CMIT 391 Linux System Administrator

Linux Virtual Installation

UMUC

Dmitry Landy

August 1 , 2019

LinuxVirtual Installation

Although transitioning IT work stations from Windows over to Linux can seem intimidating, there are certain steps that can be taken to ensure a successful transition. The Linux operating system (OS) comes in a large variety of flavors referred to as “distributions”. One such distribution is Linux Mint, specifically the Cinnamon desktop environment, which is based on Debian and Ubuntu (Linux Mint, n.d.). This distribution has a design to it that makes Windows users feel more familiarized with it due to its layout (King, 2018). This report will walk through and discuss the various steps of installing Linux Mint. This installation will be done using the free virtual machine (VM), Oracle VM VirtualBox. The VM will have two hard drives, one will be 50 GB, and the other will be 10 GB. 2GB of RAM will be allocated to the VM. The installation process will be done without an internet connection, but will have a connection afterwards for the post-installation tasks. The topics discussed will cover ISO image download process, installation process, post-installation tasks, and VM functionality.

ISO Image Download Process

Once the Linux distribution has been discussed and selected, which is Linux Mint for this report, then the next step is to download the ISO image. Although this report is using this method, installing from an optical drive or thumb drive are other options (TestOut, n.d.). To locate the installation disk image, follow the steps below:

1) Open a web browser and go to “https://linuxmint.com/download.php”.
2) Locate the latest version of Linux Mint and select the ‘64-bit’ link for Cinnamon.
3) Locate the “Download Mirrors” section and select a mirror link to download the file.

Downloading the file can be a time and resource consuming task. To remedy this, a good idea would be to download the file once onto a shared server so that it is readily available for multiple users. Once the file has been downloaded, TestOut section 2.2.1 suggests that the checksum value of the file is verified. Following the steps below will ensure that the correct file arrived without any corruption:

1) Go to “https://linuxmint.com/verify.php” and select the version of the file downloaded.
2) Follow the steps under “preparation” (Linux Mint, n.d.).
3) Go to “https://forums.linuxmint.com/viewtopic.php?f=42&t=291093”.
4) Follow the steps for the integrity and authenticity check that are outlined (Gm10, 2019).

A successful (passed) integrity and authenticity check will result in a similar output as shown in the image below:

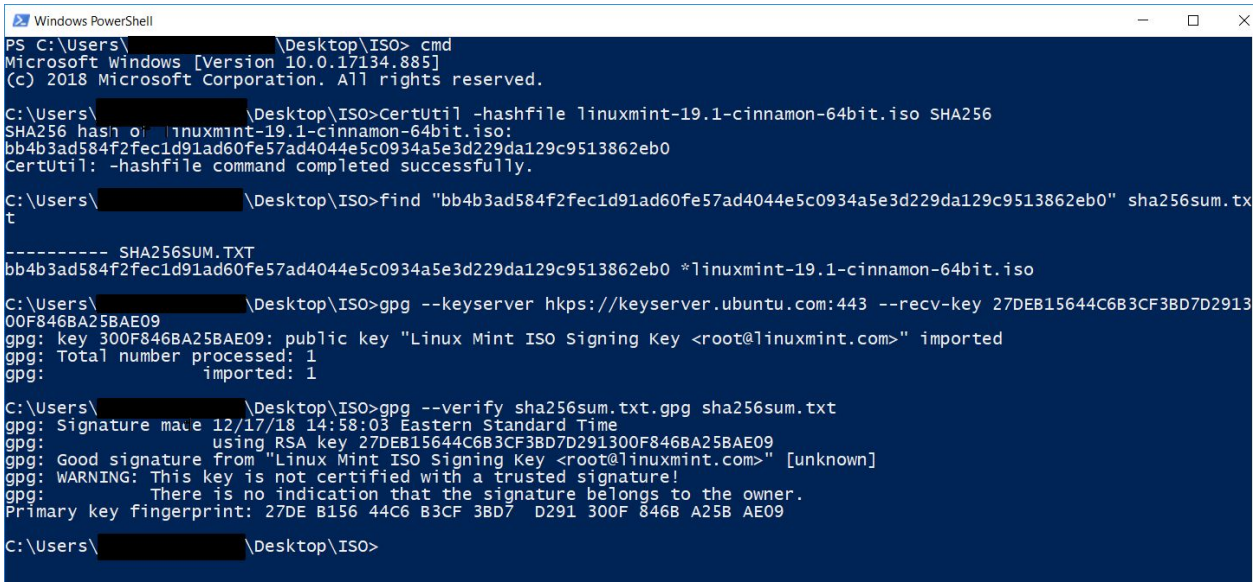

Figure 1. PowerShell output example of a successful integrity and authenticity check for an ISO image.

Installation Process

Now that the ISO image has been has an integrity check and verified for authenticity, the image can be installed. Although this report will install Linux Mint onto a VM, other options for installation include USB drive, optical disk, and server (TestOut, 2.2.1). An ISO image can be burned to a USB drive or optical disk, such as a CD. A USB drive has an advantage over an optical disk due to the speed USB drives provides. However, a key disadvantage to both is that they can only be used to install one system at a time. Installing from a server, local or on the internet, eliminates that issue. The advantage is that multiple workstations to access the server and install concurrently. The disadvantage is that the installation is slower as it depends on the network's speed.

Once the method of installation has been decided on, it is time to install the distribution to the system. If installing from a USB drive or optical disk, the system should boot from those devices. If installing from a server, the system should boot from the network installation image. Once the system has properly booted from the appropriate source, the following screen should appear:

Figure 2. Image of Linux Mint "live session" mode after system boot.

The "live session" mode that is available prior to installation allows the user to familiarize themselves with the system and try anything they would like since the changes made will not be permanent (Linux Mint Installation Guide, 2017). After familiarization in "live session" mode, select "Install Linux Mint" to begin the installation process.

The first screen is to select the keyboard layout, which is English in this case. There are multiple options for each keyboard layout that cater to those with disabilities. The second screen gives an option to select a box to install necessary multimedia codecs. According to the Multimedia Codecs section of the Linux Mint Installation Guide (2017), if there is going to be an active internet connection during the installation process, the box should be selected to avoid having to install later. The third screen gives options for the installation type. Since the VM has no previous OS, the "Erase disk and install Linux Mint" option is the only one selected.

However, if there is going to be another OS with Linux Mint, then the “Something else” option would be selected instead to partition the disk appropriately. This option can also be used to partition the disk. However, this report will discuss disk partitioning process when discussing the post-installation tasks, since the VM is set up with two hard disks. If there is only one hard disk available, it is encouraged to partition the disk during this step by selecting the “Something else” option.

Once the installation type has been selected, there are two additional options to explore on the screen. The first option available on the third screen is to check the “Use LVM with the new Linux Mint installation” option. TestOut section 7.3.1 states that LVM, or logical volume manager, is an alternative to the standard disk partitioning process which allows the creation of storage area known as “volume groups”. These volume groups are mounted on mount points and have a lot of flexibility to change the amount of storage space is available. If a volume starts to run out of space, then it can get more memory space by taking from another volume group or another physical disk. The second option available is “Encrypt the new Linux Mint installation for security”, which gives the option for full disk encryption.

The fourth screen allows the user to select the appropriate time zone. This can be accomplished by selecting an area on the map, or by typing in the city, followed by selecting the state and country. The fifth screen has empty areas that are reserved for the user’s information, such as name, username, and password. As the Linux Mint Installation Guide describes, the computer name is the name of the host machine that will be displayed on the network, whereas the username is what the user logs in as (2017). For security reasons, select “Require my password to log in” and create a strong password. After filling out this section appropriately, the

system will begin installing. After the installation, an option to continue “live session” or restart the system is available. Restart the system to proceed to post-installation tasks.

Post-Installation Tasks

Now that Linux Mint has been properly installed, there are some post-installation tasks that should be performed. When the system is booted, the user will be prompted to login. Once the user is successfully logged in and the desktop environment is loaded, the user can prepare to conduct the following tasks:

1) Create the root account password.
2) Confirm network connection.
3) Update the system.
4) Create disk partitions.
5) Check running processes.

Create the Root Account Password

By default, the root account does not have a password assigned. This creates an issue with access and security. The /etc/shadow file can be used to see the status of an account’s password, in this case root. By using the *cat* command on the /etc/shadow file, the user can see the entire contents of the file and find the line with root. To view the full listing in an organized manner, the *less* command can be used to move up and down the output. The following is the output of “sudo cat /etc/shadow | less”:

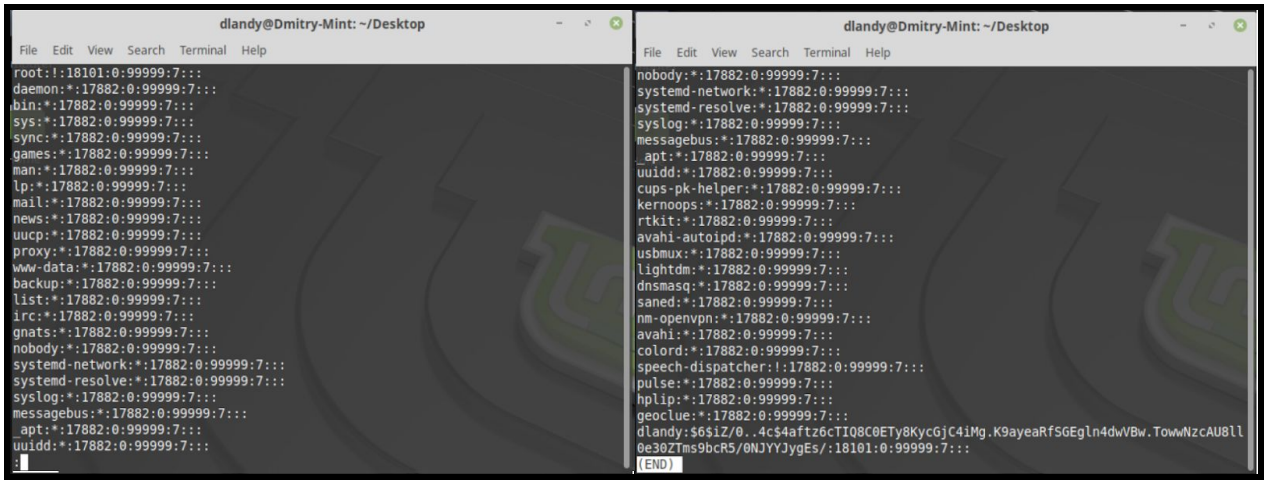

Figure 3. Terminal output of "sudo cat /etc/shadow |less".

By piping ("|") the output of the /etc/shadow contents into the *less* command, the user is able to maneuver through the output at an appropriate rate by using the space bar or the arrow keys, or press "q" to quit. *Sudo* is a command used to allow a user to execute a command as a superuser, which is required when accessing the /etc/shadow file. Although the output does have root in there, using the *grep* command would be a more efficient way to find the correct entry. *Grep*, which stands for "global regular expression print", is used to search for patterns in a file. This command can be used to find the entry containing "root" by using the command "sudo cat /etc/shadow | grep root".

Figure 4. Terminal output of *grep* command used to find an entry with "root".

This output shows only the entry with the requested term and also highlights it. The second field in the entry is for the password. Section 6.1.4 of TestOut (n.d.) states that "!" or "!!" in the password field indicates that the account can not be used to log in and is locked. This is one indicator that the password should be set for this account.

Since, only the user account created is part of the "sudo" group, that account has escalated privileges. With this, the user account can be used to access the root account and set the password. To set root's password, use "sudo passwd sudo" and input the user's password. Section 6.2.4 of TestOut states that the *passwd* command is used to change the password of an account.

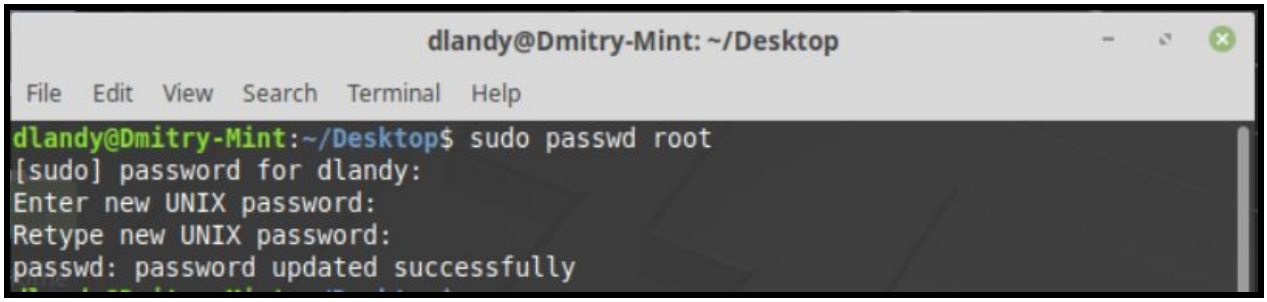

Figure 5. Terminal output showing a successful password change for the root account.

Now that the password has been set, the change should be confirmed by viewing the /etc/shadow file once more. A method to locating root in the /etc/shadow file would be to use the *head* command. This command will output the top 10 lines of a text file by default, although the number can be modified by using the "-n" option. Similarly, the *tail* command will display the last 10 lines of an output, but can also be specified to display more of less. Since root account is known to be at the top of the file, using the *head* command would be more efficient than outputting the contents of the whole file, though not as efficient as using *grep*.

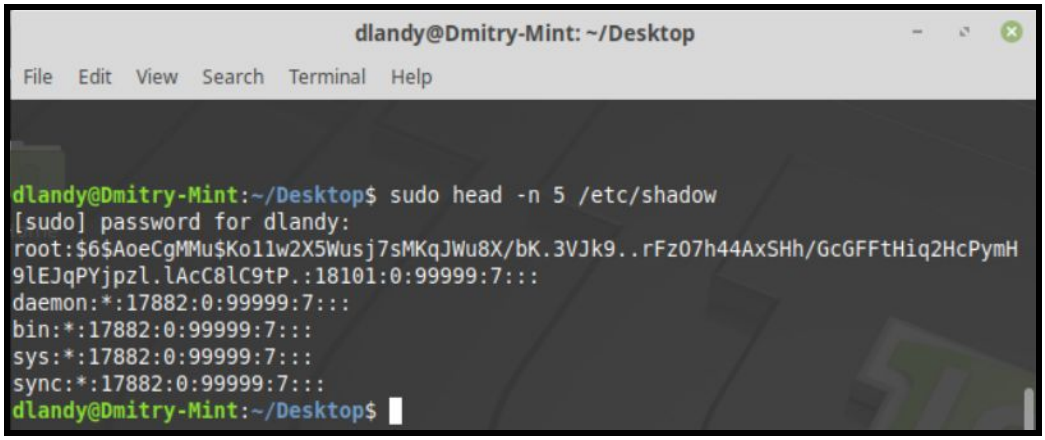

Figure 6. Output of the top five lines in the /etc/shadow file.

The terminal output is quite different from the original output since the "!" in the password field has now been replaced with a series of characters. This is the password hash for the root account's password. The "6" at the beginning indicates that the hash used a SHA-512 algorith to produce the hash (Pillai, 2013). Now, the root user can be properly accessed by using the created password.

Confirm Network Connection

Prior to conducting any other task, the system needs to be fully updated to maximize system security. However, unless the system updates are being retrieved from a local system on the network, they will have to come from an online repository. To download from the repository, a network connection out to the internet needs to be available. The *ifconfig* command can be used to identify the status of the current network connect, if there is one. TestOut section 11.2.3 discusses the appropriate use of the *ifconfig* command. *Ifconfig*, which stands for "interface configuration", is used to view and configure network interfaces. The following image shows an example of the command:

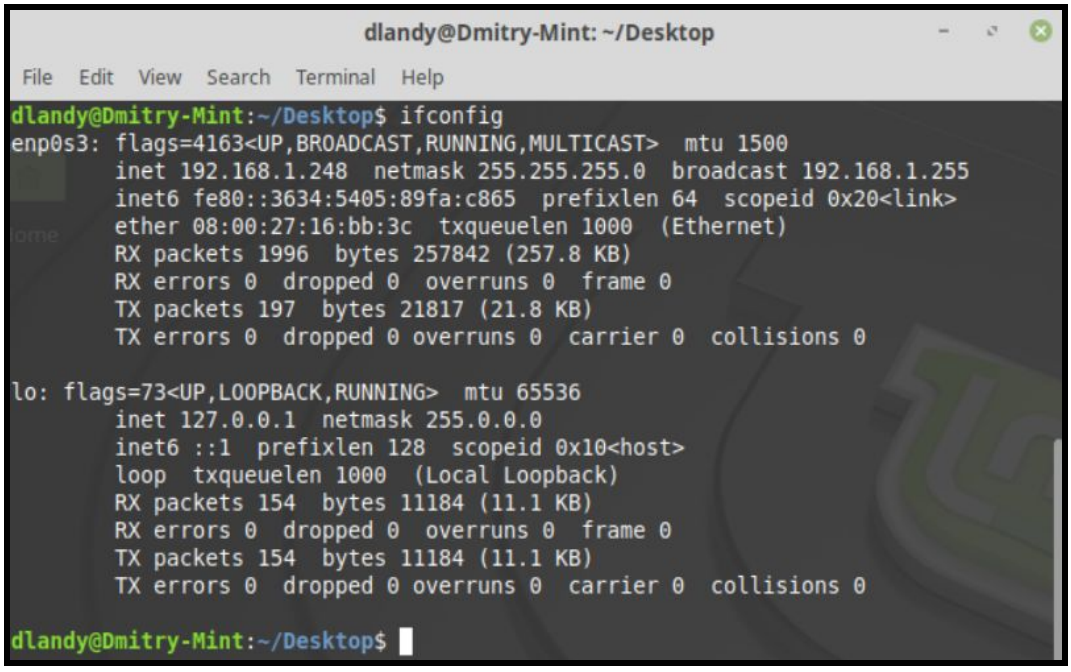

Figure 7. Result of the *ifconfig* command showing two interfaces.

The result shown reveals two active interfaces. The bottom interface, “lo”, is the default loopback address that is linked to the system itself. The top interface, listed as “enp0s3”, shows the standard network interface, which is wireless in this case. This shows that it has an assigned IPv4 (inet) and IPv6 (inet6) address with 1996 packets received (RX), 154 packets transmitted (TX), and no errors for either.

Update the System

Having confirmed an active internet connection, the system can now be updated. Since Linux Mint is Debian-based, the package management utility is *apt-get.* Section 5.3.2 of TestOut states that APT, which stands for “Advanced Packaging Tool”, is a suite of tools that includes the *apt-get* utility, which is used to manage packages. This will be used to update the system with the latest updates available. According to TestOut, using the “update” option for *apt-get* will allow the system to update the information about packages found in /etc/apt/sources.list.

Using the “dist-upgrade” option for *apt-get*, the system will upgrade all of the packages on the system with the newest versions available.

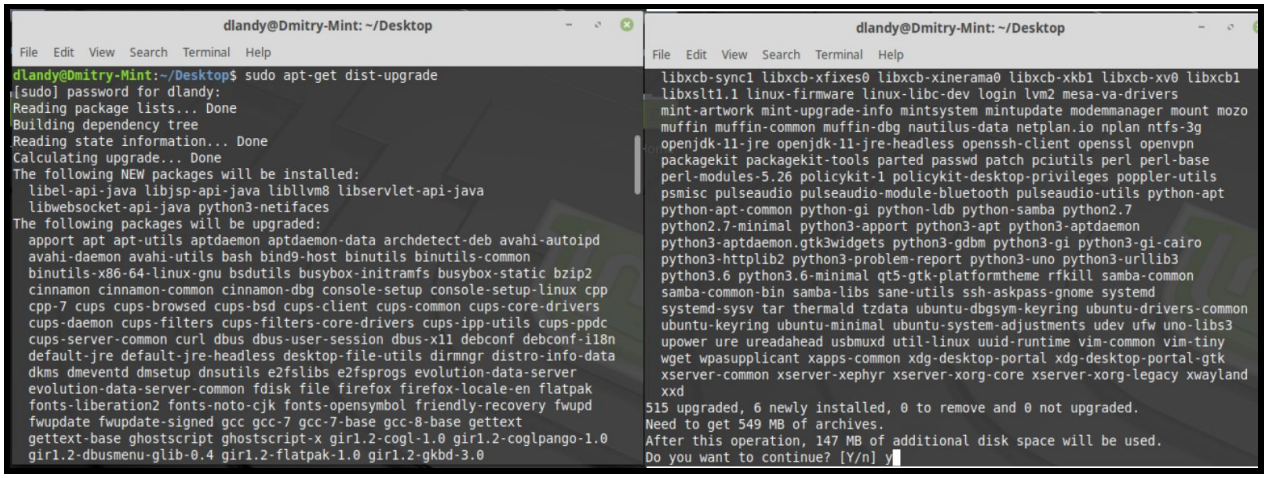

Figure 8. Partial terminal output of the command “sudo apt-get dist-upgrade”.

Upgrading every package in the system after an initial installation requires a decent amount of memory, over 500 MB in this case, that should be accounted for prior to the upgrade. When the packages are ready to be installed, the following screen will show:

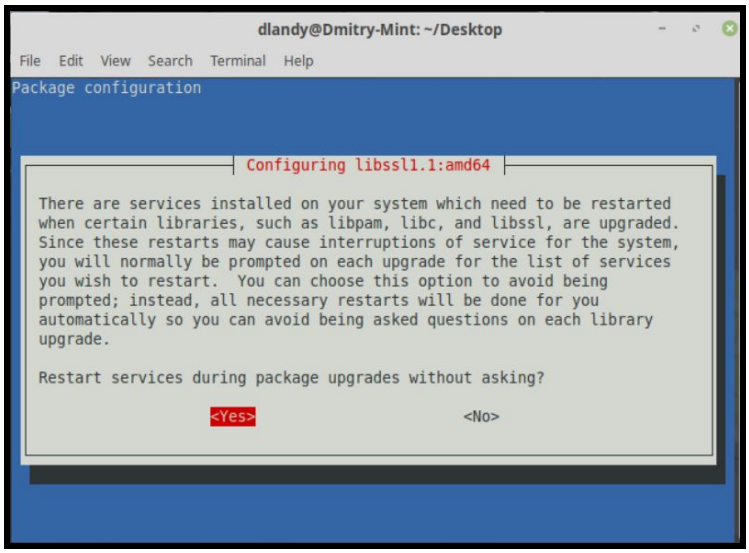

Figure 9. Permission screen to restart services for requested system upgrade.

Selecting “Yes” will allow each service to be restarted automatically, instead of prompting for each service. Once the download is completed, restart the system to ensure all changes are fully implemented. This is done using the “reboot” option for the *systemctl* command, which controls the state of the system. By using the command “systemctl reboot”, the system will immediately restart.

Create Disk Partitions

Now that the system is fully updated, disk partitions can be created using the *fdisk* utility. According to section 7.1.1 of TestOut, the *fdisk* utility is used to create, delete, and view disk partitions from the terminal. To see what hard disks are available and what partitions are created, use the *ls* command to show /dev/sd*. This will list all files in the /dev directory that start with “sd”.

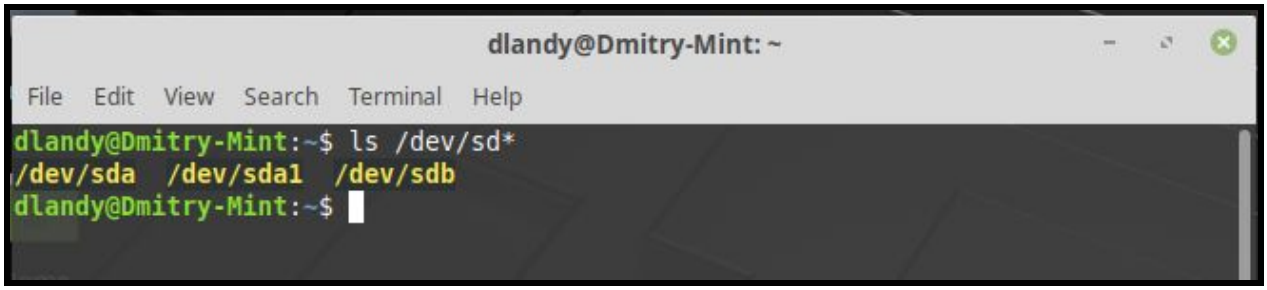

Figure 10. List of all files in the /dev directory starting with “sd”.

The output shown confirms that there are two hard disks, sda and sdb, but only one partition created , sda1. Knowing that, the “sdb” drive can be selected to create partitions using *fdisk.* To do this, the *fdisk* utility has to be run along with the device path by entering the command “sudo fdisk /dev/sdb”. Then, “n” is entered to create a new partition.

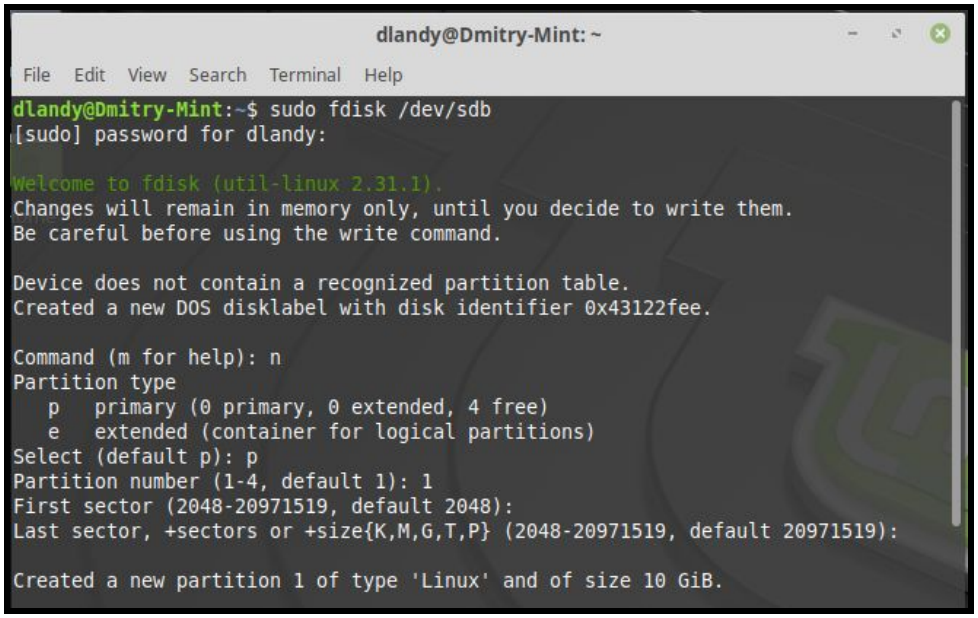

Figure 11. Using the *fdisk* utility to create a new partition for the "sdb" drive.

After "n" is input, the first step is to select the type of partition, extended or primary. Since this is the only partition on the drive, this will be a primary partition ("p"). The second step is to select the partition number, which is "1". The third and fourth steps are to select the first and last sectors, respectively, which can be done by just pressing "Enter" on the keyboard for both.

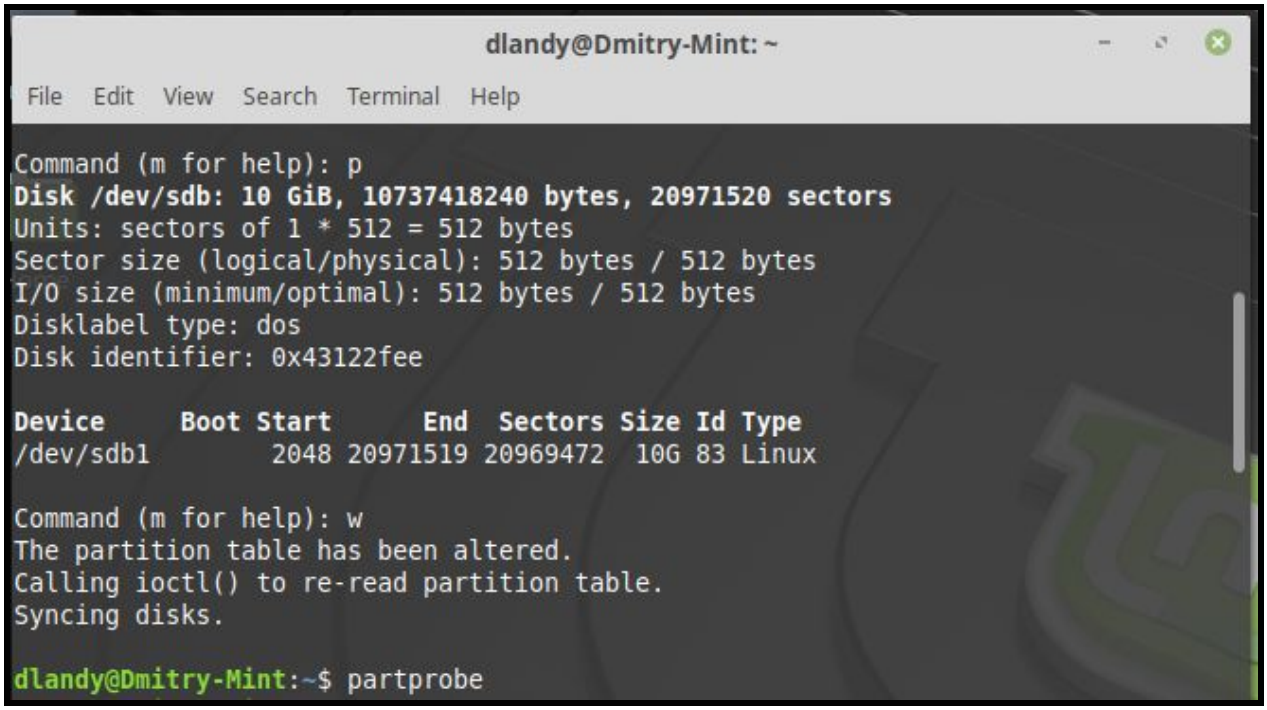

Figure 12. Using *fdisk* utility "p" and "w" options, followed by the *partprobe* command.

After creating the partition, entering the "p" option will show the partition table to confirm the partition is created correctly. To complete the creation of the new partition and writes the changes to the disk, "w" needs to be input. To avoid having to restart the system to read the new partition, the *partprobe* command can be used to have the kernel recognize it immediately. Completing these steps will allow for a partition to be created successfully.

Check Running Processes

The last task of the post-installation tasks is to check running processes to ensure the system processes are working appropriately. One option is to use the *top* utility, which outputs a dynamic table with multiple categories, which can be customized with additional options. The following shows an example of the output:

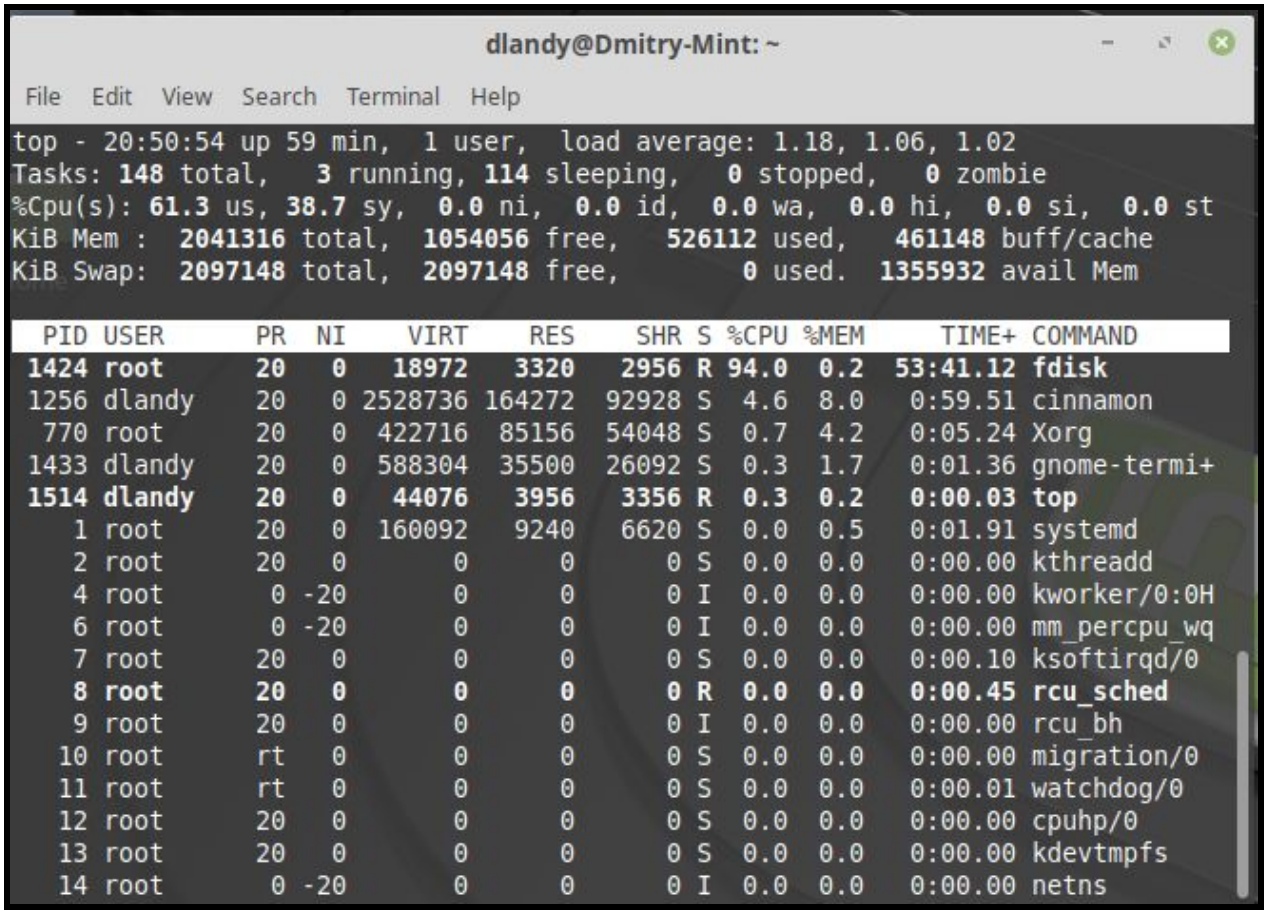

Figure 13. Output of the "top" command.

Section 9.1.5 of TestOut explains that this output is broken down into multiple sections, such as the PID, or process ID, the user, priority (PR), the nice value (NI), virtual memory used (VIRT), RAM used (RES), shared memory used (SHR), process' status (S), percentage of CPU time used (%CPU), percentage of RAM used (%MEM), amount of CPU time consumed (TIME+), and the command run (COMMAND). The MEM column is very useful to find out how memory-intensive a process is. The status column (S) is useful to find any "zombie" processes, which are listed as "Z", for troubleshooting. However, this utility does not go into as much detail about each process as the *ps* command does.

The *ps* command, as shown in section 9.1.8 of TestOut, displays a snapshot of current processes in more detail, if given the appropriate options, than the *top* utility. The following example shows the *ps* command using the "-lf" options:

```
dlandy@Dmitry-Mint:~$ ps -lf
F S UID          PID    PPID  C PRI  NI ADDR SZ WCHAN  STIME TTY          TIME CMD
0 S dlandy      1440    1433  0  80   0 -  5610 wait   19:56 pts/1    00:00:00 bash
0 R dlandy      1574    1440  0  80   0 -  9916 -      21:17 pts/1    00:00:00 ps -l
dlandy@Dmitry-Mint:~$
```

Figure 14. Terminal Output of "ps -lf" for the "dlandy" user.

Although the "-lf" options only produced a few results, having added the "-e" option would show all the processes. This would produce too much output that would be best sorted in conjunction with the *grep* command.

```
dlandy@Dmitry-Mint: ~
File  Edit  View  Search  Terminal  Help
dlandy@Dmitry-Mint:~$ ps -elf |grep fdisk
4 S root      1423   971  0  80   0 - 21015 -      19:56 ?        00:00:00 sudo
fdisk /dev/sdb
4 R root      1424  1423 98  80   0 -  4743 -      19:56 ?        01:18:38 fdisk
 /dev/sdb
0 S dlandy    1565  1440  0  80   0 -  3607 pipe_w 21:16 pts/1    00:00:00 grep
--color=auto fdisk
dlandy@Dmitry-Mint:~$
```

Figure 15. Using the *grep* command to filter the output produced by the "ps -elf" command.

The output only showed entries that had "fdisk" in them due to the addition of the *grep* command. Utilizing *grep* with ps can be extremely helpful in filtering mass amounts of processes that can be output *ps* command.

VM Functionality

Using a VM, specifically Oracle VM VirtualBox, allows an administrator to access every option available to customize their OS, with very few disadvantages. This flexibility is extremely liberating and creates an environment that allows for mistakes by providing a sandbox environment.

Suppose a system administrator wanted to eventually switch all systems from Windows to Linux, but they were not sure what issues would arise during the process. Well a simple solution is to use a VM to run through the process and discover issues along the way. If there are any commands or programs that the administrator is struggling with, then they can work on it inside the VM. VMs provide a function known as “snapshots”, which is a copy of the virtual disk at that time. This acts as a backup to the system in case any issues occur. During the same process, suppose the administrator wanted to explore the possibilities of setting up LVMs. Since the administrator can take a snapshot of the system, they can explore setting up LVMs without any worry. If anything goes wrong, then the administrator can load the previous snapshot and try again.

The only real disadvantage to using VMs is that they are limited to the resources the host machine provides, which impacts the VMs performance. However, the solution to this is to upgrade the host machine components so that the VM can run as it needs to. With as much as a VM offers to administrators, it is an incredible option for discovery and experimentation.

References

Gm10. (2019). How to verify the ISO image on Windows [Online forum comment]. Retrieved from https://forums.linuxmint.com/viewtopic.php?f=42&t=291093.

King, B., Jr. (2018). 10 Reasons to Fall in Love With Linux Mint. Retrieved from https://www.makeuseof.com/tag/reasons-to-love-linux-mint/

Linux Mint. (n.d.). How to Verify ISO Images. Retrieved from https://linuxmint.com/verify.php

Linux Mint. (n.d.). About. Retrieved from https://linuxmint.com/about.php

Linux Mint Installation Guide. (2017). Install Linux Mint. Retrieved from https://linuxmint-installation-guide.readthedocs.io/en/latest/install.html

Pillai, S. (2013). How are passwords stored in Linux (Understanding hashing with shadow utils). Retrieved from https://www.slashroot.in/how-are-passwords-stored-linux-understanding-hashing-shadow-utils

TestOut. (n.d.). 2.2.1 Linux Installation. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 6.1.4 User and Group Facts. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 6.2.4 User Management Facts. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 7.1.1 MBR Disk Partitions. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 7.3.1 Logical Volume Manager (LVM). Retrieved from https://cdn.testout.com

TestOut. (n.d.). 9.1.5 Viewing Process Information with Top. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 9.1.8 Process Display Facts. Retrieved from https://cdn.testout.com

TestOut. (n.d.). 11.2.3 Viewing and Managing IPv4 Addresses. Retrieved from https://cdn.testout.com