OPS245 Lab 5

Contents

- 1 LAB PREPARATION
  - 1.1 Purpose / Objectives of Lab 5
- 2 INVESTIGATION 1: MONITORING HARD DISK SPACE
  - 2.1 Part 1: Hard Disk Space Utilities
  - 2.2 Part 2: Using crontab to Automatically Alert System Administrator of Low Hard Disk Space
- 3 INVESTIGATION 2: MANAGING HARD DISK SPACE USING LVM
  - 3.1 Part 1: Managing File System Size with Existing Hard Drive
  - 3.2 Part 2: Adding Additional Virtual Hard Drives
  - 3.3 Part 3: Manually & Automatically Mount Partitions
- 4 INVESTIGATION 3: Scripting
- 5 LAB 5 SIGN-OFF (SHOW INSTRUCTOR)
- 6 Practice For Quizzes, Tests, Midterm & Final Exam

LAB PREPARATION

Purpose / Objectives of Lab 5

The purpose of this lab is to demonstrate how a Linux system administrator can monitor hard disk space availability, and to manage file system size via the Logical Volume Manager (LVM) application. This lab will also demonstrate how to manually mount (i.e. connect) and unmount (disconnect) partitions to file system directories, and demonstrate how to have partitions automatically mounted to directories upon Linux system startup.

Monitoring Disk Space can fix problems **before** they become a crisis (like running low on hard disk space). We will use LVM to easily resize Linux file-systems.

Linux system administrators need to schedule Linux shell scripts and commands (via **crontab**) to automatically run in order to be more productive.

Main Objectives

- Monitoring Disk Space with utilities such as **ssm list**, **df -h**, and **du -ah**.
- Use the **crontab** utility to automatically schedule the execution of a shell script to "flag" low disk space.
- Use **LVM** to **resize partitions via command-line utilities**.
- Create, partition and format **virtual hard disks** to increase the size of file systems.
- Manually connect and disconnect directories (mount points) to existing partitions (**mount**, **umount**).
- Edit the **/etc/fstab** file to automatically mount partitions upon Linux server boot-up, and test the configuration prior to Linux server boot-up.

Solid State Drive

USB key
(for backups)

Lab5 Log Book

LVM Information

vgs (http://man7.org/linux/man-pages/man8/vgs.8.html)
pvs (http://man7.org/linux/man-pages/man8/pvs.8.html)
lvs (http://man7.org/linux/man-pages/man8/lvs.8.html)
vgdisplay (http://man7.org/linux/man-pages/man8/vgdisplay.8.html)
pvdisplay (http://man7.org/linux/man-pages/man8/pvdisplay.8.html)
lvdisplay (http://man7.org/linux/man-pages/man8/lvdisplay.8.html)
ssm (http://manpages.ubuntu.com/manpages/trusty/man8/ssm.8.html)

LVM Management

lvextend (http://man7.org/linux/man-pages/man8/lvextend.8.html)
lvcreate (http://man7.org/linux/man-pages/man8/lvcreate.8.html)
lvreduce (http://man7.org/linux/man-pages/man8/lvreduce.8.html)
pvcreate (http://man7.org/linux/man-pages/man8/pvcreate.8.html)
vgextend (http://man7.org/linux/man-pages/man8/vgextend.8.html)

Miscellaneous

mount (http://man7.org/linux/man-pages/man8/mount.8.html)
umount (http://man7.org/linux/man-pages/man8/umount.8.html)
df (http://man7.org/linux/man-pages/man1/df.1.html)
du (http://man7.org/linux/man-pages/man1/du.1.html)
awk (http://man7.org/linux/man-pages/man1/awk.1p.html)
fdisk (http://tldp.org/HOWTO/Partition/fdisk_partitioning.html)
mkfs (http://www.cyberciti.biz/faq/howto-format-create-linux-filesystem/)
/etc/fstab (http://man7.org/linux/man-pages/man5/fstab.5.html)
Using crontab (http://code.tutsplus.com/tutorials/scheduling-tasks-with-cron-jobs--net-8800)

INVESTIGATION 1: MONITORING HARD DISK SPACE

Part 1: Hard Disk Space Utilities

Another essential duty of a Linux system administrator is to anticipate problems and take preventative measures to avoid computer system problems before they occur. An example would be to periodically monitor hard disk space in order to make adjustments before it impacts on system performance.

Therefore, we are going to learn in this section how to monitor disk space activity to help take corrective action.

```
[root@centos2 msaul]# df -h
Filesystem                      Size  Used Avail Use% Mounted on
/dev/mapper/centos_centos2-root  9.6G  4.0G  5.2G  44% /
devtmpfs                        912M     0  912M   0% /dev
tmpfs                           921M  148K  921M   1% /dev/shm
tmpfs                           921M  8.8M  912M   1% /run
tmpfs                           921M     0  921M   0% /sys/fs/cgroup
/dev/mapper/centos_centos2-home  1.9G   12M  1.8G   1% /home
/dev/sda1                       497M  120M  377M  25% /boot
/dev/mapper/centos_centos2-archive  976M  2.6M  907M   1% /archive
[root@centos2 msaul]#
[root@centos2 msaul]#
[root@centos2 msaul]# du -sh /home/msaul
6.1M    /home/msaul
[root@centos2 msaul]#
```

The **df** and **du** commands are useful tools for Linux system administrators to flag disk space issues and investigate their causes.

Perform the following steps:

1. Launch your **c7host** and **centos2** VMs.
2. Switch to your **centos2** machine.
3. Open a terminal.
4. Issue the command: `df -h`
5. Note the disk space usage for the / and **/home** partitions.
6. If a partition is running out of available space, the Linux System Administrator can reallocate space among partitions or add another disk and grow the file system. The administrator can also investigate the cause of low disk space. Two examples immediately come to mind: excessive use of space from users, and potential penetration from hackers.
7. To investigate excessive disk usage by regular users, you can obtain a total amount of disk usage for that user by issuing the command:
   `du -ha ~<username> | more`
8. If there is a recurring space usage problem with regular users, the Linux system administrator can impose quotas (caps on disk usage). This method is not taught in this course.
9. The methods to monitor potential penetration to a Linux system are too numerous, and are taught in other courses (for example: SEC520). One method of monitoring potential penetration is use the find command (Note that **find** relies on the permissions of the user currently running it. Compare the results of running this command with and without sudo):
   `find -P / -size +100000k`
10. The next section will apply some of these tools we have discussed into a shell script and crontab entry to periodically monitor and contact the system administrator of potential disk space issues (before they become a serious problem).

Part 2: Using crontab to Automatically Alert System Administrator of Low Hard Disk Space

This section focuses on how to automatically run shell scripts without the Linux system administrator being there to issue that shell script. It would be highly unlikely to expect a system administrator to stay up late (eg. 2 a.m.) to manually run a shell script to terminate processes or to re-boot Linux servers. Database files (tables) are used to provide instructions on how frequent shell scripts or commands can be run.

Bash Shell Scripting Tips:

Using awk to Manipulate Text

- Very useful command for report generation, text file repair, or text and floating-point decimal manipulation. The command mimics a C

The **cron** daemon is used to refer to these shell scripts (or other commands or programs) and to run them on a pre-determined basis. The term **cron** comes from the old word **chronograph** meaning a special type of watch (actually a *stop-watch*) to help monitor and schedule routine tasks.

Perform the following steps:

1. Perform this section in your **c7host** machine
2. Change to your **bin** directory.
3. Download the following shell script by issuing the following command:
   `wget`
   `https://raw.githubusercontent.com/OPS245/labs/main/monitor-disk-space.py`
4. Try to understand what this script does (refer to man pages for the **awk** command), and then run the script with elevated permissions.
5. Give execute permissions and run this shell script. This script is supposed to notify the root user by email if there are any potential partition size issues.
6. Issue the follow command: `sudo mail -u root` (if you get an error, install email by issuing the command: `yum install mailx`
   Check to see if there are any mail messages. If there are mail messages, they do not relate to this script execution. Remove all mail messages by typing d immediately followed by a mail message number range (eg. to remove all messages. For example, if there are 5 messages, type **d1-5** and then press **ENTER** and enter **q** to exit the mail application).
7. Edit the **monitor-disk-space.py** script, and set the `ALERT=90` value to `ALERT=10`. Then save your editing session, and re-run this script.
8. Run the **mail** command. Do you have a mail message? Enter the mail message number to view the message. If there is a message, what is the purpose of this message?
9. Exit from the mail command.

program, with braces { } that surround the action to perform based on records from a database file matching either test conditions, regular expressions, etc. Fields appear as numbers with $.

- Examples:

  awk '{print}' data-file.txt
  awk -F";" '{print $5,$3}' data-file.txt
  awk -F"," '$4 >= 10000 {print $1, $2}' salary.txt

Crontab (Chronograph Tables)

- Used to automatically run (as opposed to manually run) scripts, programs, or commands. There are many tables (files), but the main one is: **/etc/cron**. The **crontab** command can be used to *list*, *create*, *modify* or *remove* scheduled jobs in the file.

- Examples:

  crontab -e -u user # create/modify
  crontab -r -u user-name # remove specific user's crontab
  crontab -l -u username #List current schedules

10. The script as it is currently written will send the email to root, but we won't be logged in as root most of the time. Change the ADMIN variable in the script to your own username.
11. Run the script again and make sure the email message gets delivered to your normal user.

In order to automatically run the above-mentioned script periodically, you use the scheduler in Linux called **crontab**. The term crontab stands for **Chronograph Tables**, where a chronograph is the old term for a timepiece (the forerunner of the modern stop-watch). You can run the crontab command to schedule commands and shell script to be run in a number of different ways.

11. Quickly view the tutorial about the **Using crontab (http://code.tutsplus.com/tutorials/scheduling-tasks-with-cron-jobs--net-8800)** file to understand the purpose of this file and how to basically set up a schedule to run a shell script.
12. Issue the following command to setup a crontab entry for root:
    `crontab -e`
13. Enter the following line in order to run at 6:00 on the first day of every month:
    `0 6 1 * * /home/<YOURUSERNAME>/bin/monitor-disk-space.py #Runs first day of each month (6:00 am)`
    Note: Make sure you put your own username in that entry.
14. **Save** the crontab entry.
15. Confirm that the entry was properly saved by issuing the following command:
    `crontab -l`

Answer INVESTIGATION 1 observations / questions in your lab log book.

INVESTIGATION 2: MANAGING HARD DISK SPACE USING LVM

An application called **LVM** is a very useful tool for Linux system administrators to easily manage file systems, in some cases, even when the computer system is running!

LVM (**Logical Volume Management**) is used to manage hard disk drives / partitions for Linux and Unix systems. LVM provides more flexibility than just partitioning hard disks. **Volume Groups** are areas used to define **Physical Volumes** (i.e. hard disks, disk partitions, or other forms of storage devices). **Logical Volumes** are then used to relate directories (mount points) to a specific physical volume or for a "range" or "span" of physical volumes.

LVM allows more flexibility and growth potential for Linux systems (for example, having Logical volumes span multiple hard disks). CentOS uses LVM by default upon installation. Other Linux distributions may provide the capacity to install LVM,

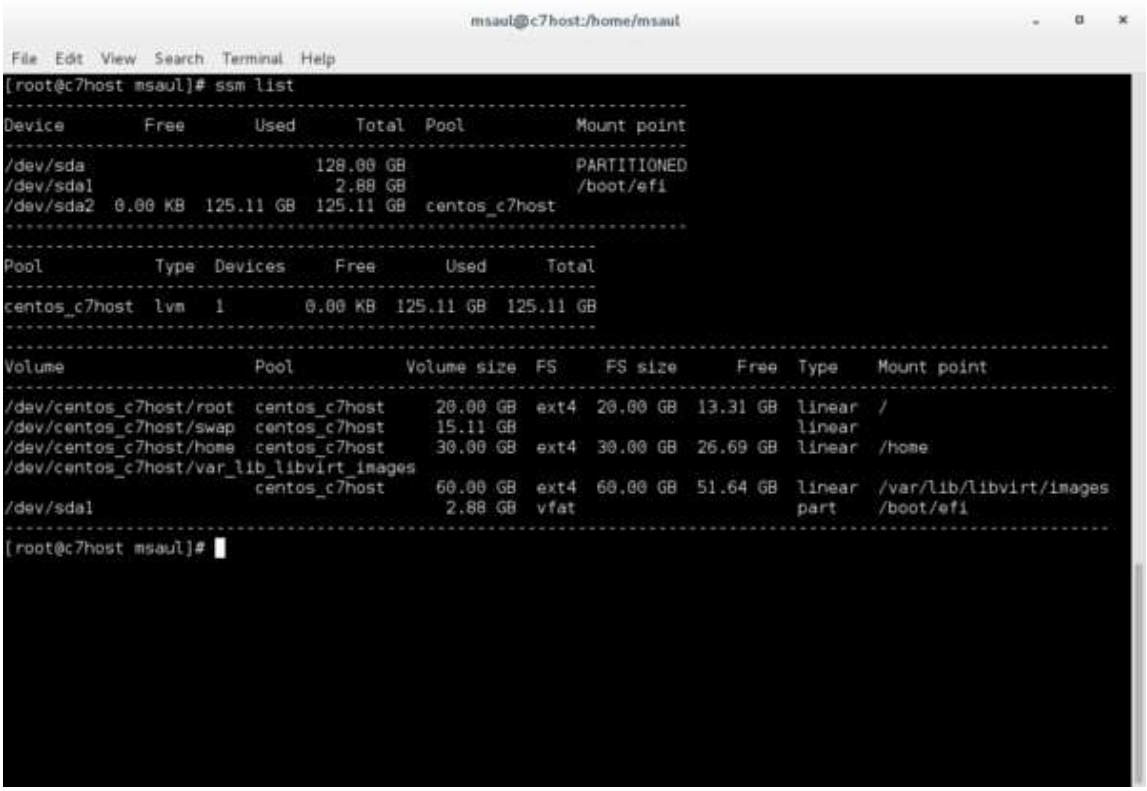
Part 1: Managing File System Size with Existing Hard Drive

We will now use LVM in order to grow and reduce our file system, using extra unallocated space on our existing (default) virtual hard disk for our centos2 VM.

Perform the following steps:

1. Remain in your **centos2** VM for this section.
2. Issue the command: `sudo ls /dev/vd*`

NOTE: If nothing displays, issue the command: **sudo ls /dev/sd*** and use that device pathname **/dev/sda** instead, and notify your instructor when about to run your lab5-check.bash shell script at the end of this lab.



Graphical programs like **system-config-lvm** are **deprecated**, and no longer come bundled with Centos. There are other graphical LVM programs, but are for the KDE desktop environment as opposed to Gnome. Command-line tools such as **ssm** (System Storage Manager), **fdisk**, **mkfs**, **pvcreate**, **lvextend**, and **lvreduce** are sufficient to resize file systems when using LVM.

3. Issue the following command to install the **ssm** command: `sudo yum install system-storage-manager`
4. Issue the command: `sudo ssm list`. Take a few moments to note the volume group, physical volume and logical volume sections of the command output.
5. Compare this output from the ssm command with these other lvm commands: `sudo lvs`, `sudo pvs`, and `sudo vgs`. Which method do you prefer to use?
6. Check to see if there is any remaining space on your existing hard disk. Can you see any?
7. You can create a partition by using the fdisk command. Issue the following command: `sudo fdisk /dev/vda` (or fdisk /dev/sda if there is no /dev/vda).
8. At the **fdisk** prompt, issue the command: `p`. What does this do?
9. Now issue the commands `n` (new partition), `p` (primary partition), `3` (i.e. next available partition number). When prompted for initial block, press **enter** to accept the default beginning block, and type: `+3G` at ending block (create a 3GB partition) and press **enter**.
10. At the fdisk prompt, issue the command `p` to review the partition information.
11. Enter the command `w` to save partition table and exit (read the WARNING message).

You MUST reboot your centos2 VM
You MUST now reboot your centos2 VM before proceeding!

13. You **must restart** your centos2 VM to allow changes to take effect.
14. Verify that you created this partition by issuing the following command: `sudo fdisk -l /dev/vda`
15. Re-issue the **ssm** command. Do you see a new /dev/vda3 partition under Physical Volumes?
16. To add the newly created partition, you need to add it into LVM to be used. Issue the following command to add the partition into LVM:
    `sudo pvcreate /dev/vda3` (or *pvcreate /dev/sda3*) (enter **y** to proceed - ignore warning)

Check your VG name
Run **vgs** to determine your Volume Group name. If it is just **centos** or **cl**, replace **centos_centos2** with **centos** or

cl for the rest of the following commands in this lab.
Do **NOT** try to rename your volume group.

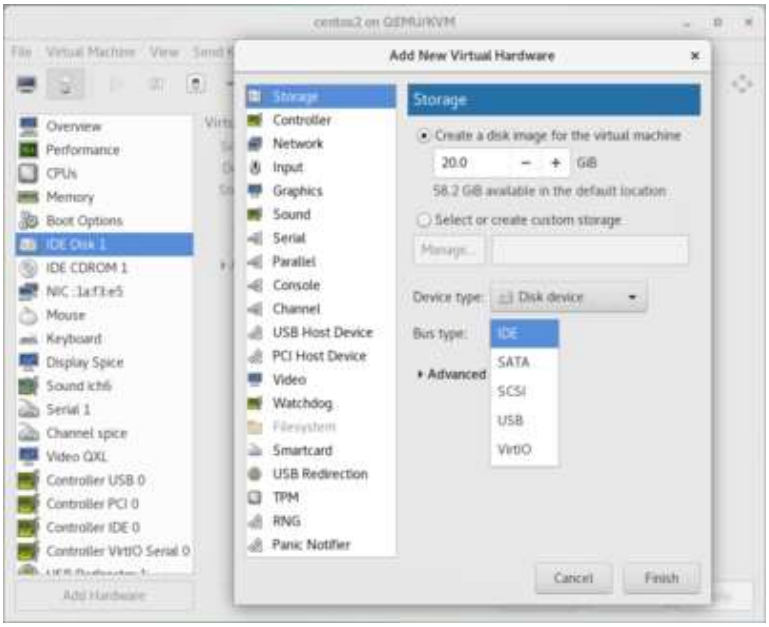
17. Issue the following command to add your new-created and formatted partition called /dev/vda3 to your volume group:
    `sudo vgextend centos_centos2 /dev/vda3`
18. Create a new logical volume by issuing the following command:
    `sudo lvcreate -L 2G -n archive centos_centos2`
19. Format your newly-created partition by issuing the command: `sudo mkfs -t ext4 /dev/centos_centos2/archive`
20. Issue the **ssm list** command to view the new physical volume and logical volume information.

Pay attention to syntax
Note that the prefixed "+" or "-" in lvextend and lvreduce will add or subtract from the current size. Omitting these prefixes will **set** the LV size to what you specified.

21. Reduce the file-size by issuing the command: `sudo lvreduce -r -L -0.5G centos_centos2/archive`
22. Issue the **ssm list** command to verify.
23. Increase the file-size by issuing the command: `sudo lvextend -r -L +1G centos_centos2/archive`
24. Issue the **ssm list** command to verify.

Part 2: Adding Additional Virtual Hard Drives

What if your have noted while monitoring disk space, that you starting to run-out of space on your **home** file-system, although you **do NOT have any available space on your current hard disk?** You could obtain an additional hard-drive. We can **add a new virtual hard-drive** (which will serve as a physical volume to the volume group), and extend the **home** logical volume to make use of the new available space. Creating virtual hard drives is not only inexpensive, but a great way for students to learn now to simulate growing the size of the file system!



You can add virtual hard disks for a VM by changing to the Details section for the VM (as opposed to console), click Add Hardware, fill information in the Add New Virtual Hardware dialog box and clicking Finish. **Notice that I have my original storage in the background so I know what type of disk to select for this second device. They should match.**

Perform the following steps:

1. Remain in your **centos2** VM for this section.
2. Run the following commands and make note of the output of the commands:
   `sudo ls /dev/vd*`,
   `sudo ssm list`, and `sudo df -h`
3. Record the size of the volume group and the amount of free space
4. At the top of your KVM window for **centos2**, click the **view** menu and change view from **Console** to **Details** .
5. At the bottom left-hand corner, click **Add Hardware** and add a new storage device of **2GBs**, make sure the **Bus type** is selected **using the same type as your first drive that's already there**. If your first drive is SATA, IDE, or VirtIO, select that.

6. Click the **VM** menu again, and return to the **console** view to access your centos2 VM display.
7. Issue the command: `sudo ls /dev/vd*`, what has changed?
8. Use **fdisk** (*refer to how to use in Part 1*) to create a new single **primary** partition for **/dev/vdb** that fills the *entire* disk, save partition table (accepting default prompts would work and **type w to write your changes**!), restart your **centos2** VM.
9. Format your new *vdb1* partition with file type: **ext4**
10. Now we'll make the new device a **physical volume**, add it to the **volume group** by issuing the following commands:
    `sudo pvcreate /dev/vdb1` (enter **y** to proceed - ignore warning)
    `sudo vgextend centos_centos2 /dev/vdb1`

    NOTE: If you experience an error message, issue the **sudo ssm list** command, and check the **volume group name** under the **"pool"** section.
    If the volume group name is different than **centos_centos2**, then use that volume group name for all remaining commands that use "centos_centos2"

11. Re-issue the **sudo ssm list** command to see if there is any change.
12. Issue the following command to extend the logical volume for the home file-system: `sudo lvextend -r centos_centos2/home --size +2G`
13. Record the size of the volume group and the amount of free space. What has changed and what caused those changes?
14. Issue the **sudo ssm list** command. Note that your home file-system is now 2GB bigger, and you have not even rebooted your machine since you used fdisk to create a partition!
15. **Reboot** your centos2 VM
16. Record the LVM Management commands in your lab log-book.

Part 3:

```
> su
> mount

proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=499244k,nr_inodes=124811,mode=755)
/dev/sda2 on / type ext4 (rw,relatime,seclabel,data=ordered)
/dev/sda1 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,noquota)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
[root@c7host msaul]#
```

Using the **mount** command with no arguments displays file-systems that are already mounted. The Linux system administrator can use the **mount** and **umount** commands to connect and disconnect different partitions from the file-system to perform maintenance.

Manually & Automatically Mount Partitions

We take for granted that a file-system must be mounted (for example, the root partition) in order for a Linux system to be usable upon system start-up. We need to learn how to do this manually by editing or adding an entry in the file system table (**/etc/fstab**). This file contains entries to mount various file systems automatically upon start-up of the Linux system.

The Linux system administrator also has the ability to manually **mount** (connect) and **un-mount** (disconnect) partitions in order to perform maintenance on the file system (for example, un-mounting the **/home** partition to install software and prevent users from logging in during that process).

Perform the following steps:

1. Perform this part in your **centos2** VM.
2. Issue the following command to create a mount-point (directory to connect /dev/dva3 partition to):
   `sudo mkdir /archive`
3. Issue the following command to mount the partition:
   `sudo mount -t ext4 /dev/centos_centos2/archive /archive`
4. Use the **ls** command to view the contents of the /archive directory. What do you see?
5. Issue the **mount** command (without arguments) to confirm it has been mounted.
6. Unmount the **/archive** directory by issuing the following commmand: `sudo umount /archive`
7. Issue the **mount** command (without arguments) to confirm it has been unmounted.

We will now edit the /etc/fstab file in order to have the /dev/vda3 partition automatically mounted to the /archive directory upon system boot-up

8. View the contents of the file-system table **/etc/fstab** by issuing the following command:
   `cat /etc/fstab`
9. use `sudo vi /etc/fstab` to add an entry to automatically mount the /archive directory upon bootup:
   `/dev/centos_centos2/archive /archive ext4 defaults 1 2`

   Note: do not alter any of the lines that are already in that file.

10. Issue the command `sudo mount -a` to see if the entry in **/etc/fstab** works correctly. If there are any errors you must correct them before rebooting your machine.
11. Reboot the machine and make sure the **/archive** directory is automatically mounted when the machine boots.

Answer the INVESTIGATION 2 observations / questions in your lab log book.

INVESTIGATION 3: Scripting

The script used to monitor disk space earlier in the lab is very useful but has a noticeable flaw: You have to manually change the script to modify the percentage of disk space usage that will trigger a warning. It also will only ever send the email to root (unless you manually change the script).

1. Make a copy of of "monitor-disk-space.py" called **disk-monitor.py** and place it in your user's bin directory.
2. Keep the original preamble comments in the script, but add a line indicating that you modified it (and when you did so). Make sure you indicate the change in script name too.
3. Modify the script so that it will use **argparse** to accept command line input from the user for their preferred values for who to send the email to, and what percentage of use is worth sending an email over.

   You may choose the letters for each of these options, just document your choice in the comments in the script.
   Make sure your script keeps the original values as defaults, in case the user doesn't specify one of them (i.e. if they don't use the option to set who to send the email to, just continue to use root).

4. Test your script with both good and bad data to make sure it works.
5. When you are confident your script works, you are ready to submit the lab.

LAB 5 SIGN-OFF (SHOW INSTRUCTOR)

Time for a new backup!
If you have successfully completed this lab, make a new backup of your virtual machines as well as your host machine.

Perform the Following Steps:

1. Make certain that your **c7host** and **centos2** VMs are running.
2. Switch to your **c7host** machine.
3. Open a shell terminal, change to your user's **bin** directory.
4. Issue the Linux command: `wget https://raw.githubusercontent.com/OPS245/labs/main//lab5-check.bash`
5. Give the **lab5-check.bash** file execute permissions (for the file owner).
6. Run the shell script (with elevated permissions) and if there are any warnings, make fixes and re-run shell script until you receive "congratulations" message.
7. Arrange proof of the following on the screen:

   ✓ **centos2** VM:

   - Output from **sudo ssm list** command.
   - Proof that **/archive** has been mounted

   ✓ **c7host** Machine:

   - Proof of creation of the shell script: **monitor-disk-space.py**
   - Crontab entry for **root** account
   - Run the **lab5-check.bash** script in front of your instructor (must have all `OK` messages)

   ✓ **Lab5** log-book filled out.

8. Upload a screen of the proof from the previous step, along with the file generated by **lab5-check.bash**, your log book, and your disk-monitior.py script to blackboard.

Practice For Quizzes, Tests, Midterm & Final Exam

1. What is a VG? PV? LV?
2. What is the total size of the "main" VG on your system?
3. How do you create an LV?
4. How do resize an LV?
5. How would you add the disk partition **/dev/sdb7** to your volume group "main"?
6. How would you increase the size of the root filesystem by 50 MB?
7. How can you determine if a partition has been mounted onto a system?
8. How can you unmount an existing partition from the file-system?
9. How can you temporarily mount a partition on a file-system?
10. How can you permanently mount a partition on a file-system upon boot-up?
11. What are the separate elements (fields) of the **/etc/fstab** file?
12. Describe the tools that a Linux system administrator have to monitor disk space usage.

- This page was last edited on 8 May 2022, at 23:27.