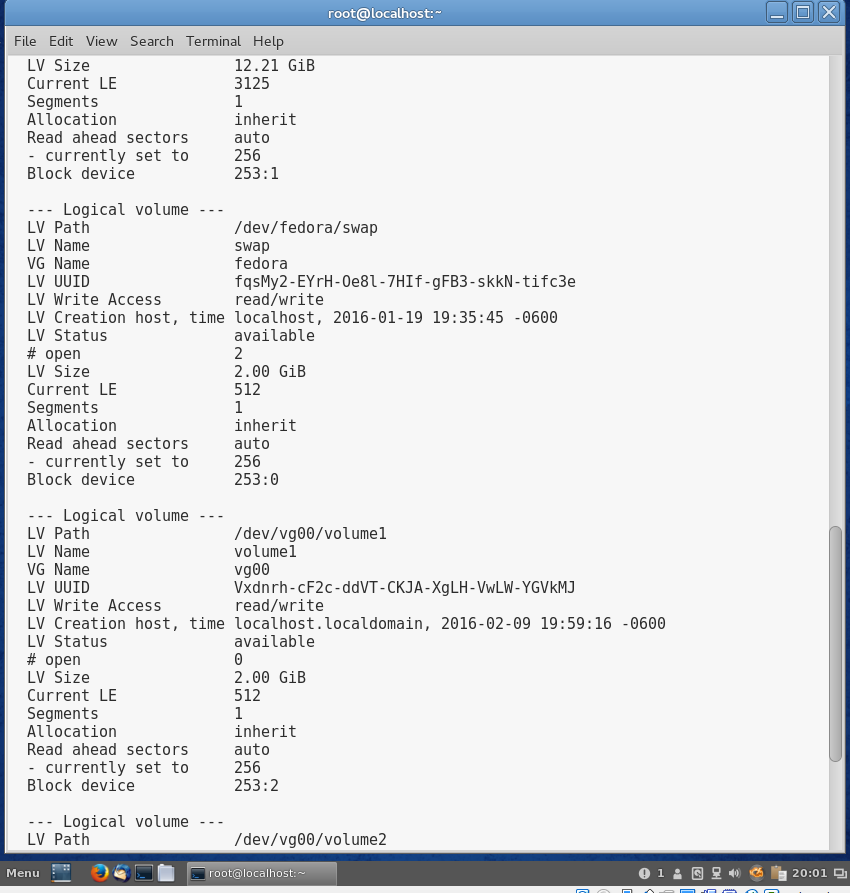
## Project 5-4

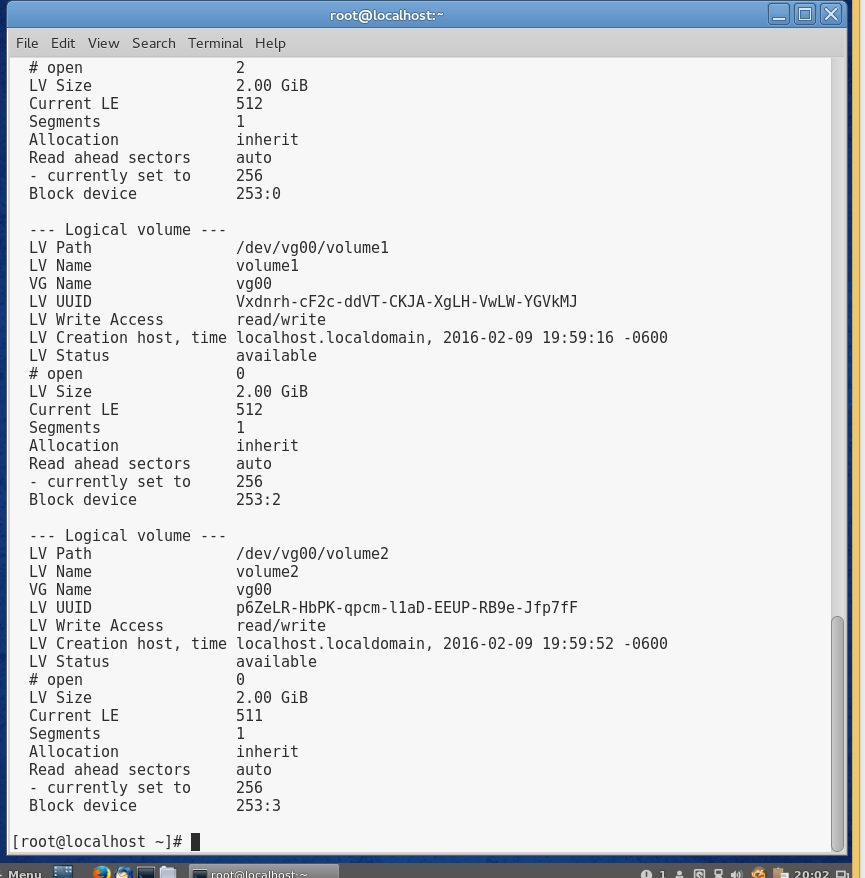
In this hands-on project, you create a new partition using the GNU Parted utility, and configure the LVM to host two logical volumes using the space within. Next, you will format these logical volumes and mount them to the directory tree, as well as edit the /etc/fstab file to ensure that they are mounted at boot time.

1. Boot your Fedora Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
3. At the command prompt, type **pvcreate /dev/sda3** and press Enter. We just added a new physical volume for use with Logical Volume Manager (LVM).
4. At the command prompt, type **pvdisplay** and press Enter. You should see /dev/sda3 added as a new physical volume.
5. At the command prompt, type **vgcreate vg00 /dev/sda3** and press Enter. We have now created a new volume group named vg00.
6. At the command prompt, type **vgdisplay** and press Enter. You should see the new vg00 with 4GB of space available.
7. At the command prompt, type **lvcreate –L 2GB –n volume1 vg00** and press Enter to create a 2GB logical volume called volume1 from the vg00 volume group.
8. At the command prompt, type **lvcreate –l 100%FREE –n volume2 vg00** and press Enter to create a logical volume called volume2 from the vg00 volume group with the rest of the available space.
9. At the command prompt, type **lvdisplay** and press Enter. You should see the new volumes created in the vg00 volume group.
10. **Provide screenshot(s) of steps 3 through 9.**

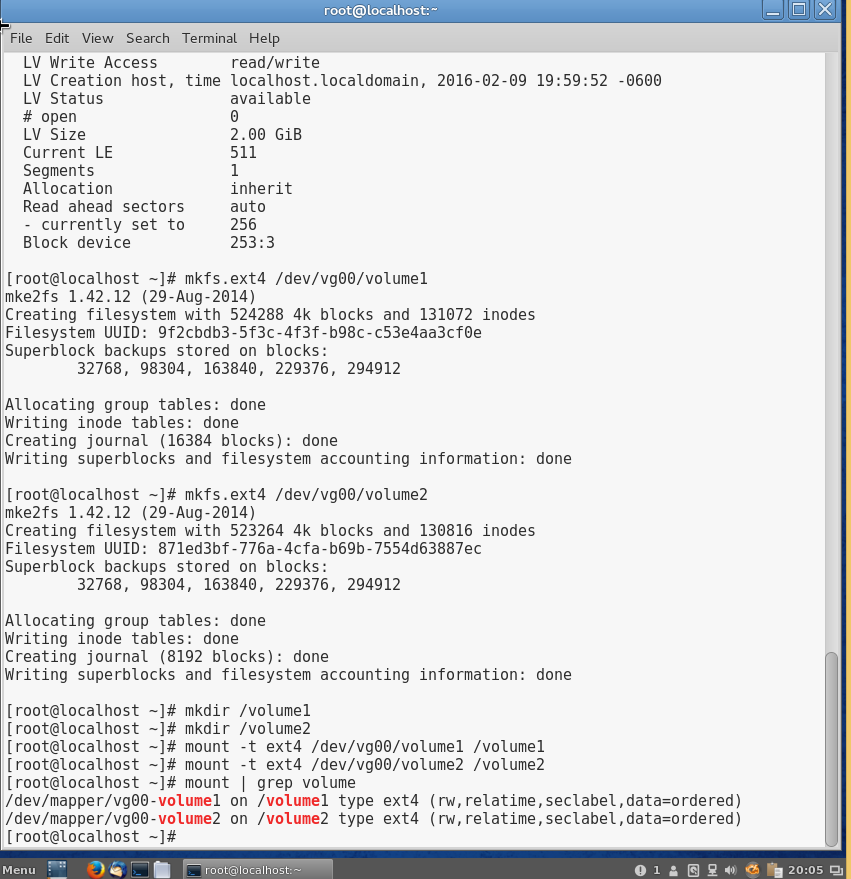
****

****

****

****

1. At the command prompt, type **mkfs.ext4 /dev/vg00/volume1** and press Enter to format the volume1 logical volume using the ext4 filesystem. Next, type **mkfs.ext4 /dev/vg00/volume2** and press Enter to format the volume2 logical volume using the ext4 filesystem.
2. At the command prompt, type **mkdir /volume1** and press Enter to create a mount point for the volume1 logical volume. Next, type **mkdir /volume2** and press Enter to create a mount point for the volume2 logical volume.
3. At the command prompt, type **mount –t ext4 /dev/vg00/volume1 /volume1** and press Enter to mount the volume1 logical volume to the /volume1 directory. Next, type **mount –t ext4 /dev/vg00/volume2 /volume2** and press Enter to mount the volume2 logical volume to the /volume2 directory.
4. Verify the directories are mounted by typing **mount | grep volume** and press Enter. Verify both volumes are mounted.
5. **Provide screenshot(s) of steps 11 through 14.**

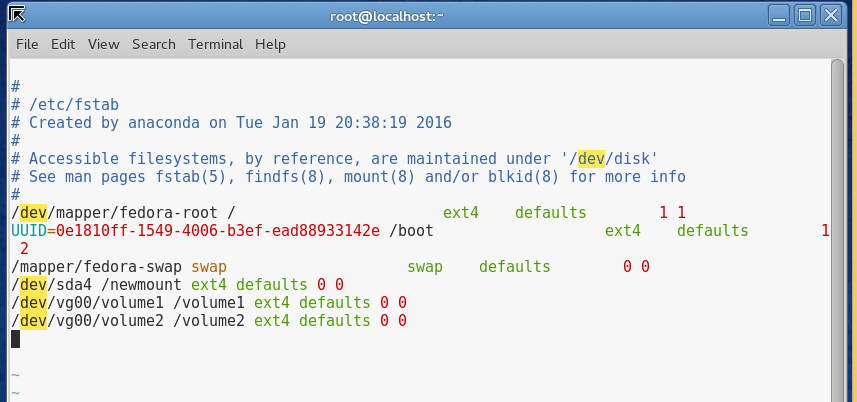
****

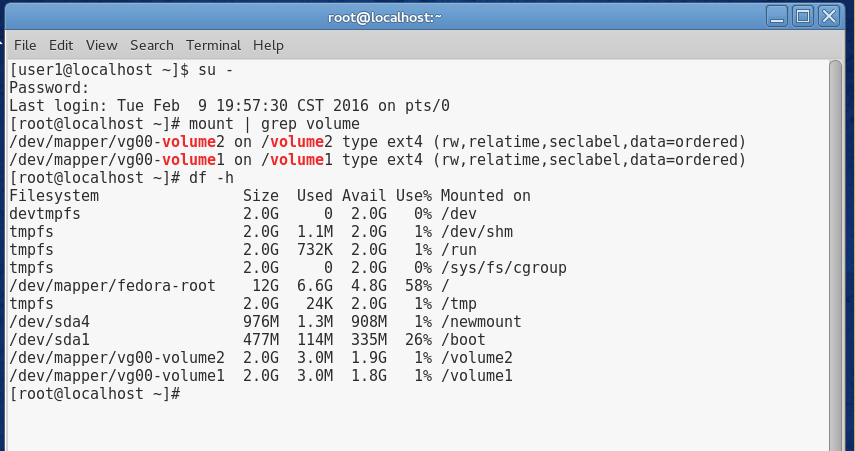
1. At the command prompt, type **vim /etc/fstab** and press Enter. Add the following lines to the bottom of the file, as shown below, to ensure that the volume1 and volume2 logical volumes are mounted at boot time:

/dev/vg00/volume1 /volume1 ext4 defaults 0 0

/dev/vg00/volume2 /volume2 ext4 defaults 0 0

1. Save your changes and quit the vim editor.
2. At the command prompt, type reboot and press Enter. Log back into your system and open a terminal and become the **root** user.
3. Ensure the volumes are mounted by typing **mount | grep volume** and press Enter.
4. At the command prompt, type **df -h** and press Enter to view the available free space of your partitions.
5. **Provide screenshot of step 16 through 19.**

****

****