INFO EXTRACTOR

Table Of Contents

Introduction

This project required me to write my own script and research how to find the command to display other information not known to me in areas like CPU usage, memory usage, active system services and status. I needed to write a script from scratch to automate these commands for a first time user and it should display all output in a clean readable output easy for anyone to read.

INFO EXTRACTOR

Methodologies

For the Public IP address I used ifconfig.io. Only the first 2 blocks had to be shown so it was printed with ".XX.XX" at the back to sensor the last 2 blocks.

```
curl -s ifconfig.io | awk -F. '{print "Your Public IP address is" " "  $1 "." $2 ".XX.XX" }'
```

I used a similar method to block out certain parts of the MAC address.

```
ifconfig | grep ether | awk '{ print $2 }' |  awk -F: '{ print "Your Mac address is" " " "XX" ":" $2 ":"  $3 ":" $4 ":" $5":" $6}'
```

Using commands I know, I started writing the script. But I faced the first issue as shown in the picture below. There are no spaces in the output.

```
Your Public IP Address is 42.60.XX.XX
Your IP is 192.168.94.255
```

I wanted to add a space between the commands so I searched for ways to do it which worked for me.

```
5   #public IP address
6   curl -s ifconfig.io | awk -F. '{print "Your Public IP Address is" " "  $1 "." $2 ".XX.XX" }'
7   echo
8   #private ip address
9   IPX=$(ip a |grep "inet "| awk '{print $4}'| tail -1 )
10  echo "Your IP is $IPX"
11
12
13
14
15
16

┌──(kali㉿kali)-[~/Scripting]
└─$ bash Project.sh
Your Public IP Address is 42.60.XX.XX

Your IP is 192.168.94.255
```

INFO EXTRACTOR

Following which I was finding a command to show CPU usage.

```
echo "Displayed below are your top 5 processes(shown in percentage):"
ps -e -o cmd,%cpu --sort=-%cpu | head -n 6
```

Then I found a webpage to show me what I needed to input to find the memory space of the machine:

```
#Display the Memory Usage, Free and Used -> This Script
disk1=$(free -m | grep 'Mem' | awk '{print $2}')
disk2=$(free -m | grep 'Mem' | awk '{print $3}')
disk3=$(free -m | grep 'Mem' | awk '{print $7}')

echo "Listed below is how much disk space you have:
```

Next I was looking at how to display the active system services and status.

```
#Display your Active system services and status -> This script s
echo "Listed below are you active system services and status:"
systemctl list-units --type=service --state=running
```

Lastly I search how to find the largest files in the home directory and how to present it

```
#Display the top 10 files(size) from the `/home` directory ->
display=$(du -a -h /home |sort -n -r | head -n 10)
echo "Listed below are the top 10 files in your Home directory
```

INFO EXTRACTOR

Discussion

I first determined how many factors I needed to display on the output.

I wrote all of these out in the script with a '#' symbol so it won't be displayed in the command prompt but it is only for me to know where I need to focus when writing the script.

```
1   #!/bin/bash
2
3
4   #Public IP Address (censor the last two bytes ⇒ last two blocks of numbers)
5   #Internal IP address of the machine
6   #The MAC address of the machine(censor the 1st section)
7   #Display the top 5 processes' CPU usage(percentage)
8   #Display the Memory Usage, Free and Used
9   #Display your Active system services and status
10  #Display the top 10 files(size) from the `/home` directory
11
```

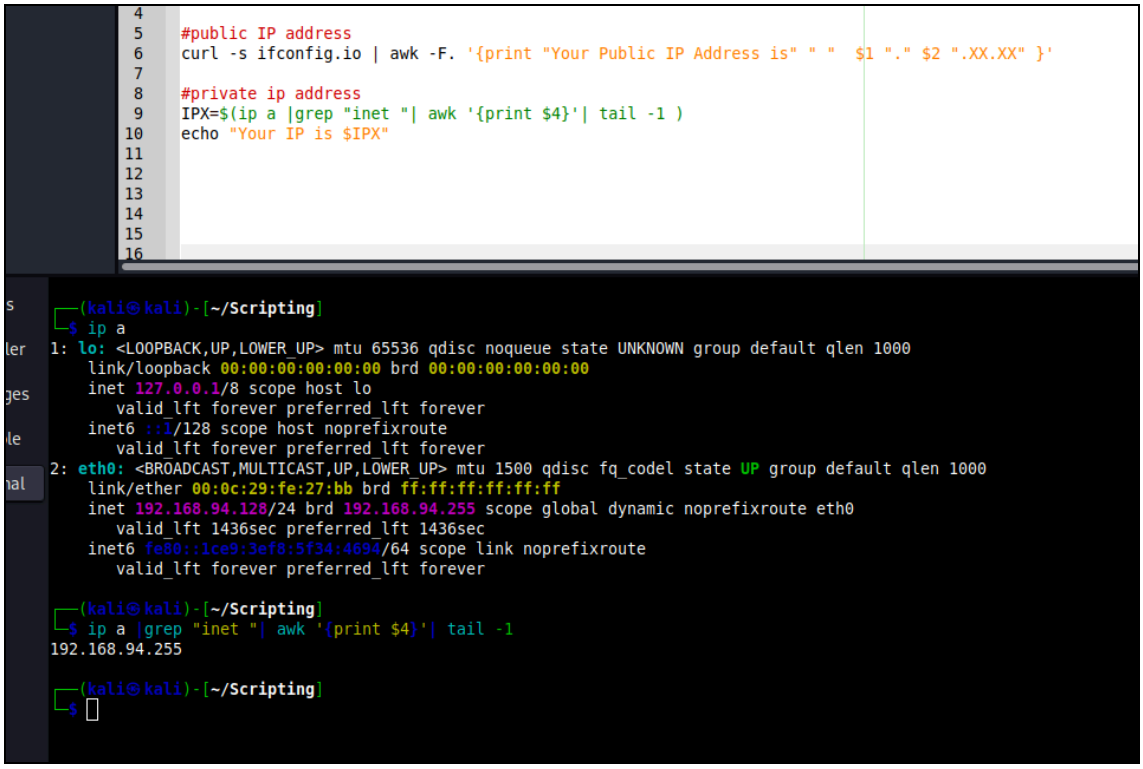
I tested one command at a time in the terminal.

To display the Public IP address, I used awk -F. to make fields using the '.' as a filter as I needed to block out some portions of the IP address.

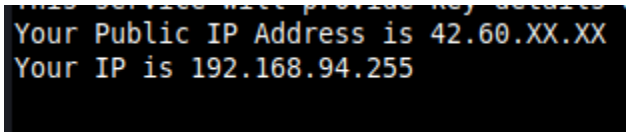
I used the command ip a to get the internal IP address. I saw many lines in the output so I added grep "inet" to only filter out those lines, I added awk '{print $4}' to only print the details in column 4 and lastly added tail -1 to only get the last line with the IP address as there were 2 lines with inet.

After testing out the commands in the terminal, I moved the commands to my script and displayed it in a clean way.

INFO EXTRACTOR



Here is an image of after I had cleaned up the commands and presented it better :



I wanted to add a space between the commands so it looks nicer and less clustered as there are going to be so many parts to this output.

I search online for methods I could use in my script.

INFO EXTRACTOR

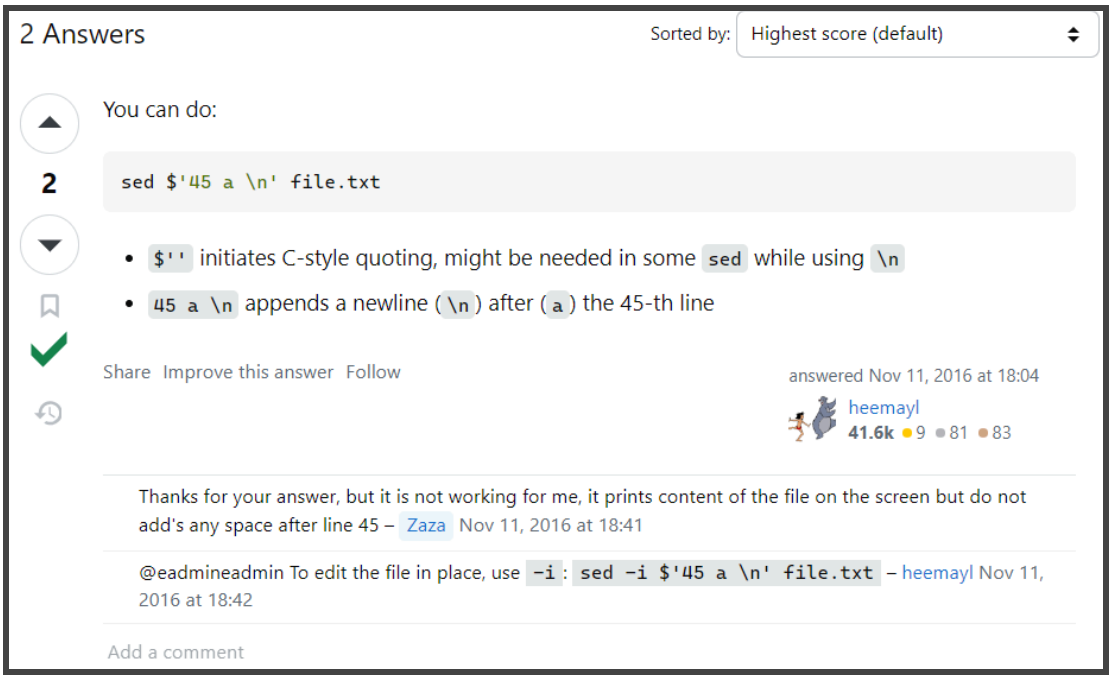
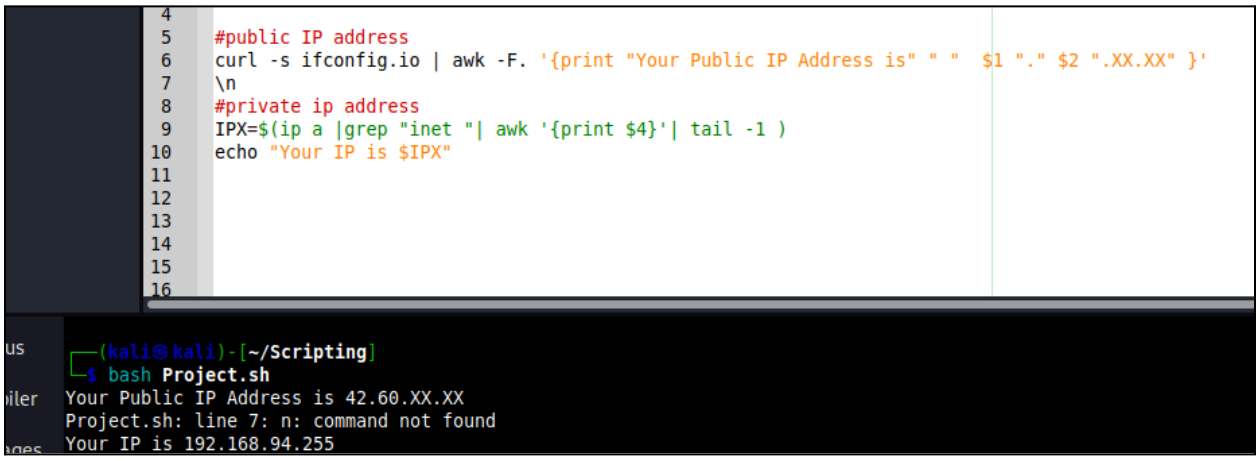


Figure 1: Creating a Blank Line After Some Specific Line Using Bash / Linux (Stack Overflow, 2024)



I tried the method I found online but it did not work.

So I looked for other methods.

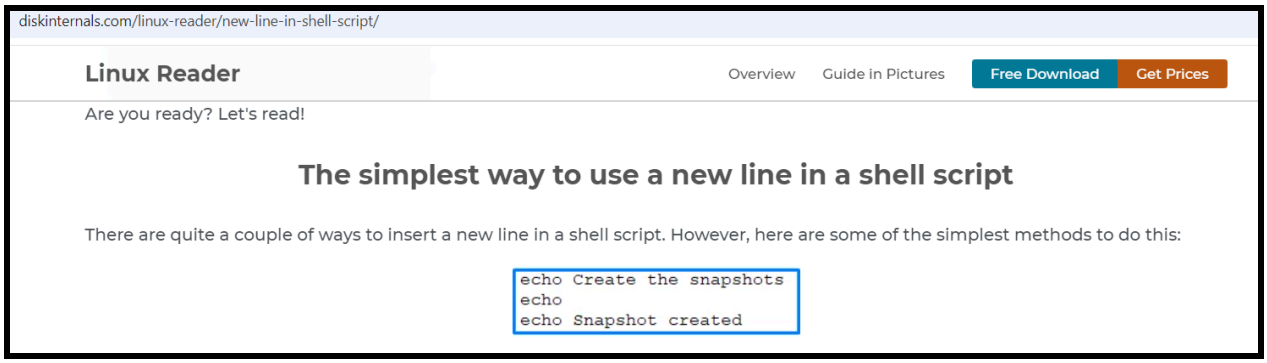
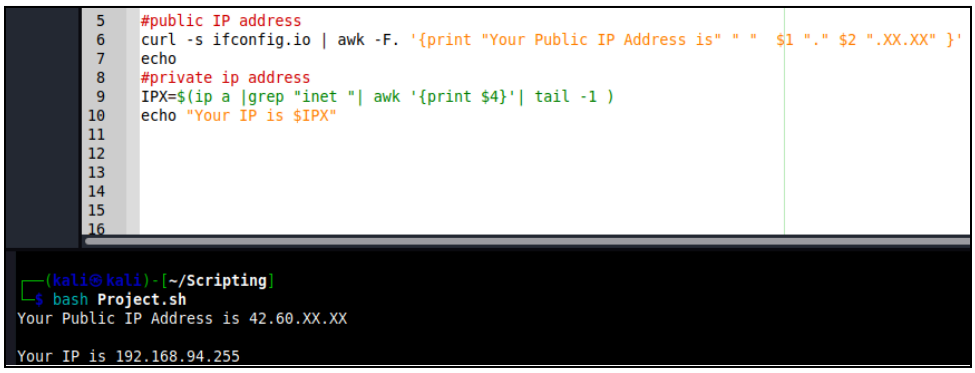


Figure 2 : How to Get a New Line in Shell Script (Disk Internals, 2024)

As I found this method worked in this case I will use it throughout my script.



INFO EXTRACTOR

Next I had to search online how to get the command I want to display the top 5 processes' CPU usage and display it in percentage.

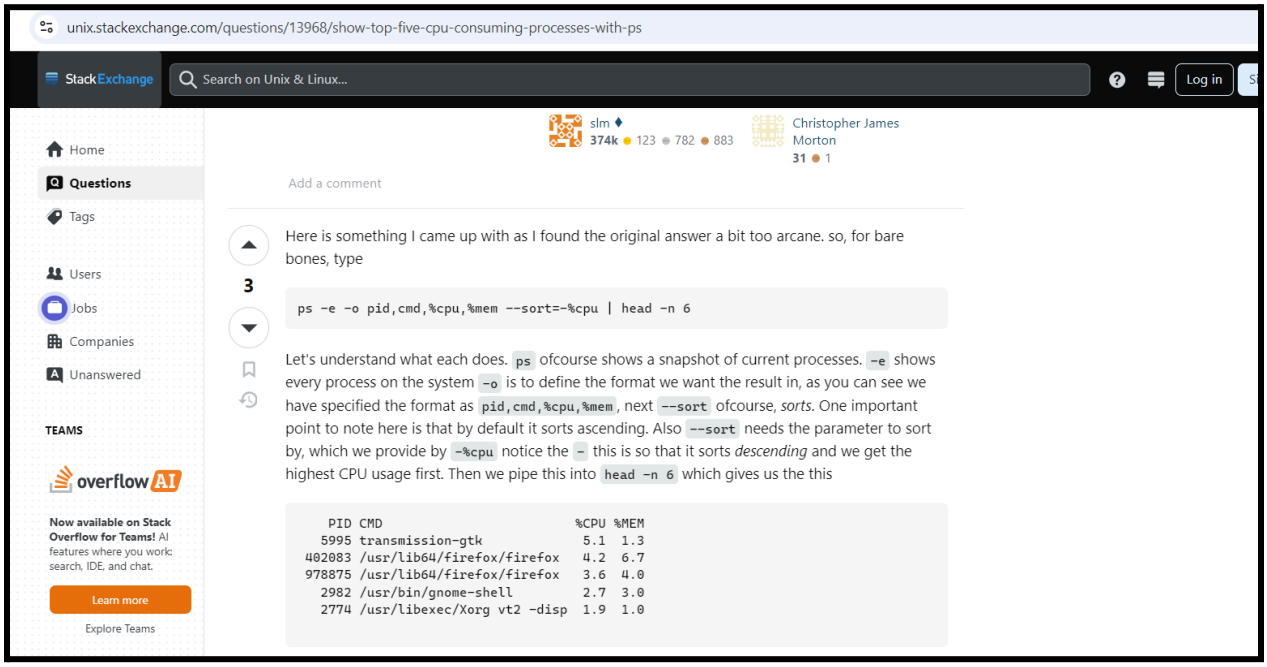


Figure 3: Show Top Five CPU Consuming Processes With `Ps (Stack Exchange, 2024)

I was wondering why the given prompt was head -n 6 when we want only the top 5 processes. I tried out the code with head -n 5 and realised the heading is one of the lines so that's why we need to use head -n 6.

I didn't use %mem as I didn't need the information about the memory here.

I removed pid as the id was not necessary for the portion as well.

```
┌──(kali㉿kali)-[~/Scripting]
└─$ ps -e -o cmd,%cpu --sort=-%cpu | head -n 5
CMD                          %CPU
/usr/lib/xorg/Xorg :0 -seat  3.3
xfwm4 --display :0.0 --sm-c  0.5
geany                        0.4
/usr/lib/x86_64-linux-gnu/x  0.3
```

INFO EXTRACTOR

Now I was looking at how to display the memory usage, free and used.

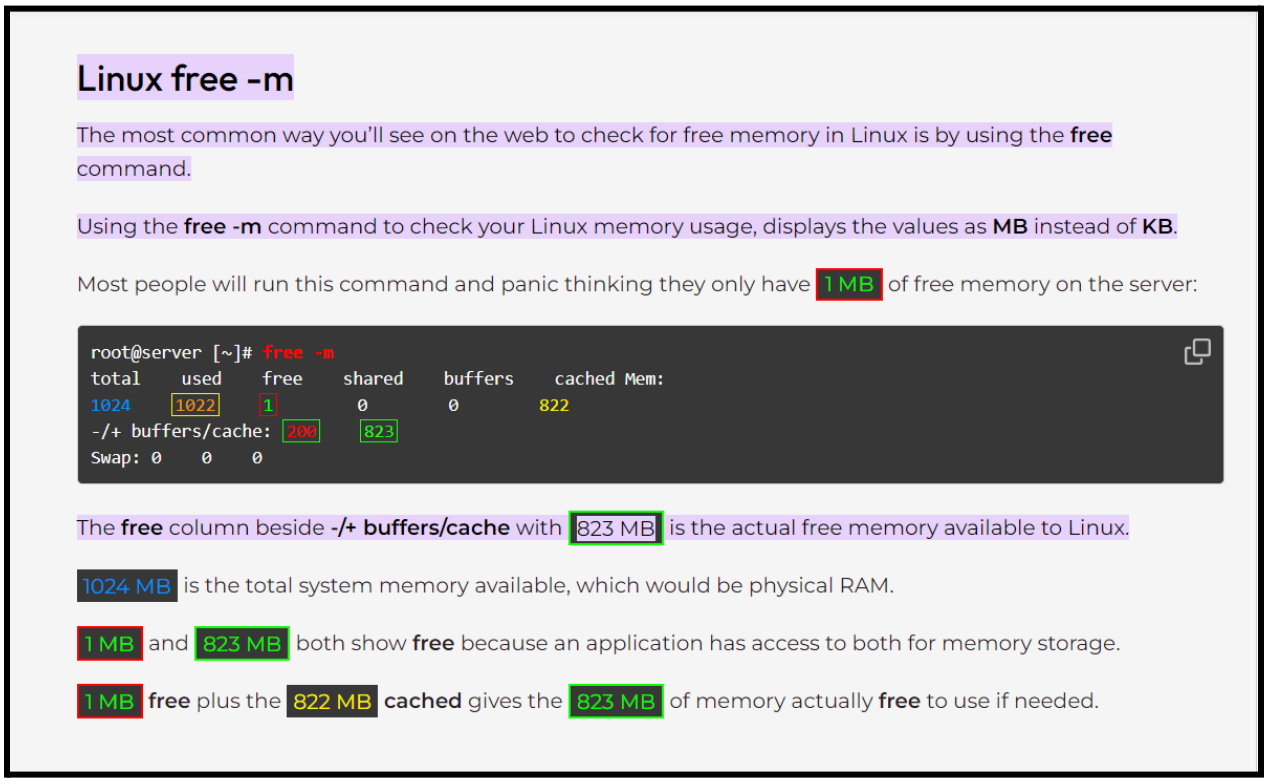


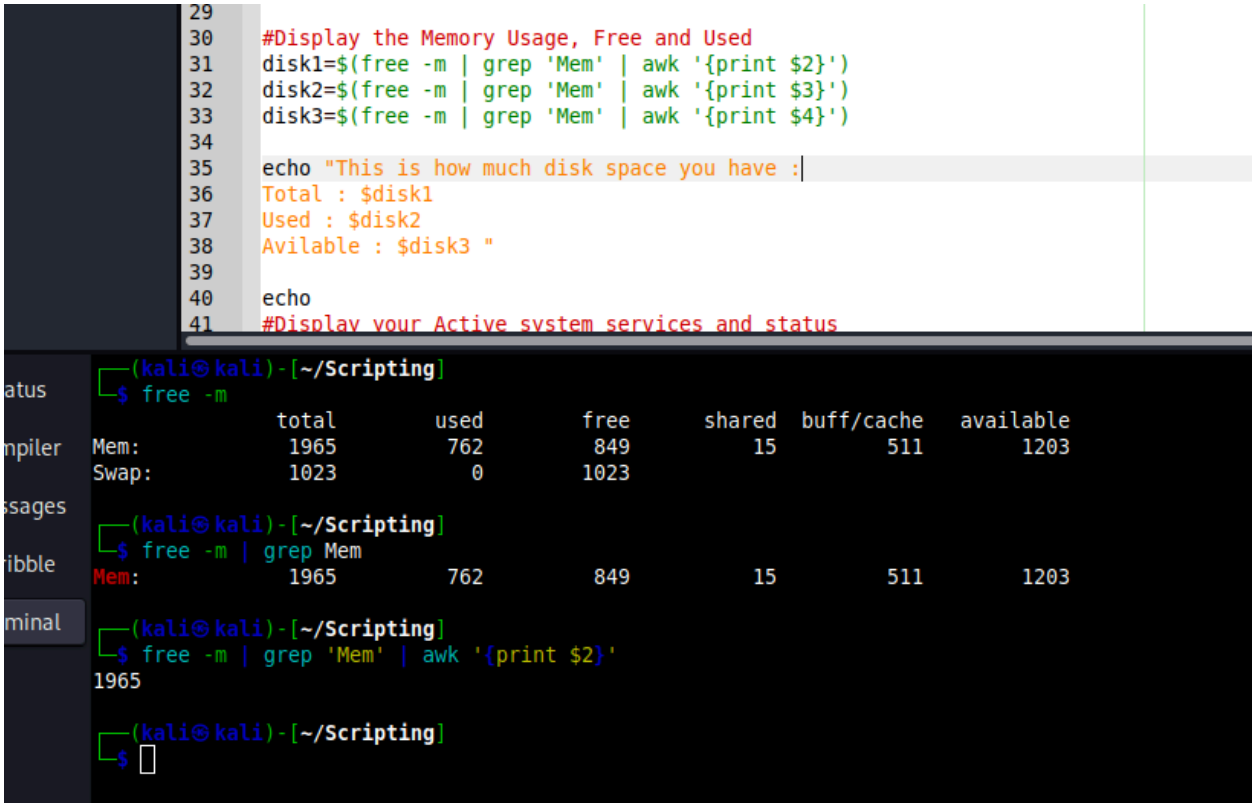
Figure 4 : How to Check the Memory Usage on Linux (InMotion Hosting, 2024)

INFO EXTRACTOR

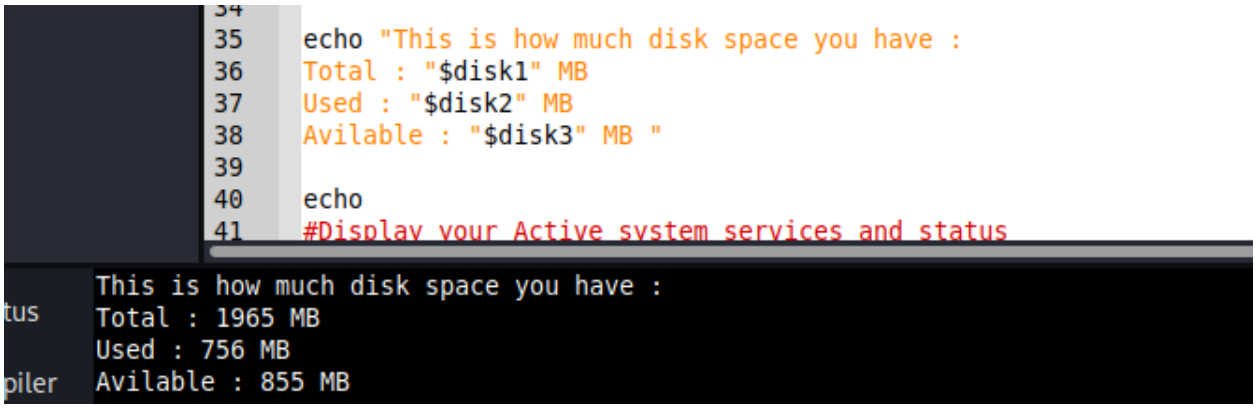
I tested the commands then added it to my script with a different way of presentation.

I only wanted the row under the Mem so I used grep ‘Mem’. I also had to filter out the column I wanted based on the information I wanted to display.

I presented them in 3 rows instead so it is easier for the user to read so I first had to create the variables then displayed it under echo.



I added a “MB” at the end so the user will know the unit the output is showing.



INFO EXTRACTOR

Next I was looking at how to display the active system services and status.

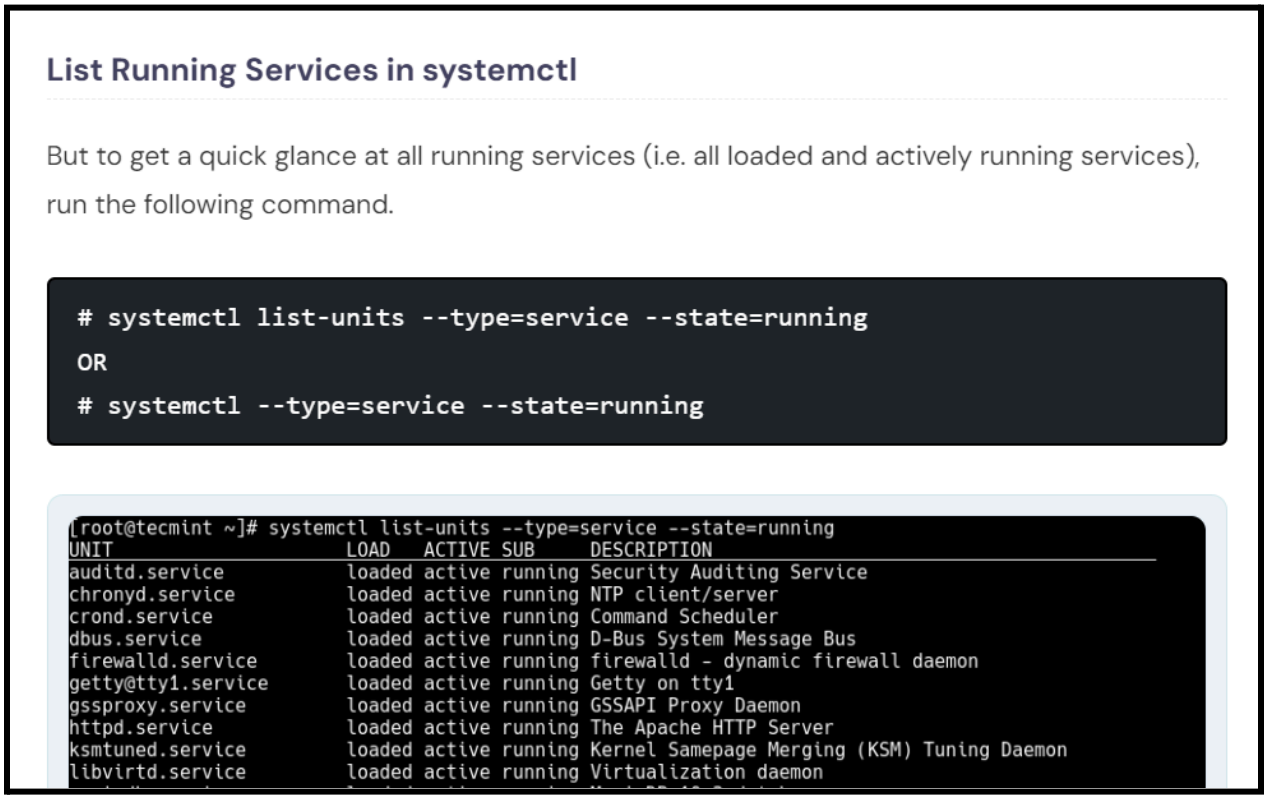
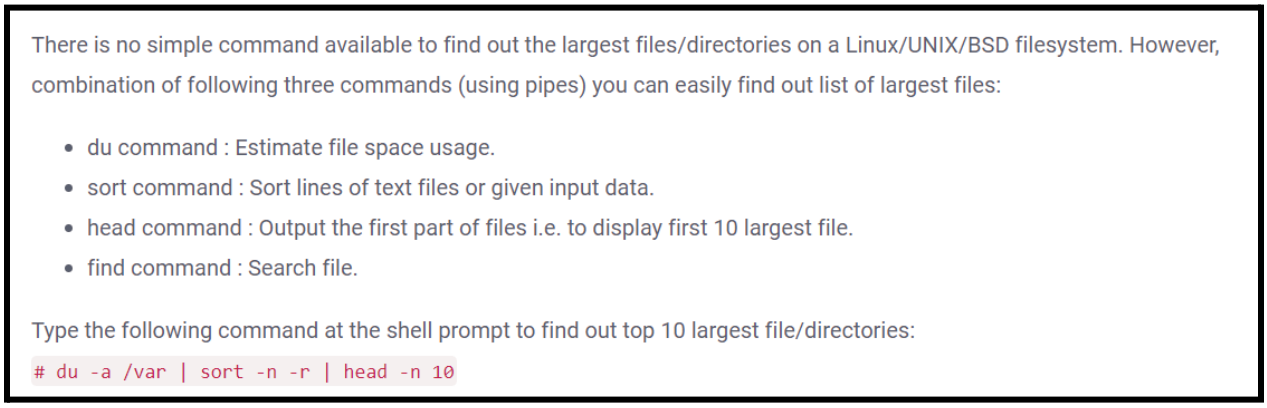


Figure 5: How to List All Running Services Under Systemd in Linux (Tecmint, 2024)

I tested the command and it works and I added it to my script.

INFO EXTRACTOR

Lastly I search how to find the largest files in the home directory and how to present it.



There is no simple command available to find out the largest files/directories on a Linux/UNIX/BSD filesystem. However, combination of following three commands (using pipes) you can easily find out list of largest files:

- du command : Estimate file space usage.
- sort command : Sort lines of text files or given input data.
- head command : Output the first part of files i.e. to display first 10 largest file.
- find command : Search file.

Type the following command at the shell prompt to find out top 10 largest file/directories:

```
# du -a /var | sort -n -r | head -n 10
```

Figure 6: How Do I Find The Largest Top 10 Files (Cloudfanatic, 2024)

I tried the command without `-r` as I was wondering why it was necessary. I realised the output shows the smallest files on top hence we use `-r` in the command. `-r` will reverse the list so the biggest files are shown from the top.

```
└─$ sort --help
Usage: sort [OPTION]... [FILE]...
  or:  sort [OPTION]... --files0-from=F
Write sorted concatenation of all FILE(s) to standard output.

With no FILE, or when FILE is -, read standard input.

Mandatory arguments to long options are mandatory for short options too.
Ordering options:

  -b, --ignore-leading-blanks  ignore leading blanks
  -d, --dictionary-order      consider only blanks and alphanumeric characters
  -f, --ignore-case           fold lower case to upper case characters
  -g, --general-numeric-sort  compare according to general numerical value
  -i, --ignore-nonprinting    consider only printable characters
  -M, --month-sort            compare (unknown) < 'JAN' < ... < 'DEC'
  -h, --human-numeric-sort    compare human readable numbers (e.g., 2K 1G)
  -n, --numeric-sort          compare according to string numerical value
  -R, --random-sort           shuffle, but group identical keys.  See shuf(1)
      --random-source=FILE    get random bytes from FILE
  -r, --reverse               reverse the result of comparisons
      --sort=WORD             sort according to WORD:
                                general-numeric -g, human-numeric -h, month -M,
                                numeric -n, random -R, version -V
  -V, --version-sort          natural sort of (version) numbers within text
```

I read further down the article and saw I could add more to the command to display a human readable form.

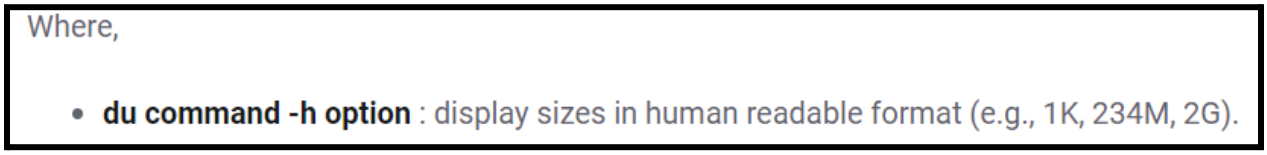
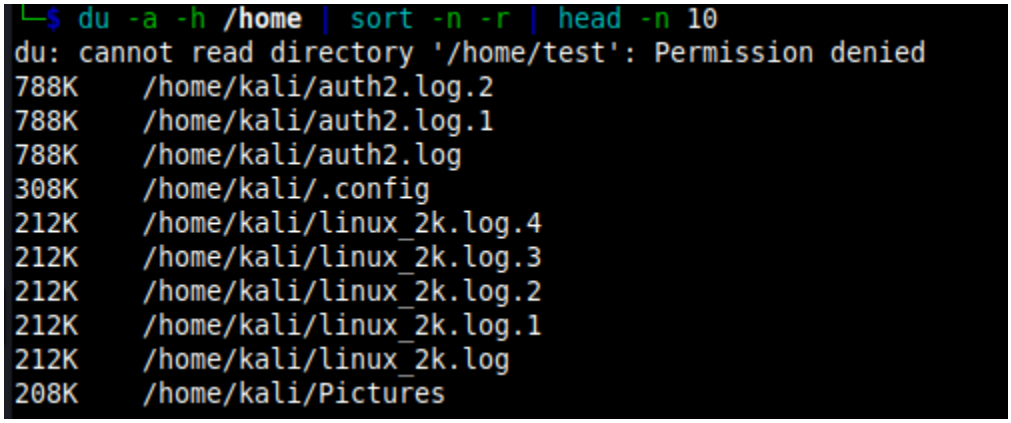


Figure 7: How Do I Find The Largest Top 10 Files (Cloudfanatic, 2024)

So I added this to my script.



I added some added information at the start and end of the prompt and references in all prompts.

Conclusion

In Conclusion, this project helps me to put into practice what I have learnt so far and extends my learning out of the classroom as I had to look for solutions online from other trusted sources. This was a great exercise as in a real life scenario I would have to find other sources for information instead of relying on the little information I have.

Recommendations

The user has no input in what commands were used in the script. As some machines may not have all protocols installed, maybe the option could be extended to the user. However this would make it more interactive instead of an automated script.

References

Stack Overflow *"Creating a Blank Line After Some Specific Line Using Bash / Linux."*, stackoverflow.com/questions/40553714/creating-a-blank-line-after-some-specific-line-using-bash-linux.

DiskInternals. *How to Get a New Line in Shell Script*. 29 July 2021, www.diskinternals.com/linux-reader/new-line-in-shell-script.

"Show Top Five CPU Consuming Processes With `Ps`." *Unix & Linux Stack Exchange*, unix.stackexchange.com/questions/13968/show-top-five-cpu-consuming-processes-with-ps.

Contributor, InMotion Hosting. "How to Check the Memory Usage on Linux." *InMotion Hosting Support Center*, 7 Dec. 2023, www.inmotionhosting.com/support/server/linux/check-memory-usage/#:~:text=%2Fproc%2Fmeminfo-,Linux%20free%20%2Dm,as%20MB%20instead%20of%20KB.&text=The%20free%20column%20beside%20%2D%2F%2B,free%20memory%20available%20to%20Linux.

Kili, Aaron, and Aaron Kili. "How to List All Running Services Under Systemd in Linux." *How to List All Running Services Under Systemd in Linux*, 16 Nov. 2023, www.tecmint.com/list-all-running-services-under-systemd-in-linux.

How Do I Find the Largest Top 10 Files and Directories on a Linux / UNIX / BSD? - Cloudfanatic. my.cloudfanatic.net/index.php/knowledgebase/5/How-Do-I-Find-The-Largest-Top-10-Files-and-Directories-On-a-Linux-or-UNIX-or-BSD.html.