Golisano College of Computing and Information Sciences
School of Information

NSSA221 Systems Administration I

Lab 4: Partitioning and RAID

Introduction

Managing file systems and partitions on Linux and Windows is a crucial skill every system administrator must have, and while much of the focus on this lab will be Linux, the same concepts can be applied to Windows. Understanding file systems and how to use them is particularly important when working in a heterogeneous environment because different operating systems have different default file systems. For the end user this is mostly transparent, but for system administrators, it is essential to understand how partitions, partition tables and RAIDs work for both troubleshooting and maintaining a healthy system.

Lab Summary

In this lab, you will create a new Rocky virtual machine for storage, this will be your storage server for the rest of the course. This virtual machine will be used to partition drives using both MBR and GPT, format drives and partitions, create software RAIDs, and recover data from a broken (degraded) RAID on Linux. Throughout the lab, you will gain experience in creating partitions; using tools like **gdisk**, **fdisk**, and **mdadm**; and learn how to mount file systems by using the mount command and creating mount points.

Goals

At the end of this lab you will...

- Learn about the differences between GPT and MBR Partition Table Formats.
- Have experience using the **fdisk** and **gdisk** utilities to create partitions.
- Gain experience using the **mdadm** utility to build RAID configurations.
- Have a better understanding of disk storage systems.
- Gain familiarity with building file systems, creating logical volumes, and mounting partitions.

Preparation

- Complete the week 7 readings.
- Know where to find the Rocky Linux documentation for reference.

Activity Summary

Activity 1 – Setting up the Storage Server
Activity 2 – Formatting and Mounting Drives
Activity 3 – Creating a RAID 1 with an MBR Partition Tables
Activity 4 – Creating a RAID 5 with a GPT Partition Table
Activity 5 – Creating Persistent Mounts
Activity 6 – RAID 5 Redundancy
Activity 7 – Logical Volume Management

Activities

Activity 1 – Setting up the Storage Server

For this activity, you will create a new linked/full clone of the Linux virtual machine. This new device will be the storage server used throughout the lab.

Please Note: Any reference to a "server," in this lab, is the Linux virtual machine created for the RAIDs and partitions.

a. Create a linked clone of Rocky Linux. Give it a meaning name to differentiate it from the Linux client. For example, "storage."

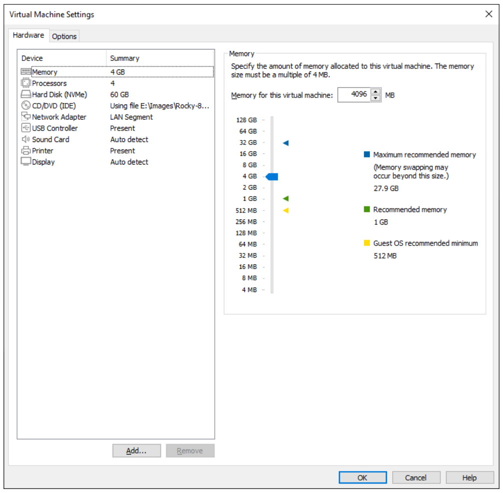
b. Once the clone is created give it a meaningful hostname and assigned it a static network configuration setting, giving it an IPv4 address from the excluded addresses configured in DHCP.

c. Update the system.

d. Power the system off and add eight additional drives.

e. From the virtual machine settings window. Click "Add…"

Figure 1 – Virtual Machine Settings



Virtual Machine Settings

Device	Summary
Memory	4 GB
Processors	4
Hard Disk (NVMe)	60 GB
CD/DVD (IDE)	Using file E:\Images\Rocky-8...
Network Adapter	LAN Segment
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: 4096 MB

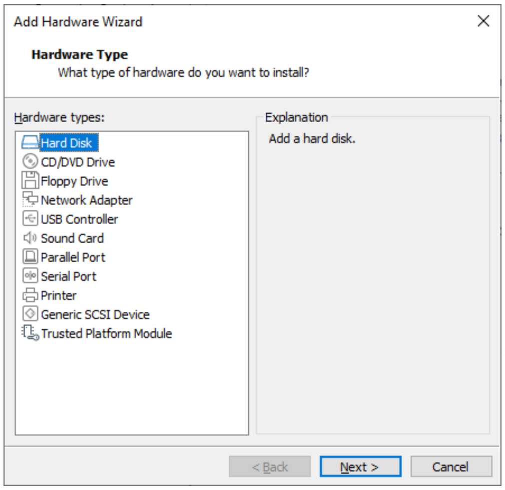
🟦 Maximum recommended memory
(Memory swapping may occur beyond this size.)
27.9 GB

🟩 Recommended memory
1 GB

🟨 Guest OS recommended minimum
512 MB

f. Using the "Add Hardware Wizard," select "*Hard Disk*" Figure 2. Click *Next*.

Figure 2 – Disk Selection



Add Hardware Wizard

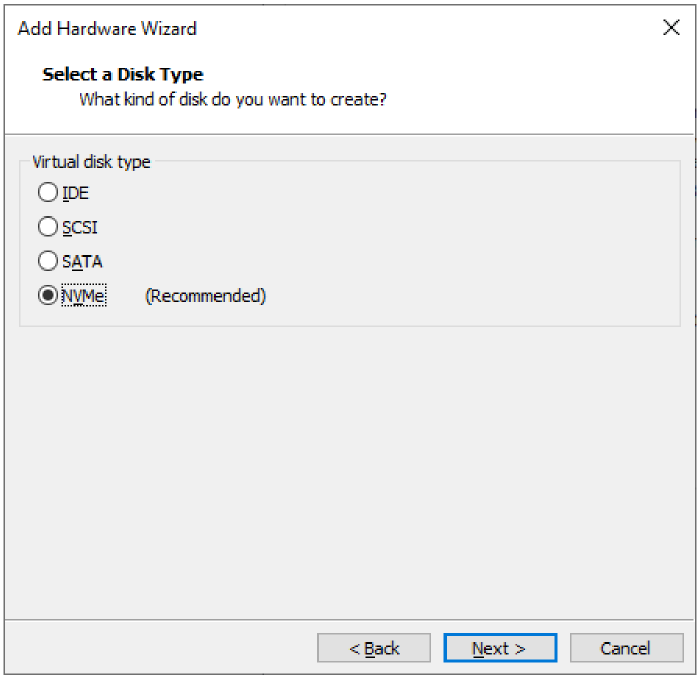
Hardware Type
What type of hardware do you want to install?

Hardware types:
- Hard Disk
- CD/DVD Drive
- Floppy Drive
- Network Adapter
- USB Controller
- Sound Card
- Parallel Port
- Serial Port
- Printer
- Generic SCSI Device
- Trusted Platform Module

Explanation
Add a hard disk.

g. Select the default *NVMe* for the disk type (Figure 3). Click *Next*.

Figure 3 – Disk Type



Add Hardware Wizard

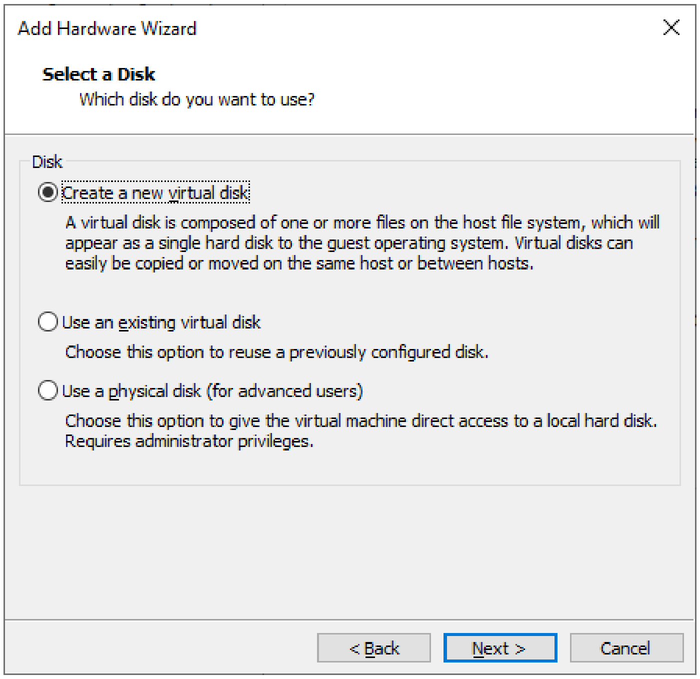
Select a Disk Type
What kind of disk do you want to create?

Virtual disk type

○ IDE
○ SCSI
○ SATA
● NVMe (Recommended)

h. From the "Select a Disk," window, select "Create a new virtual disk," Figure 4. Click *Next*.

Figure 4 – Select a Disk



Add Hardware Wizard

Select a Disk
Which disk do you want to use?

Disk

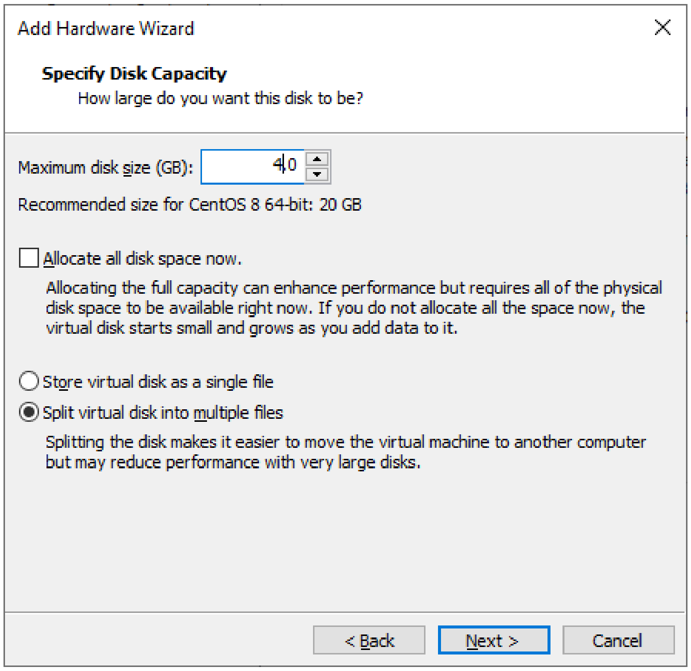
● Create a new virtual disk
A virtual disk is composed of one or more files on the host file system, which will appear as a single hard disk to the guest operating system. Virtual disks can easily be copied or moved on the same host or between hosts.

○ Use an existing virtual disk
Choose this option to reuse a previously configured disk.

○ Use a physical disk (for advanced users)
Choose this option to give the virtual machine direct access to a local hard disk. Requires administrator privileges.

i. On the next window select 4.0 GB for the disk size, Figure 5. Click *Next*.

Figure 5 – Disk Capacity



Add Hardware Wizard

Specify Disk Capacity

How large do you want this disk to be?

Maximum disk size (GB): 40

Recommended size for CentOS 8 64-bit: 20 GB

☐ Allocate all disk space now.

Allocating the full capacity can enhance performance but requires all of the physical disk space to be available right now. If you do not allocate all the space now, the virtual disk starts small and grows as you add data to it.

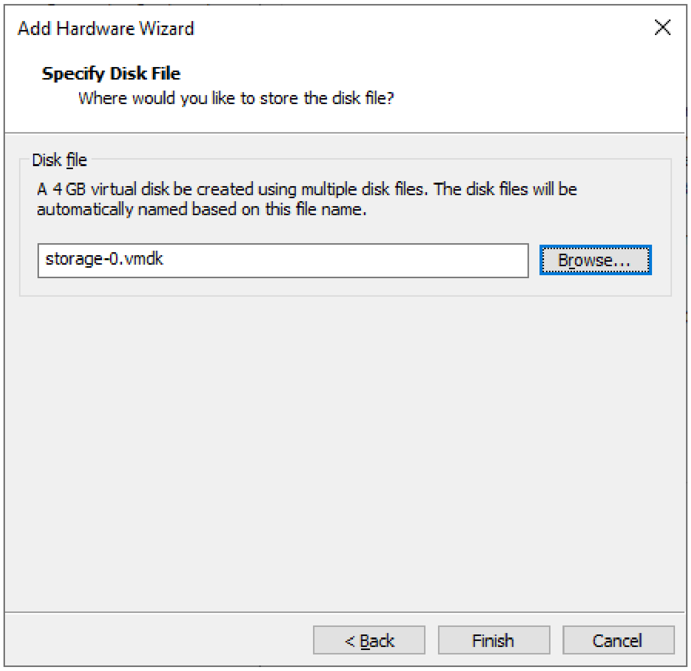
○ Store virtual disk as a single file

● Split virtual disk into multiple files

Splitting the disk makes it easier to move the virtual machine to another computer but may reduce performance with very large disks.

j. The image files for the disk will be created, you may want to note the location of where these files are being stored. It's good practice to store them in the same directory as the linked clone files so everything is in the same place. Click *Finish*.

Figure 6 – Saving Virtual Disk Image Files



Add Hardware Wizard

Specify Disk File

Where would you like to store the disk file?

Disk file

A 4 GB virtual disk be created using multiple disk files. The disk files will be automatically named based on this file name.

storage-0.vmdk

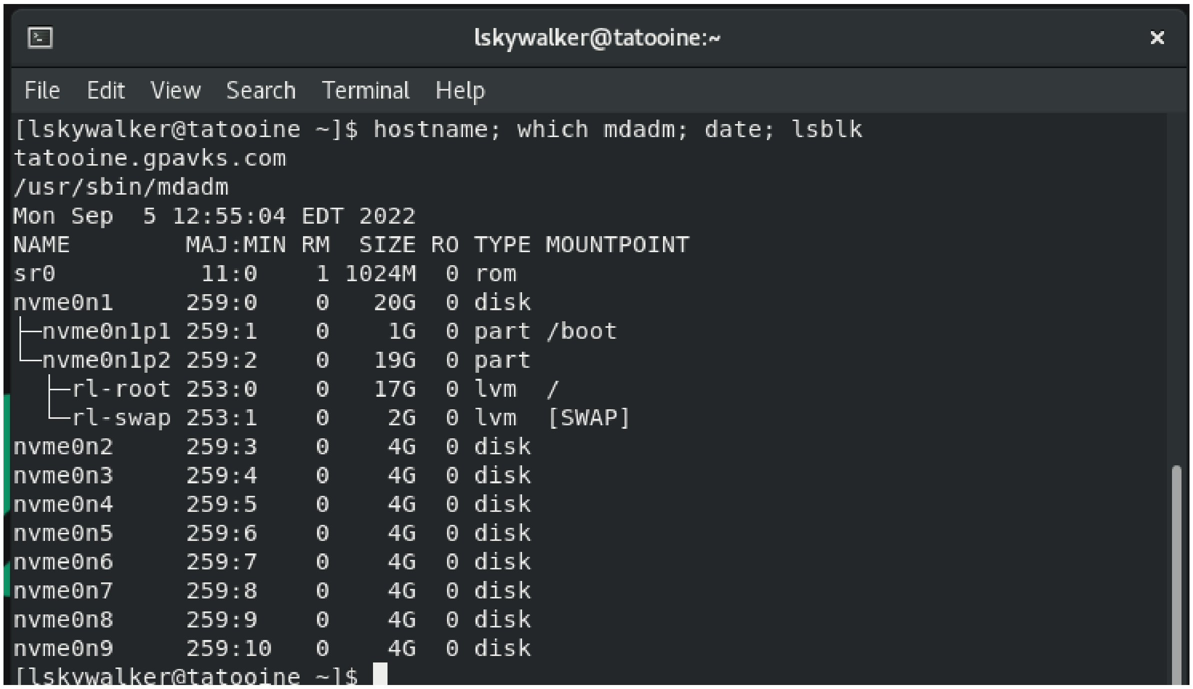
k. Repeat the procedure for the remaining seven drives.

l. Power on the virtual machine.

IMPORTANT NOTICE

a. **For the report**, include a **single** screenshot showing the output of the **hostname, which mdadm**, and the **date** commands. In the same screenshot use the **lsblk** command to show that the eight virtual drives have been added to the virtual machine. See Figure 7 for an example.

Figure 7 – Sample Drive Verification



```
[lskywalker@tatooine ~]$ hostname; which mdadm; date; lsblk
tatooine.gpavks.com
/usr/sbin/mdadm
Mon Sep  5 12:55:04 EDT 2022
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sr0            11:0    1 1024M  0 rom
nvme0n1       259:0    0   20G  0 disk
├─nvme0n1p1 259:1    0    1G  0 part /boot
└─nvme0n1p2 259:2    0   19G  0 part
  ├─rl-root 253:0    0   17G  0 lvm  /
  └─rl-swap 253:1    0    2G  0 lvm  [SWAP]
nvme0n2       259:3    0    4G  0 disk
nvme0n3       259:4    0    4G  0 disk
nvme0n4       259:5    0    4G  0 disk
nvme0n5       259:6    0    4G  0 disk
nvme0n6       259:7    0    4G  0 disk
nvme0n7       259:8    0    4G  0 disk
nvme0n8       259:9    0    4G  0 disk
nvme0n9       259:10   0    4G  0 disk
[lskywalker@tatooine ~]$
```

Based on the output where is the **mdadm** binary located?

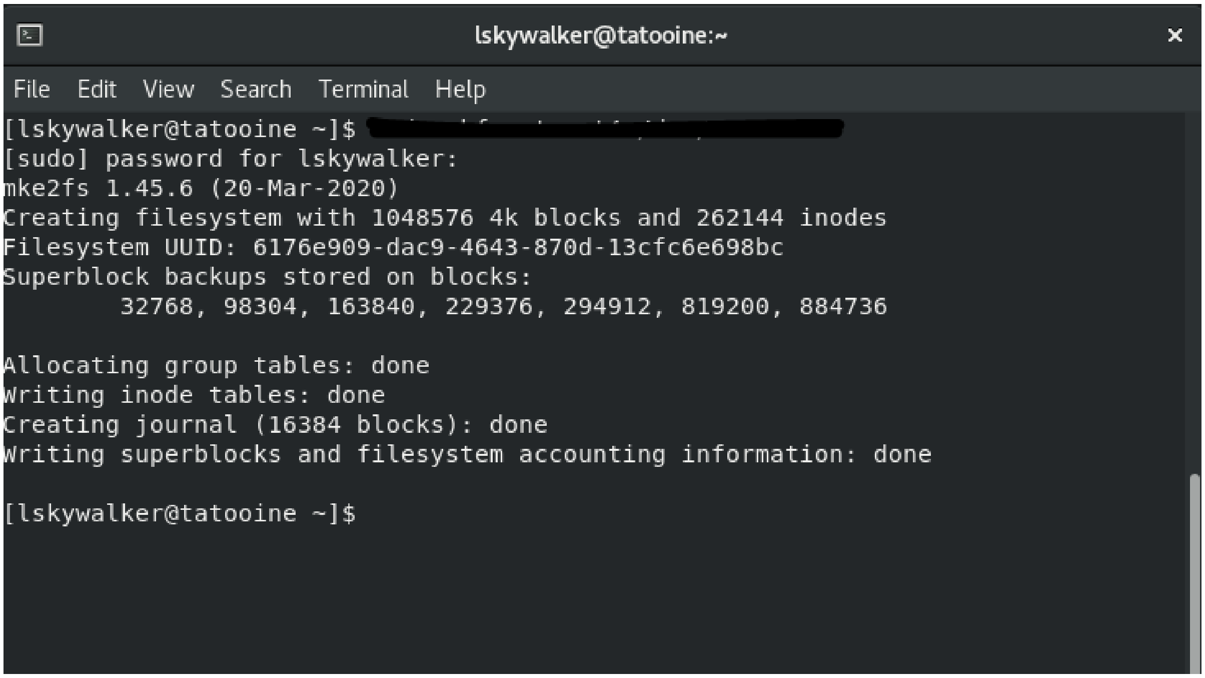
Activity 2 – Formatting and Mounting Drives

In this activity, the first drive will be formatted and mounted. It will be formatted as a raw block device; meaning a partition table will not be used and the resulting filesystem will span the entire drive. Up to this point the previous labs provided detailed instructions and the commands to use. Now that you have some familiarity with using Linux commands and know how to look information up in the man pages you will need to figure out some of the commands for the following activities on your own.

b. Using the **mkfs** command format the first drive that you added to the virtual machine. **Caution: NOT the system drive**, which contains the boot partition, referring to Figure 4 that would be **nvme0n1**. More than likely the first newly added drive will be **nvme0n2**. Using the man page for the **mkfs** command, make the **file type** for that drive **XFS**. Remember that the drives are located in the device (/dev) directory.

c. For reference the output will look similar to Figure 8.

Figure 8 – Sample **`mkfs`** Output



```
[lskywalker@tatooine ~]$ [redacted]
[sudo] password for lskywalker:
mke2fs 1.45.6 (20-Mar-2020)
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: 6176e909-dac9-4643-870d-13cfc6e698bc
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

[lskywalker@tatooine ~]$
```

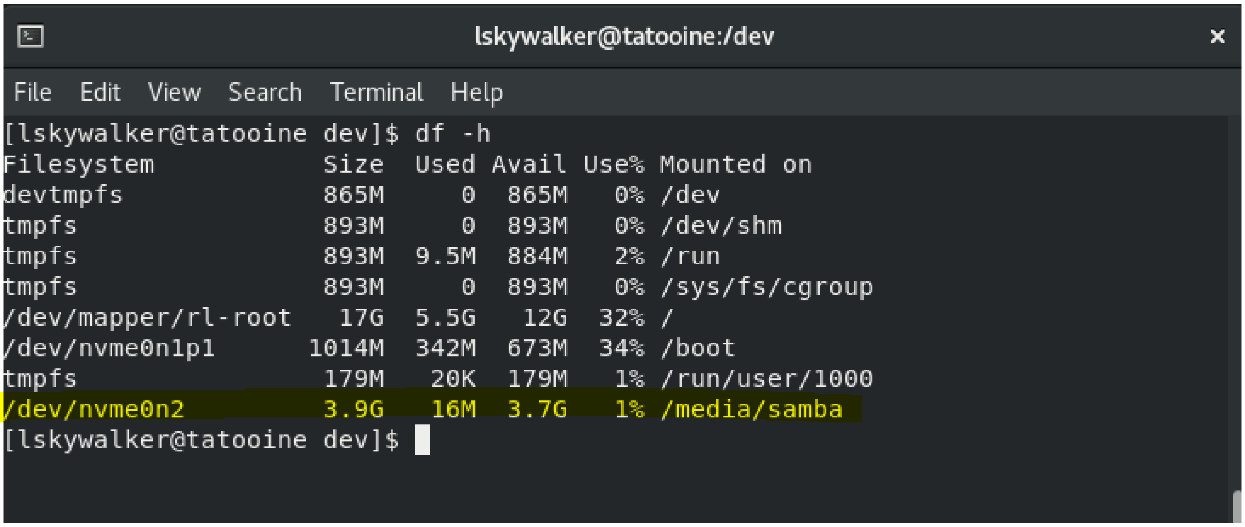
d. Create the directory */media/samba*, this will be used to mount the newly reformatted drive (i.e., the mount point).

e. Use the mount command to mount to the device, for help refer to the man page. The basic syntax of the command is provided below.

```
$ mount <device> <mount point>
```

f. To verify that the device mounted correctly run the **`df -h`** command, your output should be similar to Figure 9, assuming the device is /dev/nvme0n2.

Figure 9 – Sample Output from **`df -h`**

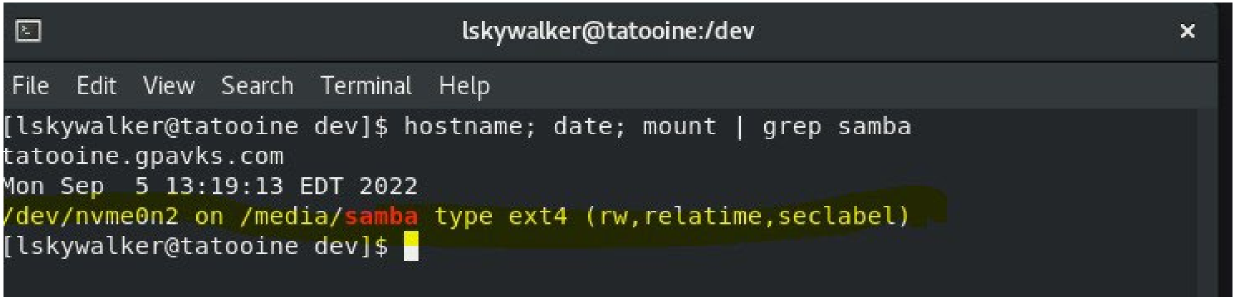


```
[lskywalker@tatooine dev]$ df -h
Filesystem           Size  Used Avail Use% Mounted on
devtmpfs             865M     0  865M   0% /dev
tmpfs                893M     0  893M   0% /dev/shm
tmpfs                893M  9.5M  884M   2% /run
tmpfs                893M     0  893M   0% /sys/fs/cgroup
/dev/mapper/rl-root   17G  5.5G   12G  32% /
/dev/nvme0n1p1      1014M  342M  673M  34% /boot
tmpfs                179M   20K  179M   1% /run/user/1000
/dev/nvme0n2         3.9G   16M  3.7G   1% /media/samba
[lskywalker@tatooine dev]$
```

IMPORTANT NOTICE

For the lab report, include a single screenshot showing your `hostname`, and the `date`. In the same screenshot show the output of the `mount | grep samba` command. See Figure 10 for an example.

Figure 10 – Mount Verification



```
[lskywalker@tatooine dev]$ hostname; date; mount | grep samba
tatooine.gpavks.com
Mon Sep  5 13:19:13 EDT 2022
/dev/nvme0n2 on /media/samba type ext4 (rw,relatime,seclabel)
[lskywalker@tatooine dev]$ 
```

Activity 3 – Creating a RAID 1 with an MBR Partition Table

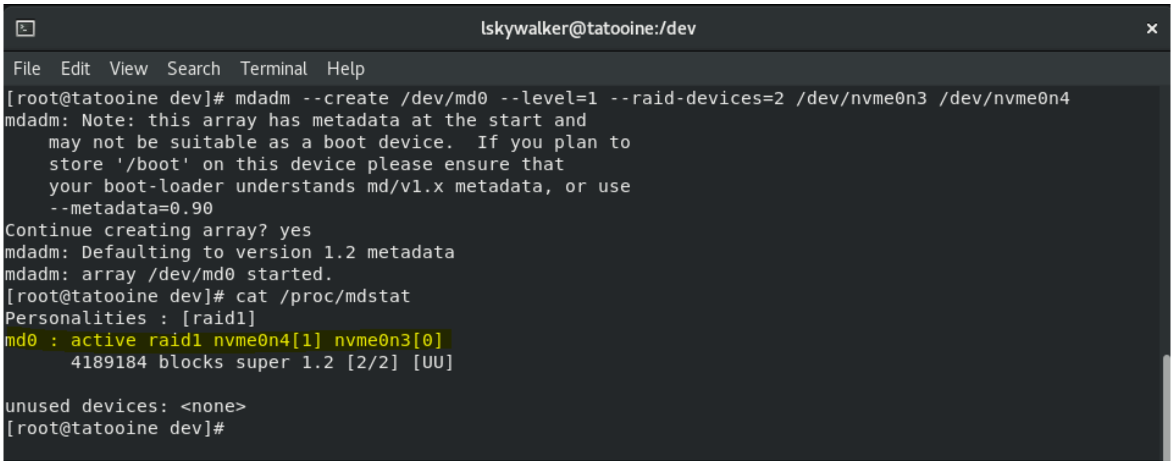
For this activity, you will create a mirrored RAID array, or RAID 1 using the `mdadm` utility. Once the RAID is created it will be partitioned using a Master Boot Record (MBR) partition table. To do this, you will use the `fdisk` utility and create a primary, extended, and logical partition.

a. Open a terminal and enter he `mdadm` command as root. As always, for help using `mdadm` utility refer to the man page. To help you along with the command here are some hints.
   - You are creating a new array from **unused** devices.
   - By convention **/dev/md0** (**m**ultiple **d**isk **0**) is use to identify the first RAID, the second will be /dev/md1, and so on.
   - Additional information you will need is the RAID "level" and the number of devices.
   - You will also need to use the paths for the two unused drives, assuming that the system drive is nvme0n1, and the drive you mounted in Activity 2 was nvme0n2, the next two available drives are nvme0n3, and nvme0n4.

   Please Note: If you are having trouble determining the correct command syntax to use, there is an examples section towards the bottom of the manual page.

b. Run the `cat /proc/mdstat` command to display information about the RAID. The output will be similar to Figure 11.

Figure 11– RAID Creation



```
[root@tatooine dev]# mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/nvme0n3 /dev/nvme0n4
mdadm: Note: this array has metadata at the start and
    may not be suitable as a boot device.  If you plan to
    store '/boot' on this device please ensure that
    your boot-loader understands md/v1.x metadata, or use
    --metadata=0.90
Continue creating array? yes
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
[root@tatooine dev]# cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 nvme0n4[1] nvme0n3[0]
      4189184 blocks super 1.2 [2/2] [UU]

unused devices: <none>
[root@tatooine dev]#
```

c. Next, partition /dev/md0 using the **`fdisk`** utility. Again, hints are provided but you will need to refer to the man pages for further assistance.

e. To view all available commands in **`fdisk`** enter the letter “m”.

f. Find the command to create an MBR partition table. On some Linux distributions it is referred to as a **DOS partition table**.

g. Find the command to add a new partition.

h. The primary partition will be 1GB in size.

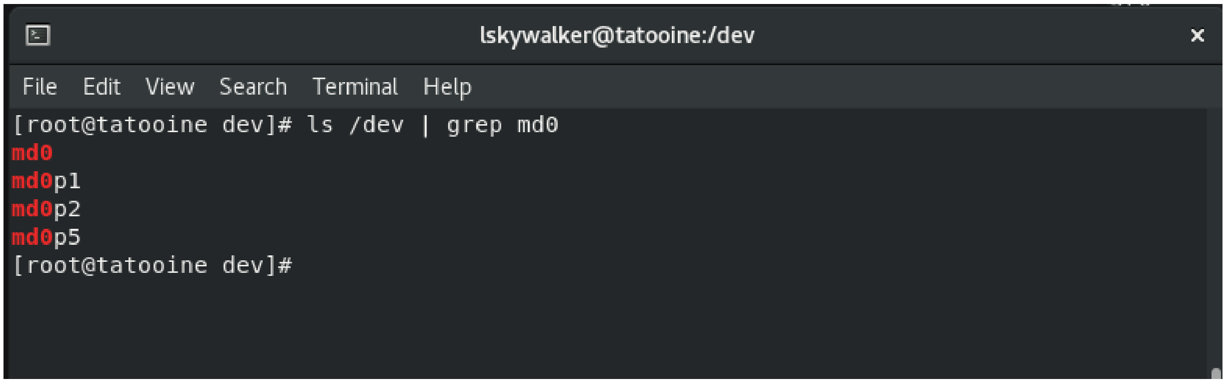
i. Create a second partition, and make it an **extended** partition. This partition will use the remaining available space, if no size is specified **`fdisk`** will use the remaining space by default.

j. Create a third **logical** partition filling the extended partition space.

k. Finally, make sure to write the table to disk and exit the utility.

l. If you did things correctly, you will see output similar to Figure 12 when you run the **`ls /dev | grep md0`** command.

Figure 12 – Verifying Disk Partition Creation



```
[root@tatooine dev]# ls /dev | grep md0
md0
md0p1
md0p2
md0p5
[root@tatooine dev]#
```

m. **Please Note:** md0 is the raw block device and it is where the partition table is written to, if you format it, you will erase the partition table. **DO NOT format md0.** To identify the “extended” partition run the **`fdisk -l`** command and **`grep`** for md0. **DO NOT format the extended partition either.**

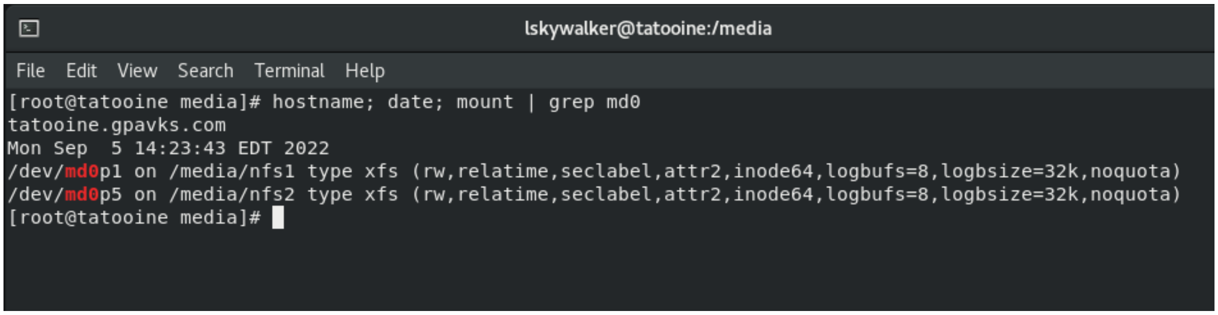
n. Format the primary and logical partitions using a file system of your choice.

o. Create two directories */media/nfs1* and */media/nfs2* and mount the partition to them.

IMPORTANT NOTICE

For the lab report, include a single screenshot showing the hostname, the date, and the output from the **`mount | grep md0`** command. Refer to Figure 13 for an example.

Figure 13 – Sample Mount Verification


lskywalker@tatooine:/media
File Edit View Search Terminal Help

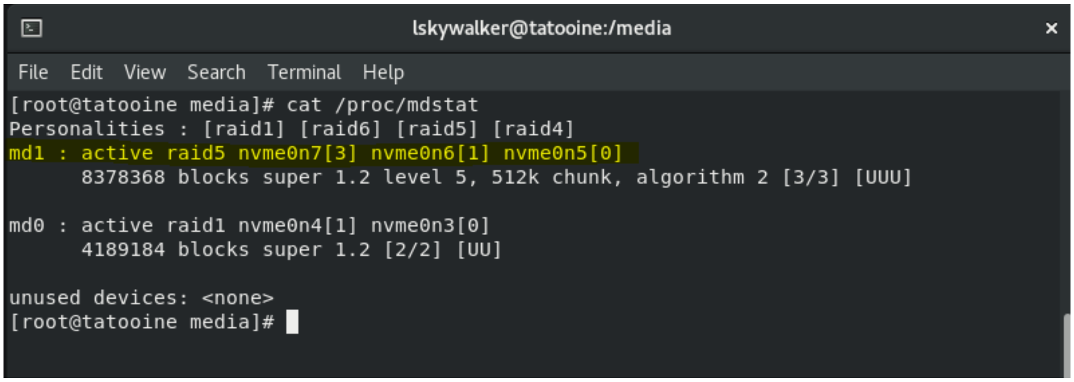
```
[root@tatooine media]# hostname; date; mount | grep md0
tatooine.gpavks.com
Mon Sep  5 14:23:43 EDT 2022
/dev/md0p1 on /media/nfs1 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/md0p5 on /media/nfs2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
[root@tatooine media]#
```

Activity 4 – Creating a RAID 5 with a GPT Partition Table

In this activity, you will create a RAID 5 using the remaining three drives and partition it with a GUID partition table (GPT), using **`gdisk`**. Having experience using the **`mdadm`** and **`fdisk`** utilities, you'll find that **`gdisk`** is nearly identical to **`fdisk`**. The only significant difference is that **`gdisk`** partitions drives using GPT, whereas **`fdisk`** partitions drives using MBR. **NEVER** use **`fdisk`** on a GPT drive or **`gdisk`** on an MBR drive, unless you want to overwrite the partition tables.

a. Open a terminal. Use the mdadm utility to create a RAID 5 using the next three drives, nvme0n5, nmve0n6, and nvme0n7.
b. The output of cat /proc/mdstat may look similar to Figure 14 show that the RAID 5 is active.

Figure 14 – Sample Output


lskywalker@tatooine:/media
×
File Edit View Search Terminal Help

```
[root@tatooine media]# cat /proc/mdstat
Personalities : [raid1] [raid6] [raid5] [raid4]
md1 : active raid5 nvme0n7[3] nvme0n6[1] nvme0n5[0]
      8378368 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

md0 : active raid1 nvme0n4[1] nvme0n3[0]
      4189184 blocks super 1.2 [2/2] [UU]

unused devices: <none>
[root@tatooine media]#
```

c. Open a terminal and run the **`gdisk`** command as root to partition the newly created RAID. Or give **`cfdisk`** a try.

d. Again, use the manual page to determine how to specify the device to partition. Notice that the commands to **`gdisk`** are similar to the **`fdisk`** commands, however, you can enter “m” for the menu options.

e. Create a new GPT partition table with three partitions, each 2GB in size.

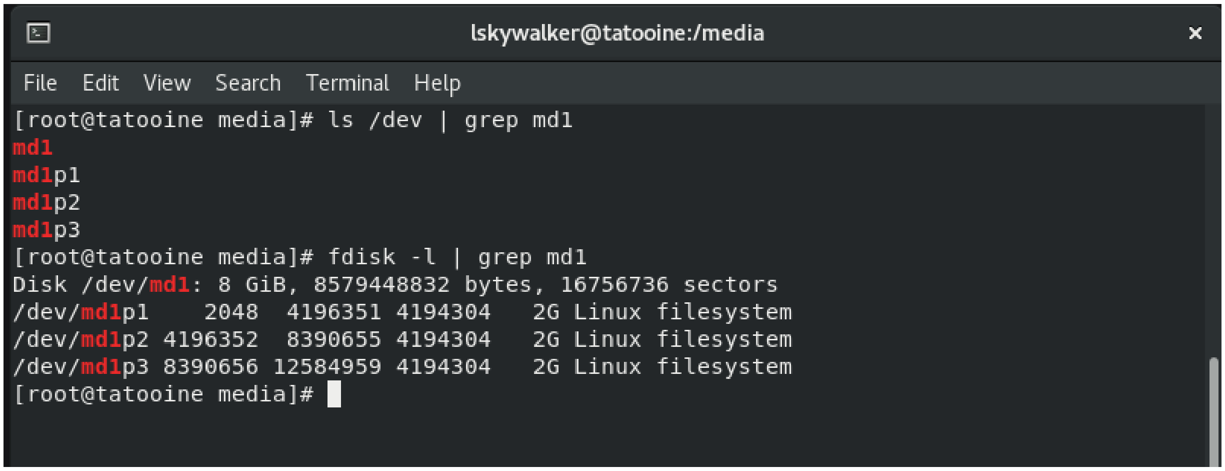
f. When asked to enter the hex code, use the default, 8300.

g. Write the partition table and exit.

h. Format the partitions using a filesystem of your choice. Again, do not format the raw device, md1, because it contains the partition table.

i. Run the command **`ls /dev | grep md1`**. You should see output similar to Figure 15. If not, then repeat steps d through g. Alternatively, you can use the **`fdisk -l | grep md1`** command.

Figure 15 – Sample RAID 5



lskywalker@tatooine:/media
File Edit View Search Terminal Help

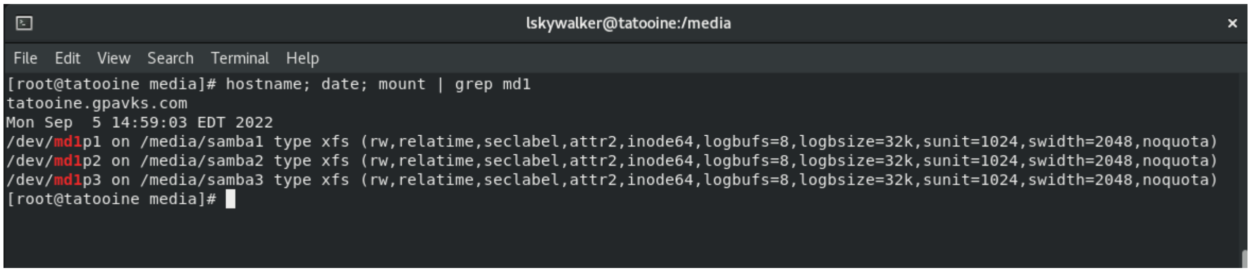
```
[root@tatooine media]# ls /dev | grep md1
md1
md1p1
md1p2
md1p3
[root@tatooine media]# fdisk -l | grep md1
Disk /dev/md1: 8 GiB, 8579448832 bytes, 16756736 sectors
/dev/md1p1    2048  4196351 4194304   2G Linux filesystem
/dev/md1p2 4196352  8390655 4194304   2G Linux filesystem
/dev/md1p3 8390656 12584959 4194304   2G Linux filesystem
[root@tatooine media]#
```

j. Create the directories /media/samba1, /media/samba2 and /media/samba3 mount the partitions.

IMPORTANT NOTICE

For the lab report, include a single screenshot showing the **`hostname`**, the **`date`**, and the output from the **`mount | grep md1`** command. The figure must be a single screen shot properly labeled and included in the lab report. Refer to Figure 13 for an example.

Figure 16 – GPT Mount Verification


lskywalker@tatooine:/media
File Edit View Search Terminal Help

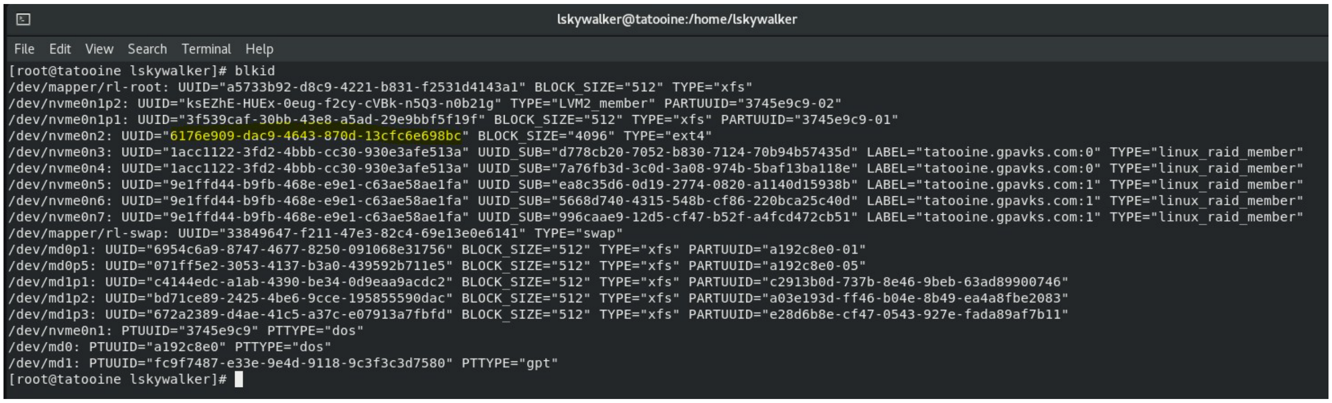
```
[root@tatooine media]# hostname; date; mount | grep md1
tatooine.gpavks.com
Mon Sep  5 14:59:03 EDT 2022
/dev/md1p1 on /media/samba1 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
/dev/md1p2 on /media/samba2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
/dev/md1p3 on /media/samba3 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
[root@tatooine media]# 
```

Activity 5 – Creating Persistent Mounts

Currently, the file system mounts are not persistent, and if the machine is rebooted, they won't be mounted automatically. For this activity, you will make the mounts persistent after a reboot by editing the */etc/fstab* file. While it is good to know how to mount a drive manually, it is certainly not something you would expect an end-user to know how to do. To mount on boot, you will need to place an entry in the */etc/fstab* file. The */etc/fstab* file contains six fields you can learn out about in the manual page.

There are several ways to mount drives and file systems. In this lab, the UUID will be used to mount the file system. Sometimes the path to the device node in /dev can change when the system is rebooted; this is especially true for RAIDs. The UUID is often the best option to prevent mount failures because the UUID does not change. To find the UUID, enter the **`blkid`** command as root (Figure 17). You might even want to redirect it to a file so you can copy/paste the UUIDs into */etc/fstab*. Optionally, you can open another terminal and use some of the commands used in the other activities .to copy and paste the UUIDs.

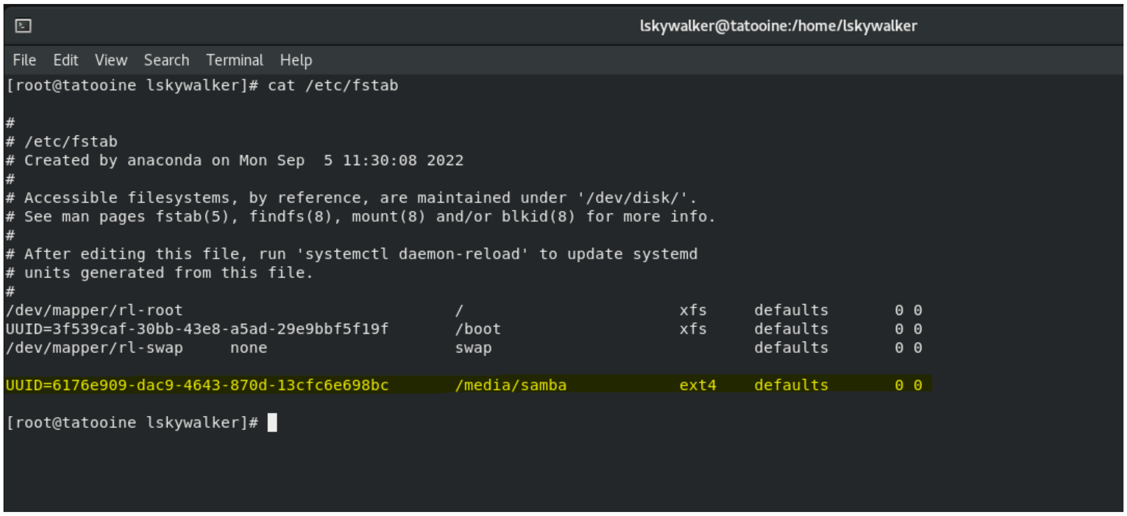
Figure 17 – UUID location


lskywalker@tatooine:/home/lskywalker
File Edit View Search Terminal Help

```
[root@tatooine lskywalker]# blkid
/dev/mapper/rl-root: UUID="a5733b92-d8c9-4221-b831-f2531d4143a1" BLOCK_SIZE="512" TYPE="xfs"
/dev/nvme0n1p2: UUID="ksEZhE-HUEx-0eug-f2cy-cVBk-n5Q3-n0b21g" TYPE="LVM2_member" PARTUUID="3745e9c9-02"
/dev/nvme0n1p1: UUID="3f539caf-30bb-43e8-a5ad-29e9bbf5f19f" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="3745e9c9-01"
/dev/nvme0n2: UUID="6176e909-dac9-4643-870d-13cfc6e698bc" BLOCK_SIZE="4096" TYPE="ext4"
/dev/nvme0n3: UUID="1acc1122-3fd2-4bbb-cc30-930e3afe513a" UUID_SUB="d778cb20-7052-b830-7124-70b94b57435d" LABEL="tatooine.gpavks.com:0" TYPE="linux_raid_member"
/dev/nvme0n4: UUID="1acc1122-3fd2-4bbb-cc30-930e3afe513a" UUID_SUB="7a76fb3d-3c0d-3a08-974b-5baf13ba118e" LABEL="tatooine.gpavks.com:0" TYPE="linux_raid_member"
/dev/nvme0n5: UUID="9e1ffd44-b9fb-468e-e9e1-c63ae58ae1fa" UUID_SUB="ea8c35d6-0d19-2774-0820-a1140d15938b" LABEL="tatooine.gpavks.com:1" TYPE="linux_raid_member"
/dev/nvme0n6: UUID="9e1ffd44-b9fb-468e-e9e1-c63ae58ae1fa" UUID_SUB="5668d740-4315-548b-cf86-220bca25c40d" LABEL="tatooine.gpavks.com:1" TYPE="linux_raid_member"
/dev/nvme0n7: UUID="9e1ffd44-b9fb-468e-e9e1-c63ae58ae1fa" UUID_SUB="996caae9-12d5-cf47-b52f-a4fcd472cb51" LABEL="tatooine.gpavks.com:1" TYPE="linux_raid_member"
/dev/mapper/rl-swap: UUID="33849647-f211-47e3-82c4-69e13e0e6141" TYPE="swap"
/dev/md0p1: UUID="6954c6a9-8747-4677-8250-091068e31756" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="a192c8e0-01"
/dev/md0p5: UUID="071ff5e2-3053-4137-b3a0-439592b711e5" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="a192c8e0-05"
/dev/md1p1: UUID="c4144edc-a1ab-4390-be34-0d9eaa9acdc2" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="c2913b0d-737b-8e46-9beb-63ad89900746"
/dev/md1p2: UUID="bd71ce89-2425-4be6-9cce-195855590dac" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="a03e193d-ff46-b04e-8b49-ea4a8fbe2083"
/dev/md1p3: UUID="672a2389-d4ae-41c5-a37c-e07913a7fbfd" BLOCK_SIZE="512" TYPE="xfs" PARTUUID="e28d6b8e-cf47-0543-927e-fada89af7b11"
/dev/nvme0n1: PTUUID="3745e9c9" PTTYPE="dos"
/dev/md0: PTUUID="a192c8e0" PTTYPE="dos"
/dev/md1: PTUUID="fc9f7487-e33e-9e4d-9118-9c3f3c3d7580" PTTYPE="gpt"
[root@tatooine lskywalker]# 
```

a. Use the manual page for information on the /etc/fstab file and the required syntax. Figure 18 provides an example entry for mounting to /media/samba.

Figure 18 – Sample Entry


lskywalker@tatoine:/home/lskywalker
File Edit View Search Terminal Help

```
[root@tatooine lskywalker]# cat /etc/fstab

#
# /etc/fstab
# Created by anaconda on Mon Sep  5 11:30:08 2022
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rl-root                             /                       xfs     defaults       0 0
UUID=3f539caf-30bb-43e8-a5ad-29e9bbf5f19f       /boot                   xfs     defaults       0 0
/dev/mapper/rl-swap     none                    swap                            defaults       0 0

UUID=6176e909-dac9-4643-870d-13cfc6e698bc       /media/samba            ext4    defaults       0 0

[root@tatooine lskywalker]# 
```

b. If your partition nodes in /dev are the same as the partition nodes in the lab instructions, you will add entries for the partitions created Activities 2 through 4. Using the `blkid` command use the information provided to create a table similar to the one below, recording the UUIDs, mount points and file types. This will help you when adding the entries to /etc/fstab.

Device	UUID	Mount Point	File Type
/dev/nvme0n2	6176e909-dac9-4643-870d-13cfc6e698bc	/media/samba	ext4
/dev/md0p1		/media/nfs1	xfs
/dev/md0p5		/media/nfs2	xfs
/dev/md1p1		/media/samba1	xfs
/dev/md1p2		/media/samba2	xfs
/dev/md1p3		/media/samba3	xfs

c. To assist you, the **UUID** is what you will place in the first field (check the syntax, quotes are not needed), followed by the **mount point** of the device to be mounted, then the **file system type**, followed by the **default** file system options, next a zero because we are not going to dump the file system, and another zero so the system does not run a file system check. Again, **read the manual pages** to understand the purpose of each field.

d. **Syntax is incredibly important.** If you enter incorrect information or create a typo odds are the system will not boot. In which case you will need to going into single use mode to access the /etc/fstab file and check your syntax. You might want to comment out the entries and reboot.

Pro Tip – After you enter the first entry reboot the system to see if it will boot. If it doesn't check the syntax. You might even want to reboot after each entry so you know that the last entry is where the problem is.

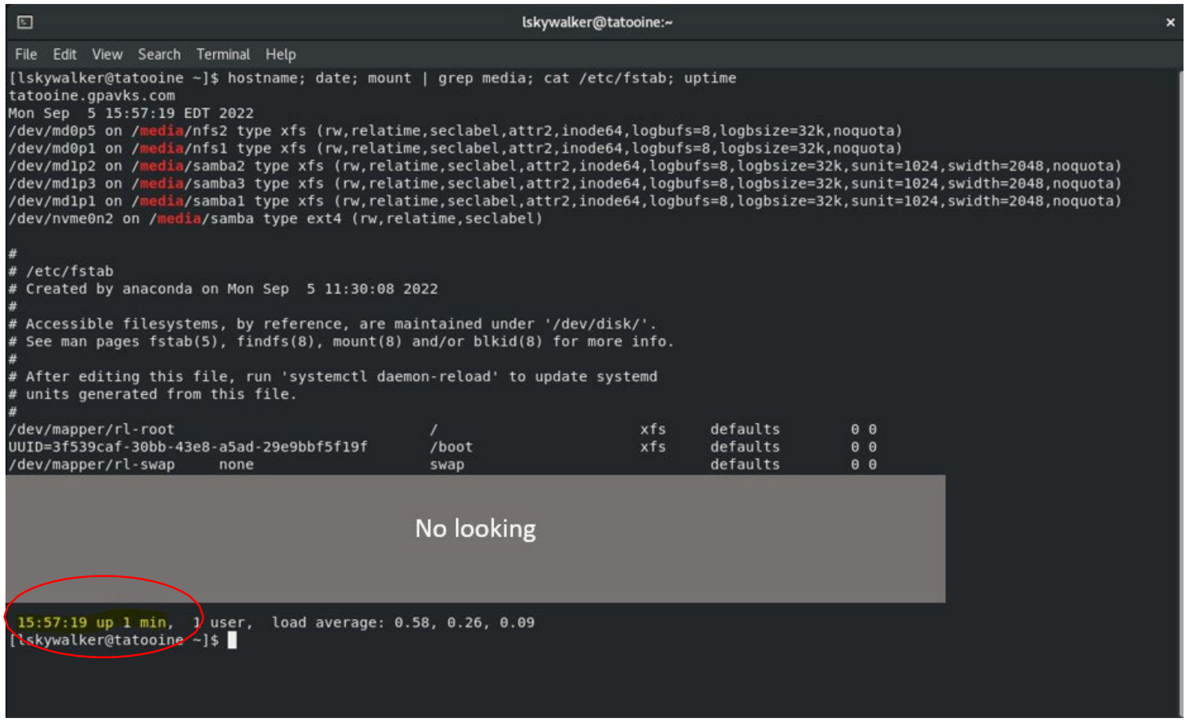
RIT | Golisano College of Computing and Information Sciences
School of Information

IMPORTANT NOTICE

For the report, include a **single** screenshot showing the output from the, **`hostname`**, and **`date`** commands. In the same screenshot enter the following commands to show the output, **`mount | grep media`**, **`cat /etc/fstab`**, and **`uptime`**. The single screen shot must be in the lab report. Refer to Figure 15 for an example.

Important – The uptime in the screenshot must show 5 minutes or less.

Figure 19 – Sample Output for Automount Verification.



```
[lskywalker@tatooine ~]$ hostname; date; mount | grep media; cat /etc/fstab; uptime
tatooine.gpavks.com
Mon Sep  5 15:57:19 EDT 2022
/dev/md0p5 on /media/nfs2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/md0p1 on /media/nfs1 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/md1p2 on /media/samba2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
/dev/md1p3 on /media/samba3 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
/dev/md1p1 on /media/samba1 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,sunit=1024,swidth=2048,noquota)
/dev/nvme0n2 on /media/samba type ext4 (rw,relatime,seclabel)

#
# /etc/fstab
# Created by anaconda on Mon Sep  5 11:30:08 2022
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
/dev/mapper/rl-root                            /                       xfs     defaults        0 0
UUID=3f539caf-30bb-43e8-a5ad-29e9bbf5f19f      /boot                   xfs     defaults        0 0
/dev/mapper/rl-swap     none                   swap                            defaults        0 0

                              No looking

 15:57:19 up 1 min,  1 user,  load average: 0.58, 0.26, 0.09
[lskywalker@tatooine ~]$
```

Activity 6 – RAID 5 Redundancy

Redundancy is one of the benefits of using a RAID 1 or RAID 5 array, meaning that if one of the drives fails, the data on the RAID will be unaffected. In this activity, you will deliberately cause one of the disks in the RAID to fail, verify that no data has been lost, replace the drive and rebuild the array.

a. Create a directory in one of the partitions associated with the RAID 5, if you have been following the directions then it will be one of the partitions mounted to /media/samba1, 2, OR 3. If you are not sure where the RAID is mounted use the **`lsblk`** command to assist you. Add files to this directory, in myCourses in the "*Lab Materials*" section you will find some jpeg files you can use.

b. To simulate a drive failure, use the **`mdadm`** utility and enter the following command as root.

```
# mdadm /dev/md1 -f /dev/nvme0n5
```

Please Note: The command assumes that RAID 5 is associated with **md1** and that **nvme0n5** is one of the drives in the array. To verify the drive associated with your array use the `lsblk` command.

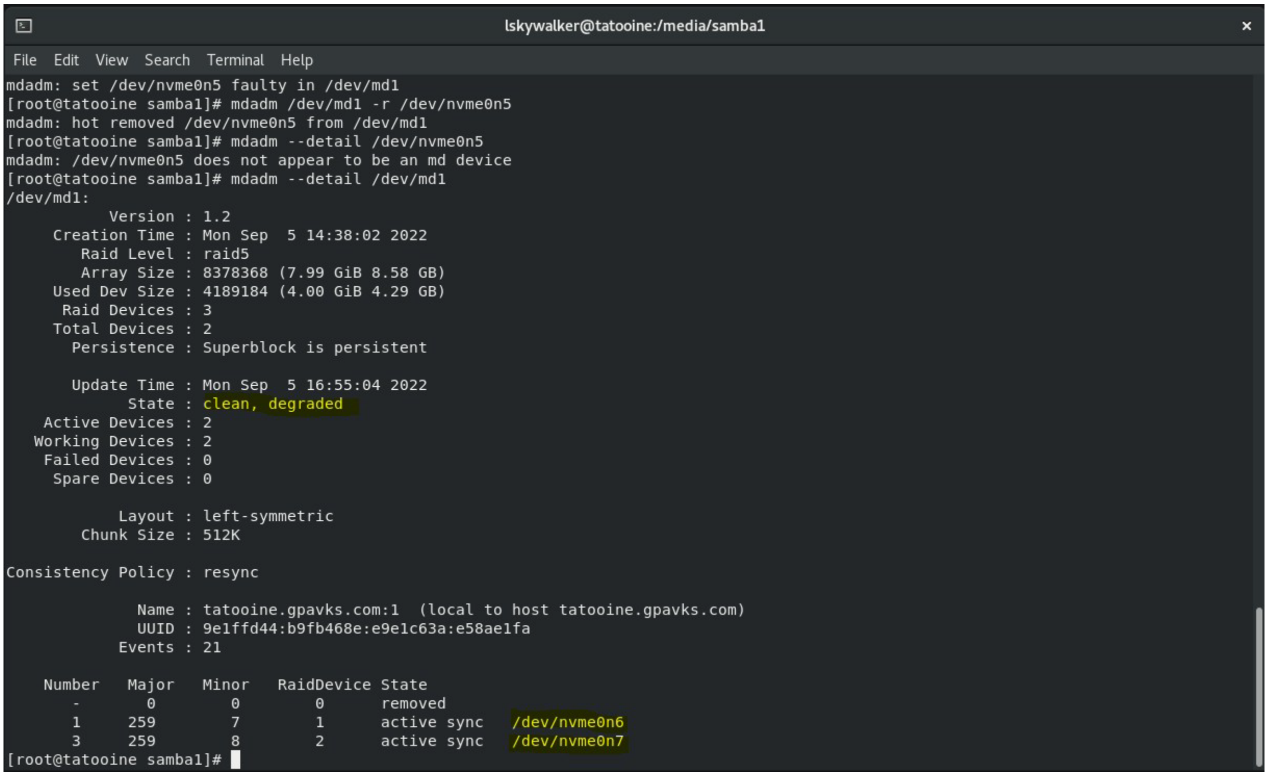
c. Now that the drive has been marked as faulty, remove it using the `mdadm` utility and the following command.

```
# mdadm /dev/md1 -r /dev/nvme0n5
```

d. To show that the array is in degraded mode, enter the following command. The output will look similar to Figure 20. Again, the command assumes that RAID 5 is associated with **md1**. Referring to Figure 20, we can see that the drive is **degraded**, and also that /dev/nvme0n6 and /dev/nvme0n7 are still part of the RAID, but that **/dev/nvme0n5 is not**.

```
# mdadm –detail /dev/md1
```

Figure 20 – Degraded RAID 5



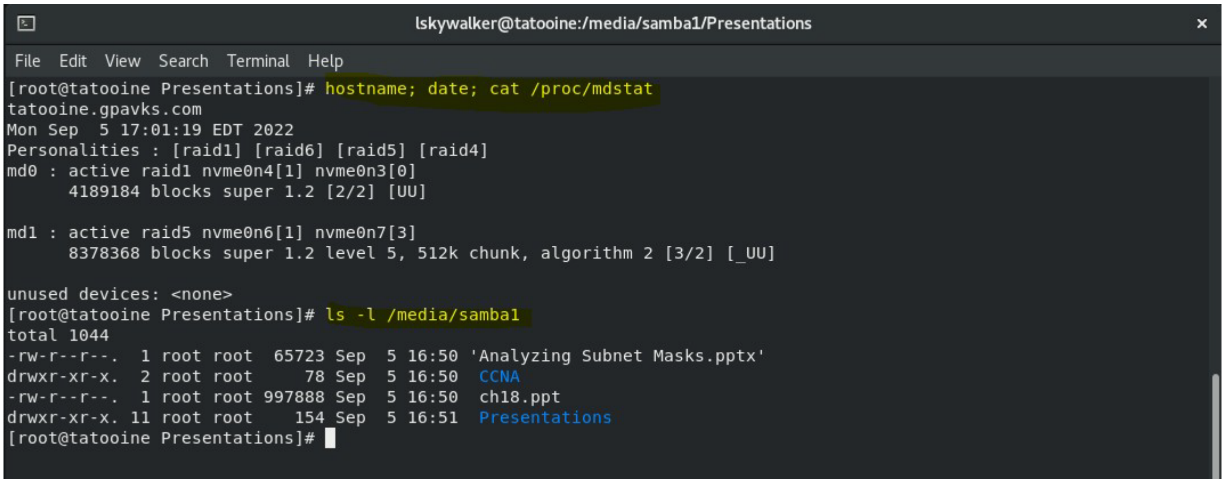
```
mdadm: set /dev/nvme0n5 faulty in /dev/md1
[root@tatooine samba1]# mdadm /dev/md1 -r /dev/nvme0n5
mdadm: hot removed /dev/nvme0n5 from /dev/md1
[root@tatooine samba1]# mdadm --detail /dev/nvme0n5
mdadm: /dev/nvme0n5 does not appear to be an md device
[root@tatooine samba1]# mdadm --detail /dev/md1
/dev/md1:
           Version : 1.2
     Creation Time : Mon Sep  5 14:38:02 2022
        Raid Level : raid5
        Array Size : 8378368 (7.99 GiB 8.58 GB)
     Used Dev Size : 4189184 (4.00 GiB 4.29 GB)
      Raid Devices : 3
     Total Devices : 2
       Persistence : Superblock is persistent

       Update Time : Mon Sep  5 16:55:04 2022
             State : clean, degraded
    Active Devices : 2
   Working Devices : 2
    Failed Devices : 0
     Spare Devices : 0

            Layout : left-symmetric
        Chunk Size : 512K

Consistency Policy : resync

              Name : tatooine.gpavks.com:1  (local to host tatooine.gpavks.com)
              UUID : 9e1ffd44:b9fb468e:e9e1c63a:e58ae1fa
            Events : 21

    Number   Major   Minor   RaidDevice State
       -       0        0        0      removed
       1     259        7        1      active sync   /dev/nvme0n6
       3     259        8        2      active sync   /dev/nvme0n7
[root@tatooine samba1]# 
```

e. Now that we have confirmed that the array is degraded navigate the directory where you placed data and observe that it has not be compromised and is still usable.

IMPORTANT NOTICE

f. For the report, include a **single** screenshot showing the output from the, `hostname`, and `date` commands. In the same screenshot enter the following commands to show the output from the, `cat /proc/mdstat`, and the `ls -l` command where your data is located on the mounted drive. The single screen shot must be in the lab report. Refer to Figure 21 for an example.

Figure 21 – Redundancy Verification


lskywalker@tatooine:/media/samba1/Presentations
File Edit View Search Terminal Help

```
[root@tatooine Presentations]# hostname; date; cat /proc/mdstat
tatooine.gpavks.com
Mon Sep  5 17:01:19 EDT 2022
Personalities : [raid1] [raid6] [raid5] [raid4]
md0 : active raid1 nvme0n4[1] nvme0n3[0]
      4189184 blocks super 1.2 [2/2] [UU]

md1 : active raid5 nvme0n6[1] nvme0n7[3]
      8378368 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/2] [_UU]

unused devices: <none>
[root@tatooine Presentations]# ls -l /media/samba1
total 1044
-rw-r--r--.  1 root root  65723 Sep  5 16:50 'Analyzing Subnet Masks.pptx'
drwxr-xr-x.  2 root root     78 Sep  5 16:50  CCNA
-rw-r--r--.  1 root root 997888 Sep  5 16:50  ch18.ppt
drwxr-xr-x. 11 root root    154 Sep  5 16:51  Presentations
[root@tatooine Presentations]# 
```

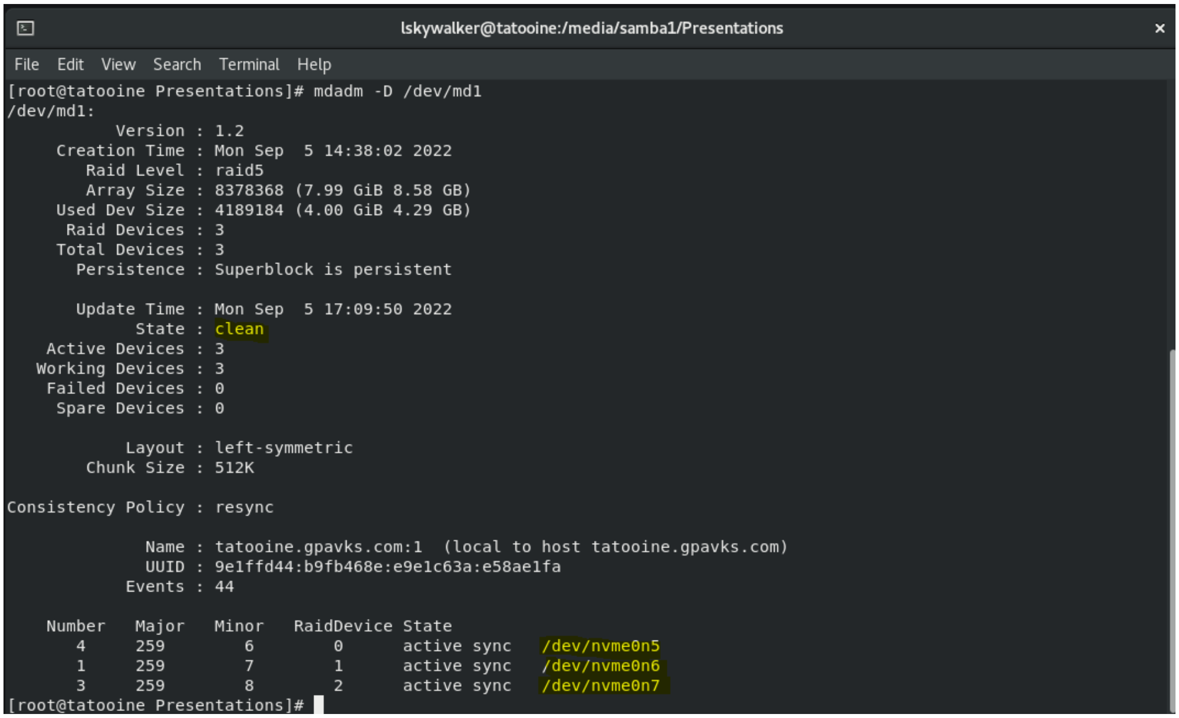
g. To rebuild the array, and add the drive back type the following command. Again, the command assumes that the array is identified by **md1** and you are adding the drive **nvme0n5**.

```
# mdadm --manage /dev/md1 -a /dev/nvme0n5
```

h. To verify that the RAID is fully operational (i.e., not degraded), type the following command. The output will be similar to Figure 22. You see that all three drives are now active.

```
# mdadm -D /dev/md1
```

Figure 22 – Rebuilt RAID 5 Array

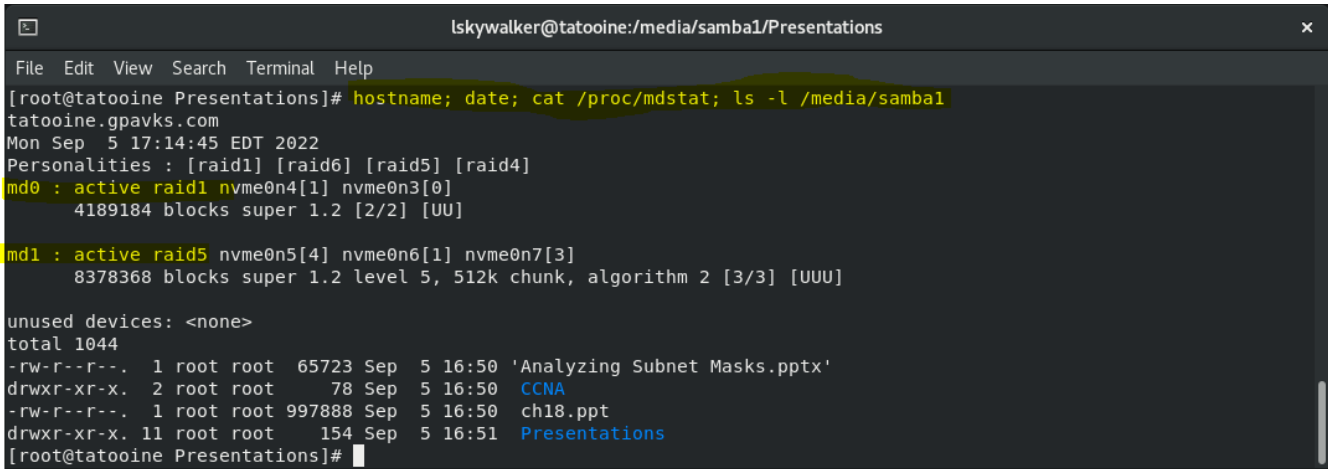

lskywalker@tatooine:/media/samba1/Presentations
File Edit View Search Terminal Help

```
[root@tatooine Presentations]# mdadm -D /dev/md1
/dev/md1:
           Version : 1.2
     Creation Time : Mon Sep  5 14:38:02 2022
        Raid Level : raid5
        Array Size : 8378368 (7.99 GiB 8.58 GB)
     Used Dev Size : 4189184 (4.00 GiB 4.29 GB)
      Raid Devices : 3
     Total Devices : 3
       Persistence : Superblock is persistent

       Update Time : Mon Sep  5 17:09:50 2022
             State : clean
    Active Devices : 3
   Working Devices : 3
    Failed Devices : 0
     Spare Devices : 0

            Layout : left-symmetric
        Chunk Size : 512K

Consistency Policy : resync

              Name : tatooine.gpavks.com:1  (local to host tatooine.gpavks.com)
              UUID : 9e1ffd44:b9fb468e:e9e1c63a:e58ae1fa
            Events : 44

    Number   Major   Minor   RaidDevice State
       4     259        6        0      active sync   /dev/nvme0n5
       1     259        7        1      active sync   /dev/nvme0n6
       3     259        8        2      active sync   /dev/nvme0n7
[root@tatooine Presentations]#
```

IMPORTANT NOTICE

For the report, include a **single** screenshot showing the output from the, `hostname`, and `date` commands. In the same screenshot enter the following commands to show the output from the, `cat /proc/mdstat`, and the `ls -l` command where your data is located on the mounted drive. The single screen shot must be in the lab report. Refer to Figure 23 for an example.

Figure 23 –Rebuilt RAID Verification


lskywalker@tatooine:/media/samba1/Presentations
File Edit View Search Terminal Help

```
[root@tatooine Presentations]# hostname; date; cat /proc/mdstat; ls -l /media/samba1
tatooine.gpavks.com
Mon Sep  5 17:14:45 EDT 2022
Personalities : [raid1] [raid6] [raid5] [raid4]
md0 : active raid1 nvme0n4[1] nvme0n3[0]
      4189184 blocks super 1.2 [2/2] [UU]

md1 : active raid5 nvme0n5[4] nvme0n6[1] nvme0n7[3]
      8378368 blocks super 1.2 level 5, 512k chunk, algorithm 2 [3/3] [UUU]

unused devices: <none>
total 1044
-rw-r--r--.  1 root root  65723 Sep  5 16:50 'Analyzing Subnet Masks.pptx'
drwxr-xr-x.  2 root root     78 Sep  5 16:50  CCNA
-rw-r--r--.  1 root root 997888 Sep  5 16:50  ch18.ppt
drwxr-xr-x. 11 root root    154 Sep  5 16:51  Presentations
[root@tatooine Presentations]#
```

Activity 7 – Logical Volume Management

Logical Volume Management is similar to RAID; however, it offers much more flexibility and efficiency needed for modern Linux storage systems. In this activity, you will use the remaining two drives to configure them to function as a single logical drive, using Logical Volume Management.

It's important that you understand some of the terminology associated with LVM management. Below is a summary of key terms you will need to know for this activity.

Physical Volume: A physical block device, such as /dev/sda. These are used as the building blocks for volume groups and logical volumes.

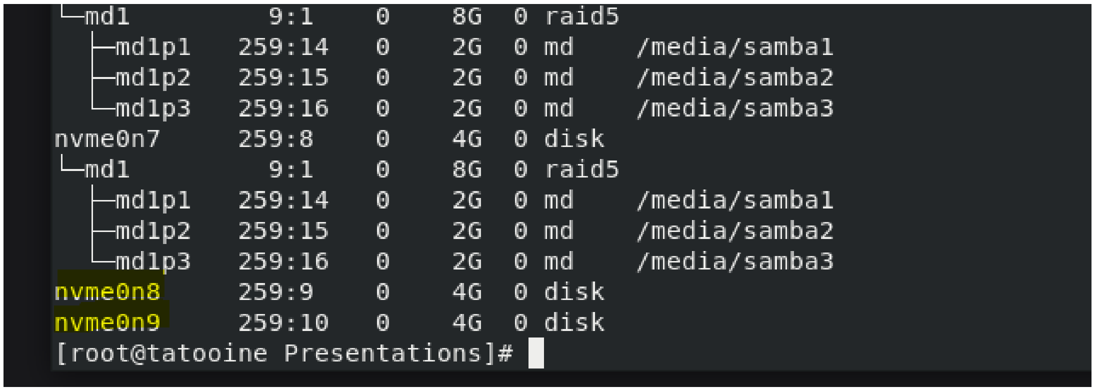
Volume Group: A combination of one or more physical volumes in a single logical storage pool. Logical volumes reside in volume groups.

Logical Volume: Are part of a volume group, similar to a partition, and can be mounted like regular partitions in a Linux File System.

Creating logical volumes starts with initializing the physical volumes (pv) that are needed to create the volume groups (vg), from which logical volumes (lv) can be created.

a. Use the **`lsblk`** command to identify the remaining two drives (Figure 24). If you have been following the instructions closely the remaining drives are nvme0n8 and nvme0n9.

Figure 24 – LVM Drives



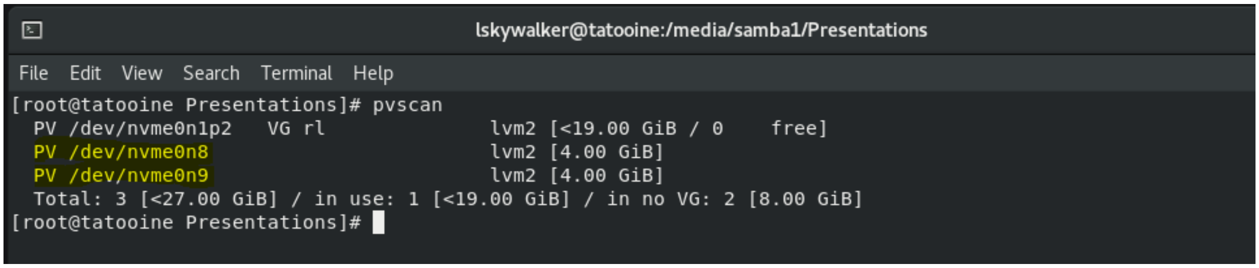
```
 └─md1         9:1     0    8G  0 raid5
   ├─md1p1   259:14    0    2G  0 md    /media/samba1
   ├─md1p2   259:15    0    2G  0 md    /media/samba2
   └─md1p3   259:16    0    2G  0 md    /media/samba3
nvme0n7      259:8     0    4G  0 disk
 └─md1         9:1     0    8G  0 raid5
   ├─md1p1   259:14    0    2G  0 md    /media/samba1
   ├─md1p2   259:15    0    2G  0 md    /media/samba2
   └─md1p3   259:16    0    2G  0 md    /media/samba3
nvme0n8      259:9     0    4G  0 disk
nvme0n9      259:10    0    4G  0 disk
[root@tatooine Presentations]#
```

a. Next, initialize the two drives as physical volumes using the **`pvcreate`** command followed by the drives to be initialized.

```
# pvcreate /dev/nvme0n8 /dev/nvmw0n9
```

b. To confirm that the drives are part of the physical volume use the **`pvscan`** command. The output will be similar to Figure 25.

Figure 25 – Sample **pvscan** Output


lskywalker@tatooine:/media/samba1/Presentations
File Edit View Search Terminal Help

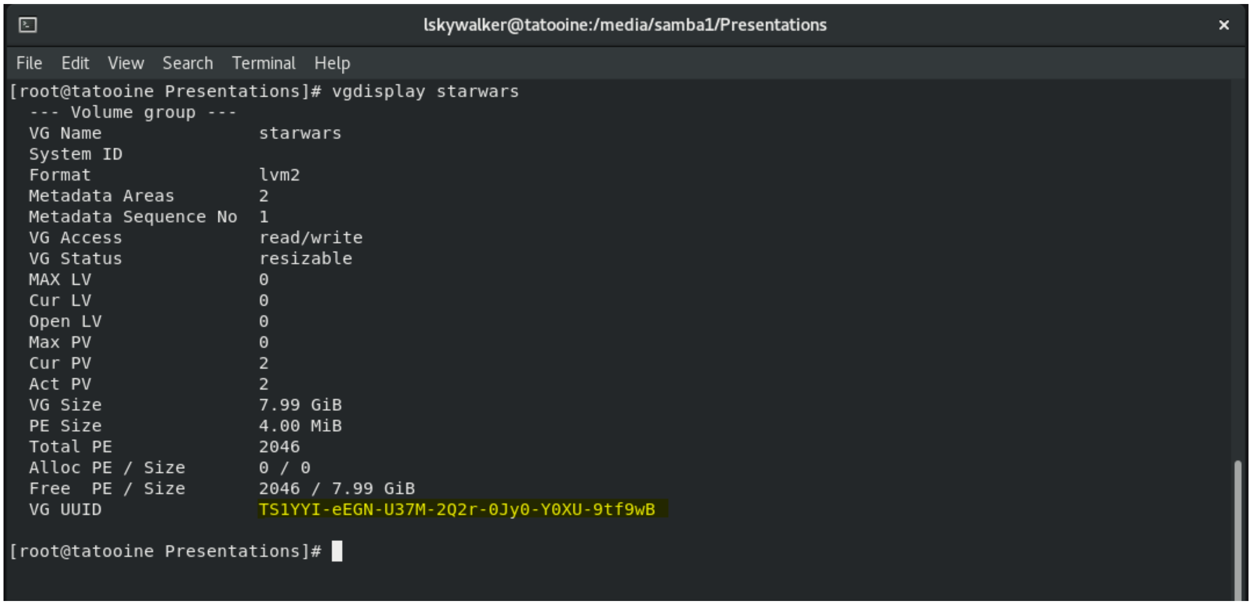
```
[root@tatooine Presentations]# pvscan
  PV /dev/nvme0n1p2   VG rl              lvm2 [<19.00 GiB / 0    free]
  PV /dev/nvme0n8                        lvm2 [4.00 GiB]
  PV /dev/nvme0n9                        lvm2 [4.00 GiB]
  Total: 3 [<27.00 GiB] / in use: 1 [<19.00 GiB] / in no VG: 2 [8.00 GiB]
[root@tatooine Presentations]# 
```

c. Notice in Figure 25 there is no volume group (no VG), so we need to create one. Use **vgcreate** to create the volume group by entering the following command and passing the volume group name (you can name the group whatever you want, have at it) and the drives as arguments. **Please note** that for the remainder of the instructions, all commands will assume that "*starwars*" is the name used to identify the volume group.

```
# vgcreate starwars /dev/nvme0n8 /dev/nvme0n9
```

d. You will receive a message that the volume group was created successfully. However, the **vgscan** command can also be used and is especially useful on systems you are not familiar with. To verify that both of the physical drives are members of the volume group use the **pvscan** command. Alternatively, you can use the **vgdisplay** command and pass the name of your volume group as the argument for more detailed information (Figure 26). One particular piece of information is the **Volume Group UUID**, which appears as the last entry.

Figure 26 – Example **vgdisplay** Output


lskywalker@tatooine:/media/samba1/Presentations
×
File Edit View Search Terminal Help

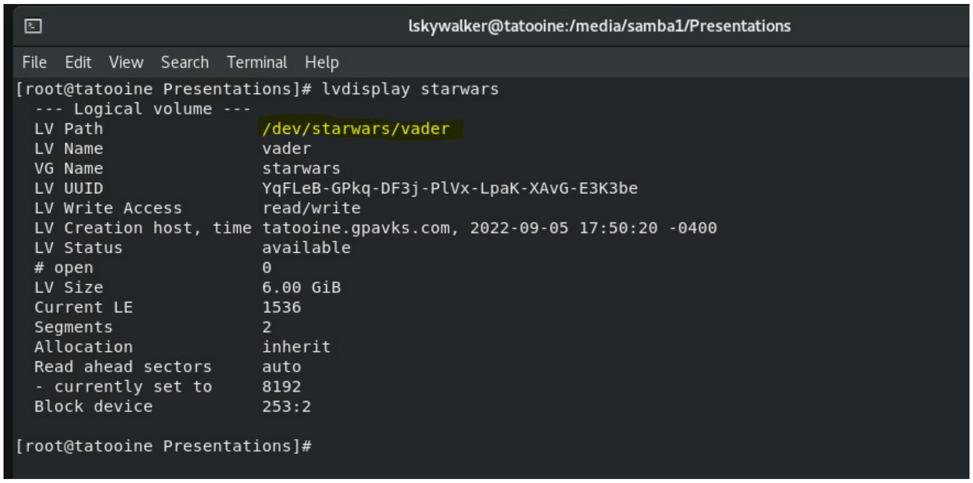
```
[root@tatooine Presentations]# vgdisplay starwars
  --- Volume group ---
  VG Name               starwars
  System ID             
  Format                lvm2
  Metadata Areas        2
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                2
  Act PV                2
  VG Size               7.99 GiB
  PE Size               4.00 MiB
  Total PE              2046
  Alloc PE / Size       0 / 0
  Free  PE / Size       2046 / 7.99 GiB
  VG UUID               TS1YYI-eEGN-U37M-2Q2r-0Jy0-Y0XU-9tf9wB

[root@tatooine Presentations]# 
```

e. To create the logical volume from the volume group, use the **lvcreate** command with the following parameters and make it 6G. Use the manual page for specific information on the arguments being passed. In the example, "vader" is the name of the logical volume and will be used for the remainder of the instructions.

```
# lvcreate -L 6G -n vader starwars
```

f. The **pvscan** command will show that the Logical Volume is now "active". Just like the partitions you created earlier you will need **format** the logical volume, create a **mount point**, and edit **fstab** to make it persistent. There are several ways to find the path to the volume, I prefer using **lvdisplay** (see Figure 27). Instead of using the UUID however use the Logical Volume Path, to find it, enter the **lvdisplay** command in the terminal. **Please note** that for the remainder of the instructions, all commands will assume that "vader" is the name used to identify the Logical volume.

Figure 27 – Logical Volume Path


lskywalker@tatooine:/media/samba1/Presentations
File Edit View Search Terminal Help

```
[root@tatooine Presentations]# lvdisplay starwars
  --- Logical volume ---
  LV Path                /dev/starwars/vader
  LV Name                vader
  VG Name                starwars
  LV UUID                YqFLeB-GPkq-DF3j-PlVx-LpaK-XAvG-E3K3be
  LV Write Access        read/write
  LV Creation host, time tatooine.gpavks.com, 2022-09-05 17:50:20 -0400
  LV Status              available
  # open                 0
  LV Size                6.00 GiB
  Current LE             1536
  Segments               2
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     8192
  Block device           253:2

[root@tatooine Presentations]#
```

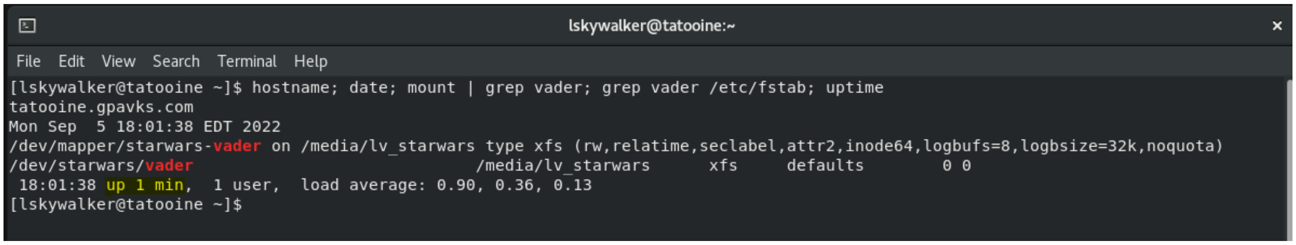
g. Reboot the system. Once the system reboots, open a terminal and be prepared to issue commands to obtain a screenshot for the lab report.

IMPORTANT
NOTICE

For the report, include a **single** screenshot showing the output from the, **hostname**, and **date** commands. In the same screenshot enter the following commands to show the output from the, **mount | grep vader**, **grep starwars /etc/fstab** and the **uptime**. Refer to Figure 28 for an example.

Important – The uptime in the screenshot must show 5 minutes or less.

Figure 28 - LVM Mount Verification



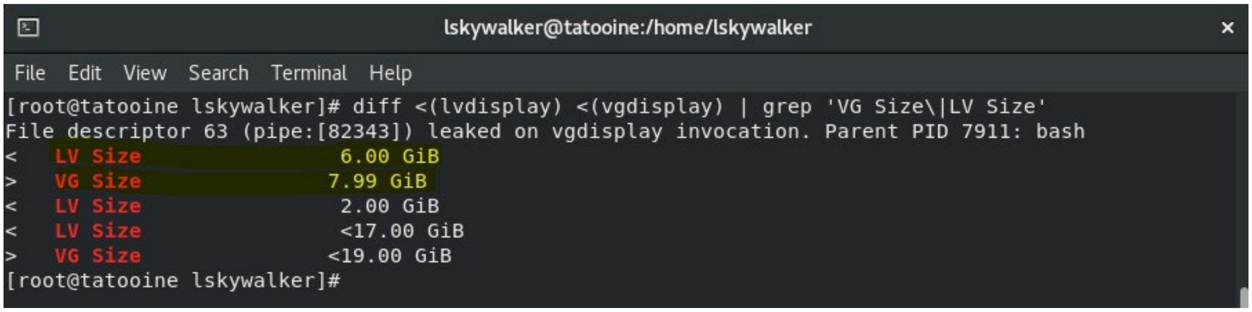
```
[lskywalker@tatooine ~]$ hostname; date; mount | grep vader; grep vader /etc/fstab; uptime
tatooine.gpavks.com
Mon Sep  5 18:01:38 EDT 2022
/dev/mapper/starwars-vader on /media/lv_starwars type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/starwars/vader                               /media/lv_starwars      xfs     defaults        0 0
 18:01:38 up 1 min,  1 user,  load average: 0.90, 0.36, 0.13
[lskywalker@tatooine ~]$
```

h. Once you have verified that the system boots successfully and all devices can auto mount, try pushing the *fstab* file to the Git repository on Serenity. Remember this is not the same device, so think about the process you need to go through and the Git workflow.

i. For the final task in this activity, we are going to “grow” the LVM logical volume. If you remember when we created the Volume Group, we made it from two 4GB drives, but when we created the Logical Volume, we only made it 6GB, leaving about 2GB available. Run the following command as root to compare the Logical Volume size with the Volume Group Size. The output should be similar to Figure 29, ignore the message.

```
# diff <(lvdisplay) <(vgdisplay) | grep ‘VG Size\|LV Size’
```

Figure 29 – Logical Volume versus Volume Group Size


lskywalker@tatooine:/home/lskywalker
File Edit View Search Terminal Help

```
[root@tatooine lskywalker]# diff <(lvdisplay) <(vgdisplay) | grep 'VG Size\|LV Size'
File descriptor 63 (pipe:[82343]) leaked on vgdisplay invocation. Parent PID 7911: bash
<   LV Size                6.00 GiB
>   VG Size               7.99 GiB
<   LV Size                2.00 GiB
<   LV Size                <17.00 GiB
>   VG Size               <19.00 GiB
[root@tatooine lskywalker]#
```

j. Adding additional space to a Logical Volume is easy, all you need to is run the `lvextend` command. But before we do that, we are going to output the commands to a text file to record the size increase. In the following example the date command is put into a file called “record.log”. Be mindful of your current directory because that is where this file will be created.

```
# date > record.log
```

k. Next, write the current size of the Logical Volume to the same log file by appending the output of the following command.

```
# lvdisplay | grep 'LV Size' >> record.log
```

l. The next command will extend the Logical Volume “MyVolGroup-testLV” by 500MB.

```
# lvextend -L +500M /dev/starwars/vader
```

m. Next, we need to resize the file system by entering the following command.

```
# xfs_growfs /dev/mapper/MyVolGroup-testLV
```

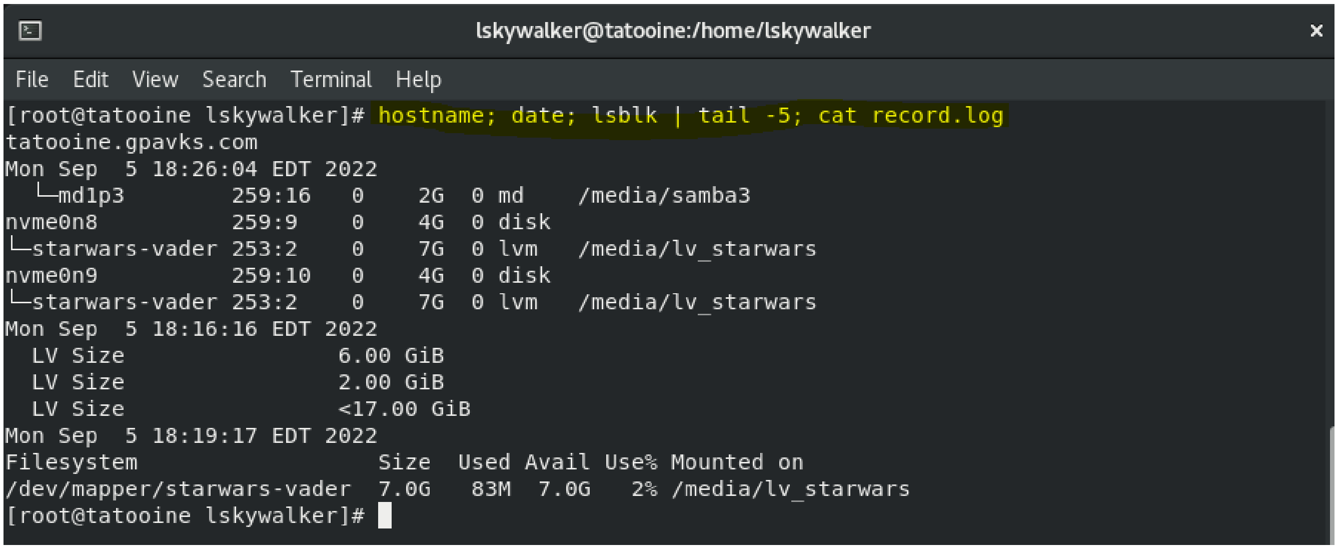
n. Enter the following commands to record the change to the record.log file.

```
# date >> record.log
# df -h /media/MyLogVol >> record.log
```

IMPORTANT NOTICE

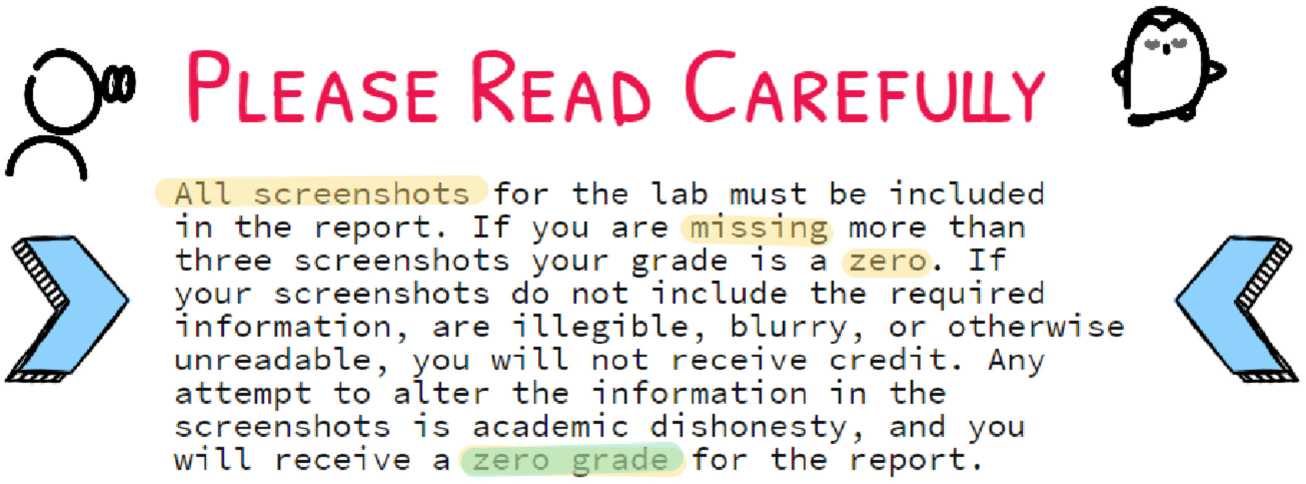
For the report, include a **single** screenshot showing the output from the, `hostname`, and `date` commands. In the same screenshot enter the following commands, `lsblk | tail -5`, and `cat record.log`. Refer to Figure 24 for an example.

Figure 30 – Resize Volume Verification



```
[root@tatooine lskywalker]# hostname; date; lsblk | tail -5; cat record.log
tatooine.gpavks.com
Mon Sep  5 18:26:04 EDT 2022
  └─md1p3        259:16   0    2G  0 md    /media/samba3
nvme0n8          259:9    0    4G  0 disk
└─starwars-vader 253:2    0    7G  0 lvm   /media/lv_starwars
nvme0n9          259:10   0    4G  0 disk
└─starwars-vader 253:2    0    7G  0 lvm   /media/lv_starwars
Mon Sep  5 18:16:16 EDT 2022
  LV Size                6.00 GiB
  LV Size                2.00 GiB
  LV Size               <17.00 GiB
Mon Sep  5 18:19:17 EDT 2022
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/starwars-vader  7.0G   83M  7.0G   2% /media/lv_starwars
[root@tatooine lskywalker]# 
```

Screenshot Summary



Please Read Carefully

All screenshots for the lab must be included in the report. If you are missing more than three screenshots your grade is a zero. If your screenshots do not include the required information, are illegible, blurry, or otherwise unreadable, you will not receive credit. Any attempt to alter the information in the screenshots is academic dishonesty, and you will receive a zero grade for the report.

All screenshots must be labeled, using the following titles.

Figure 1 – Drives Added Verification
Figure 2 – Drive Mounted Verification
Figure 3 – RAID 1 Mount Verification
Figure 4 – RAID 5 Mount Verification
Figure 5 – Persistent Mount Verification
Figure 6 – RAID Redundancy
Figure 7 – Rebuilt RAID Verification
Figure 8 – LVM Mount Verification
Figure 9 – LVM Resize Verification