

# Conducting Forensic Investigations on Linux Systems (4e)

Digital Forensics, Investigation, and Response, Fourth Edition - Lab 06

Student:

Jose Cruz

Email:

jose.cruz2@udc.edu

Time on Task:

9 hours, 20 minutes

Progress:

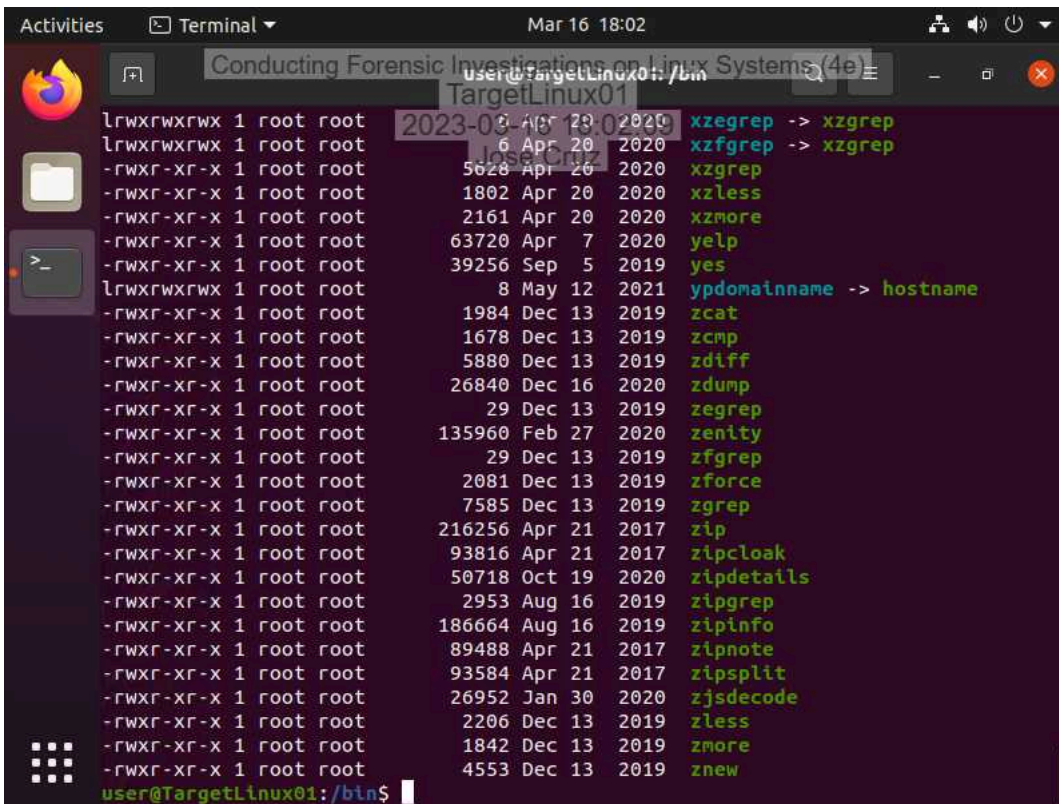
100%

Report Generated: Tuesday, April 4, 2023 at 11:42 PM

## Section 1: Hands-On Demonstration

### Part 1: Explore a Live Linux System

17. Make a screen capture showing the contents of the `/bin` directory.

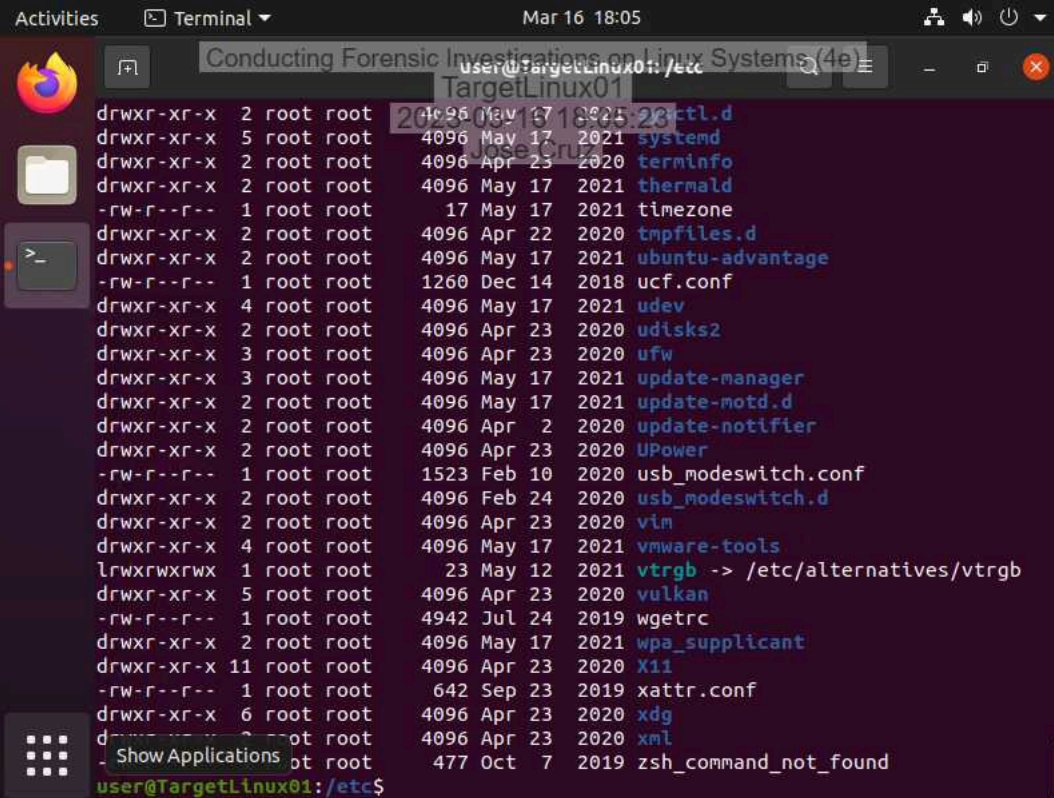


```
user@TargetLinux01: /bin$ ls -l
lrwxrwxrwx 1 root root 21 Apr 23 2020 xzgrep -> xzgrep
lrwxrwxrwx 1 root root 21 Apr 20 2020 xzfgrep -> xzfgrep
-rwxr-xr-x 1 root root 5628 Apr 26 2020 xzgrep
-rwxr-xr-x 1 root root 1802 Apr 20 2020 xzless
-rwxr-xr-x 1 root root 2161 Apr 20 2020 xzmore
-rwxr-xr-x 1 root root 63720 Apr 7 2020 yelp
-rwxr-xr-x 1 root root 39256 Sep 5 2019 yes
lrwxrwxrwx 1 root root 8 May 12 2021 ypdomainname -> hostname
-rwxr-xr-x 1 root root 1984 Dec 13 2019 zcat
-rwxr-xr-x 1 root root 1678 Dec 13 2019 zcmp
-rwxr-xr-x 1 root root 5880 Dec 13 2019 zdiff
-rwxr-xr-x 1 root root 26840 Dec 16 2020 zdump
-rwxr-xr-x 1 root root 29 Dec 13 2019 zegrep
-rwxr-xr-x 1 root root 135960 Feb 27 2020 zenity
-rwxr-xr-x 1 root root 29 Dec 13 2019 zfgrep
-rwxr-xr-x 1 root root 2081 Dec 13 2019 zforce
-rwxr-xr-x 1 root root 7585 Dec 13 2019 zgrep
-rwxr-xr-x 1 root root 216256 Apr 21 2017 zip
-rwxr-xr-x 1 root root 93816 Apr 21 2017 zipcloak
-rwxr-xr-x 1 root root 50718 Oct 19 2020 zipdetails
-rwxr-xr-x 1 root root 2953 Aug 16 2019 zipgrep
-rwxr-xr-x 1 root root 186664 Aug 16 2019 zipinfo
-rwxr-xr-x 1 root root 89488 Apr 21 2017 zipnote
-rwxr-xr-x 1 root root 93584 Apr 21 2017 zipsplit
-rwxr-xr-x 1 root root 26952 Jan 30 2020 zjsdecode
-rwxr-xr-x 1 root root 2206 Dec 13 2019 zless
-rwxr-xr-x 1 root root 1842 Dec 13 2019 zmore
-rwxr-xr-x 1 root root 4553 Dec 13 2019 znew
user@TargetLinux01: /bin$
```

## Conducting Forensic Investigations on Linux Systems (4e)

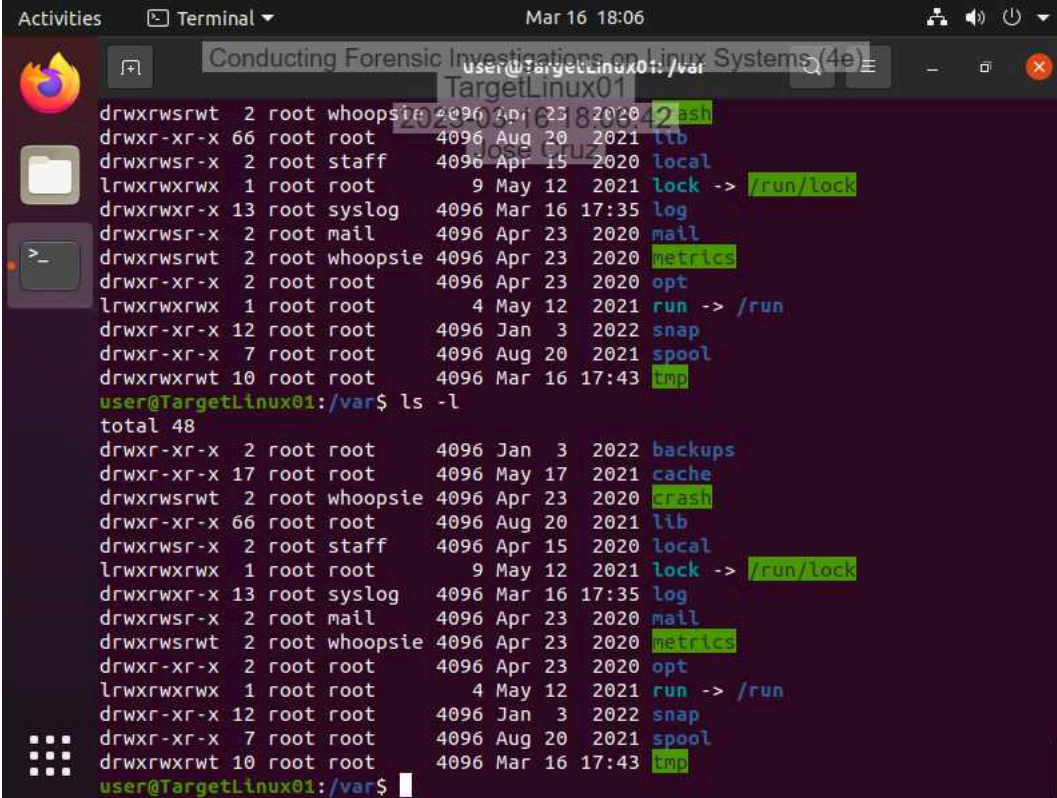
Digital Forensics, Investigation, and Response, Fourth Edition - Lab 06

20. Make a screen capture showing the contents of the `/etc` directory.



```
Activities Terminal Mar 16 18:05
Conducting Forensic Investigations on Linux Systems (4e)
user@TargetLinux01: /etc
TargetLinux01
2023-03-13 18:05:28
user@TargetLinux01: /etc$ ls -l
drwxr-xr-x 2 root root 4096 May 17 2021 .
drwxr-xr-x 5 root root 4096 May 17 2021 ..
drwxr-xr-x 2 root root 4096 Apr 23 2020 .alternatives
drwxr-xr-x 2 root root 4096 May 17 2021 .apt
-rw-r--r-- 1 root root 17 May 17 2021 .time
drwxr-xr-x 2 root root 4096 Apr 22 2020 .tmpfiles.d
drwxr-xr-x 2 root root 4096 May 17 2021 ubuntu-advantage
-rw-r--r-- 1 root root 1260 Dec 14 2018 ucf.conf
drwxr-xr-x 4 root root 4096 May 17 2021 udev
drwxr-xr-x 2 root root 4096 Apr 23 2020 udisks2
drwxr-xr-x 3 root root 4096 Apr 23 2020 ufw
drwxr-xr-x 3 root root 4096 May 17 2021 update-manager
drwxr-xr-x 2 root root 4096 May 17 2021 update-motd.d
drwxr-xr-x 2 root root 4096 Apr 2 2020 update-notifier
drwxr-xr-x 2 root root 4096 Apr 23 2020 UPower
-rw-r--r-- 1 root root 1523 Feb 10 2020 usb_modeswitch.conf
drwxr-xr-x 2 root root 4096 Feb 24 2020 usb_modeswitch.d
drwxr-xr-x 2 root root 4096 Apr 23 2020 vim
drwxr-xr-x 4 root root 4096 May 17 2021 vmware-tools
lrwxrwxrwx 1 root root 23 May 12 2021 vtrgb -> /etc/alternatives/vtrgb
drwxr-xr-x 5 root root 4096 Apr 23 2020 vulkan
-rw-r--r-- 1 root root 4942 Jul 24 2019 wgetrc
drwxr-xr-x 2 root root 4096 May 17 2021 wpa_supplicant
drwxr-xr-x 11 root root 4096 Apr 23 2020 X11
-rw-r--r-- 1 root root 642 Sep 23 2019 xattr.conf
drwxr-xr-x 6 root root 4096 Apr 23 2020 xdg
drwxr-xr-x 2 root root 4096 Apr 23 2020 xml
-rw-r--r-- 1 root root 477 Oct 7 2019 zsh_command_not_found
user@TargetLinux01: /etc$
```

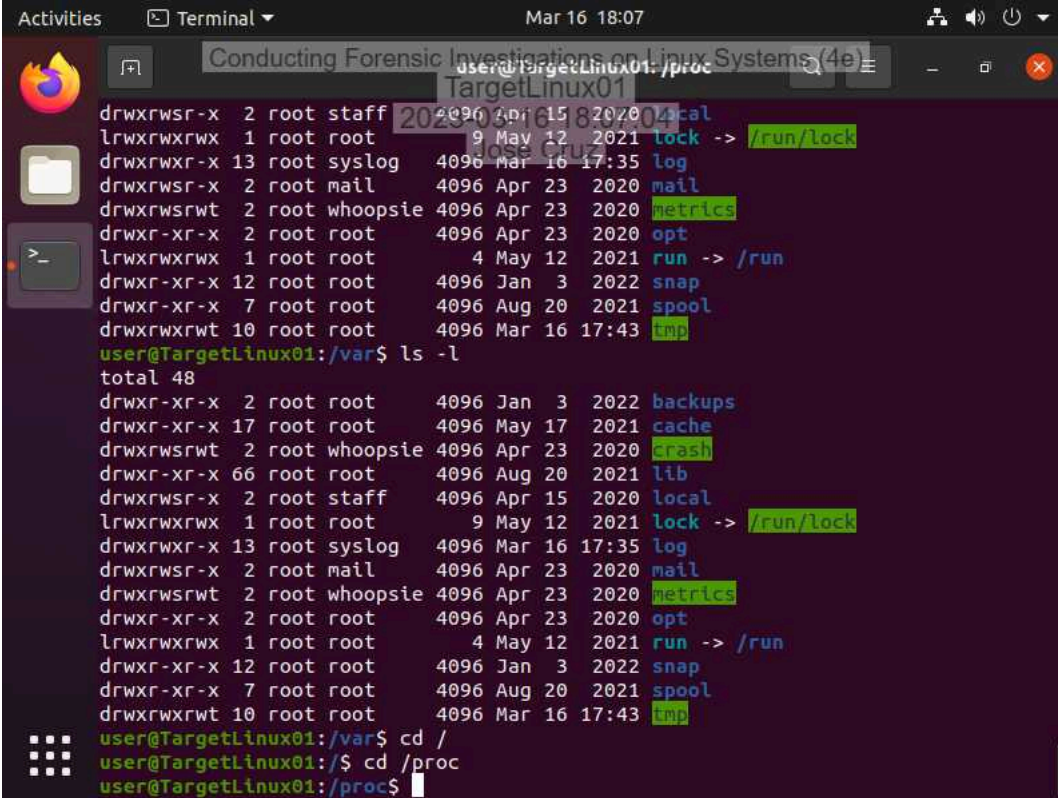
21. Make a screen capture showing the contents of the /var directory.



A terminal window titled "Terminal" with a timestamp of "Mar 16 18:06". The window shows the command `ls -l` being executed in the `/var` directory. The output lists various system directories and files with their permissions, ownership, size, and modification dates. The directories listed include `backups`, `cache`, `crash`, `lib`, `local`, `lock`, `log`, `mail`, `metrics`, `opt`, `run`, `snap`, `spool`, and `tmp`. The `lock` directory is highlighted with a green box, and the `run` directory is highlighted with a green box. The terminal window also shows the command `ls -l` being executed in the `/var` directory.

```
user@TargetLinux01: /var$ ls -l
total 48
drwxr-xr-x  2 root root    4096 Jan  3  2022 backups
drwxr-xr-x 17 root root    4096 May 17  2021 cache
drwxrwsrwt  2 root whoopsie 4096 Apr 23  2020 crash
drwxr-xr-x 66 root root    4096 Aug 20  2021 lib
drwxrwsr-x  2 root staff   4096 Apr 15  2020 local
lrwxrwxrwx  1 root root      9 May 12  2021 lock -> /run/lock
drwxrwxr-x 13 root syslog  4096 Mar 16 17:35 log
drwxrwsr-x  2 root mail    4096 Apr 23  2020 mail
drwxrwsrwt  2 root whoopsie 4096 Apr 23  2020 metrics
drwxr-xr-x  2 root root    4096 Apr 23  2020 opt
lrwxrwxrwx  1 root root      4 May 12  2021 run -> /run
drwxr-xr-x 12 root root    4096 Jan  3  2022 snap
drwxr-xr-x  7 root root    4096 Aug 20  2021 spool
drwxrwsrwt 10 root root    4096 Mar 16 17:43 tmp
```

22. Make a screen capture showing the contents of the /proc directory.



The screenshot shows a terminal window titled "Terminal" with the date "Mar 16 18:07". The user is logged in as "user@TargetLinux01". The terminal displays the output of the command `ls -l` in the `/var` directory, followed by the command `cd /` and `cd /proc`. The output of `ls -l` shows a list of files and directories in `/var` with their permissions, owner, group, size, and modification date. The files and directories listed are: `backups`, `cache`, `crash`, `lib`, `local`, `lock -> /run/lock`, `log`, `mail`, `metrics`, `opt`, `run -> /run`, `snap`, `spool`, and `tmp`. The terminal window also shows the user's prompt `user@TargetLinux01:/proc$`.

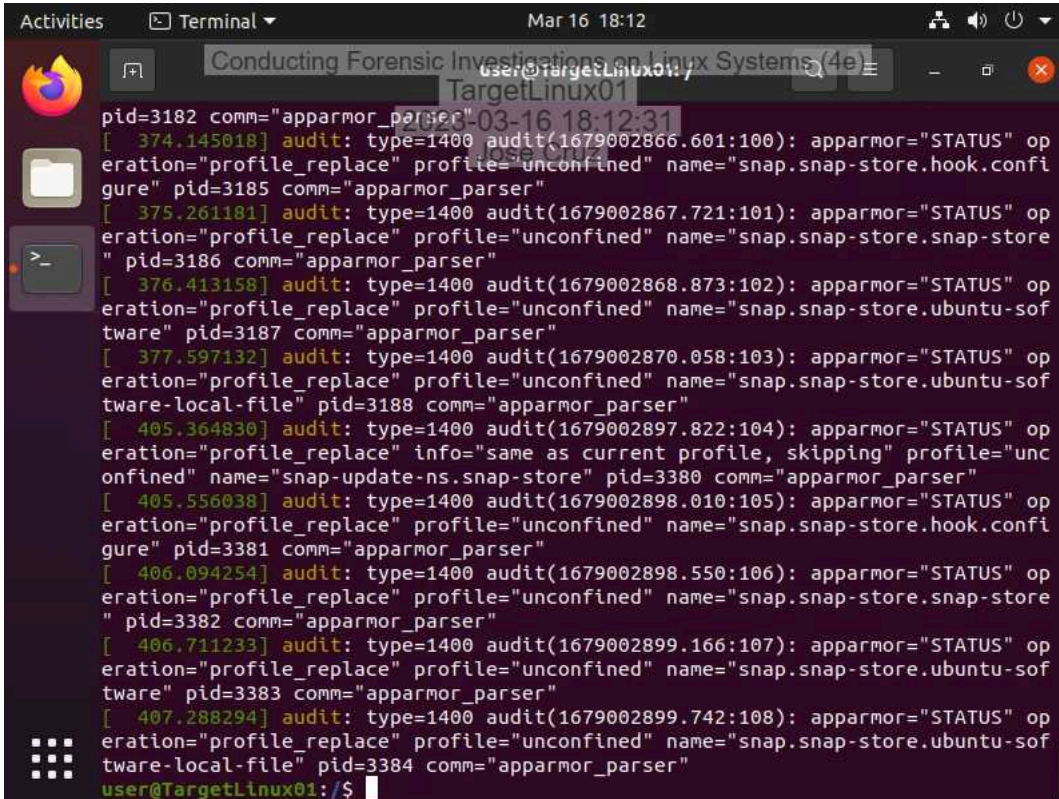
```
user@TargetLinux01:/proc$ ls -l
total 48
drwxr-xr-x  2 root root      4096 Jan  3  2022 backups
drwxr-xr-x 17 root root      4096 May 17  2021 cache
drwxrwsrwt  2 root whoopsie 4096 Apr 23  2020 crash
drwxr-xr-x 66 root root      4096 Aug 20  2021 lib
drwxrwsr-x  2 root staff    4096 Apr 15  2020 local
lrwxrwxrwx  1 root root         9 May 12  2021 lock -> /run/lock
drwxrwsr-x 13 root syslog   4096 Mar 16 17:35 log
drwxrwsr-x  2 root mail     4096 Apr 23  2020 mail
drwxrwsrwt  2 root whoopsie 4096 Apr 23  2020 metrics
drwxr-xr-x  2 root root      4096 Apr 23  2020 opt
lrwxrwxrwx  1 root root         4 May 12  2021 run -> /run
drwxr-xr-x 12 root root      4096 Jan  3  2022 snap
drwxr-xr-x  7 root root      4096 Aug 20  2021 spool
drwxrwsrwt 10 root root      4096 Mar 16 17:43 tmp

user@TargetLinux01:/var$ cd /
user@TargetLinux01:/ $ cd /proc
user@TargetLinux01:/proc$
```

## Part 2: Use Linux Shell Commands for Forensic Investigations

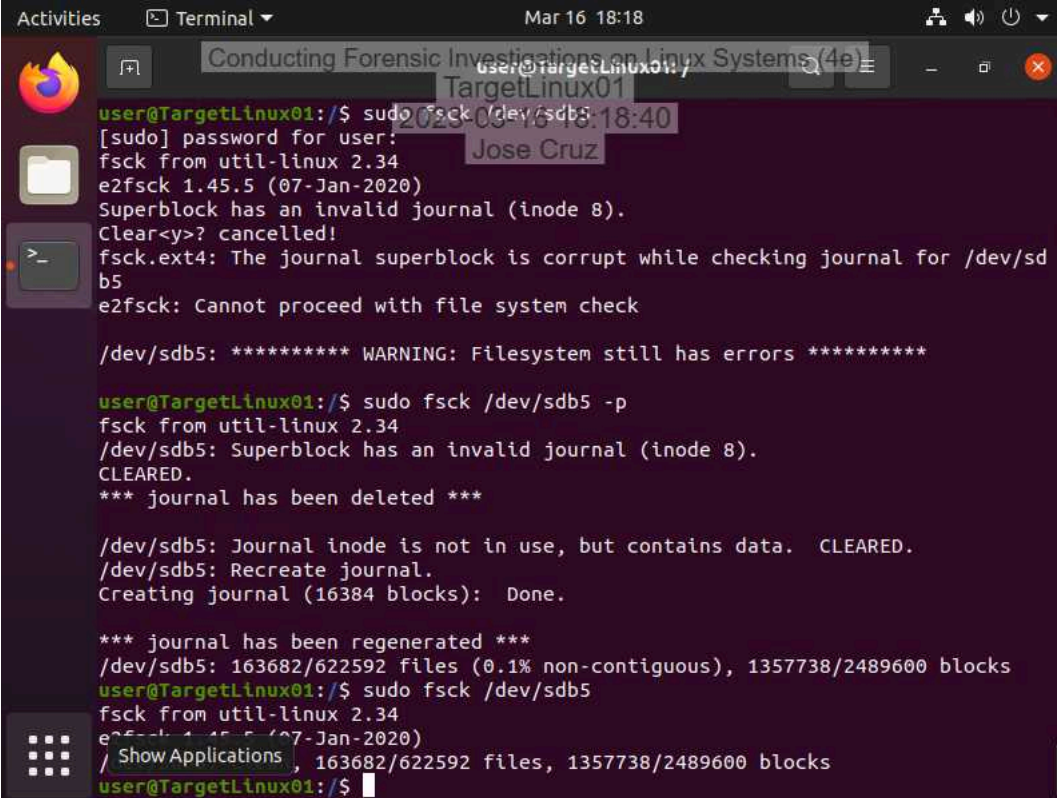


2. Make a screen capture showing the results of the `dmesg` command.

A screenshot of a Linux terminal window. The window title is "Conducting Forensic Investigations on Linux Systems (4e)" and the terminal prompt is "user@TargetLinux01:". The terminal output shows the results of the `dmesg` command, displaying a series of audit messages from the `apparmor_parser` process. The messages include timestamps, audit IDs, and details about profile replacements for various snap packages like `hook.config`, `ubuntu-software-local-file`, and `ubuntu-software`. The terminal window has a dark background with light-colored text. The top of the window shows the system clock as "Mar 16 18:12".

```
pid=3182 comm="apparmor_parser" [ 374.145018] audit: type=1400 audit(1679002866.601:100): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.hook.config" pid=3185 comm="apparmor_parser" [ 375.261181] audit: type=1400 audit(1679002867.721:101): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.snap-store" pid=3186 comm="apparmor_parser" [ 376.413158] audit: type=1400 audit(1679002868.873:102): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=3187 comm="apparmor_parser" [ 377.597132] audit: type=1400 audit(1679002870.058:103): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=3188 comm="apparmor_parser" [ 405.364830] audit: type=1400 audit(1679002897.822:104): apparmor="STATUS" operation="profile_replace" info="same as current profile, skipping" profile="unconfined" name="snap-update-ns.snap-store" pid=3380 comm="apparmor_parser" [ 405.556038] audit: type=1400 audit(1679002898.010:105): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.hook.config" pid=3381 comm="apparmor_parser" [ 406.094254] audit: type=1400 audit(1679002898.550:106): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.snap-store" pid=3382 comm="apparmor_parser" [ 406.711233] audit: type=1400 audit(1679002899.166:107): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=3383 comm="apparmor_parser" [ 407.288294] audit: type=1400 audit(1679002899.742:108): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=3384 comm="apparmor_parser" user@TargetLinux01:/$
```

### 7. Make a screen capture showing the results of the fsck command.



A terminal window titled "Terminal" with a search bar and window controls. The terminal shows the execution of the fsck command on /dev/sdb5. The output indicates a corrupted journal superblock and a warning that the filesystem still has errors. The user then runs fsck with the -p flag, which successfully clears the errors and regenerates the journal. The final output shows the filesystem is now clean with 163682/622592 files and 1357738/2489600 blocks used.

```
user@TargetLinux01:/$ sudo fsck /dev/sdb5
[sudo] password for user:
fsck from util-linux 2.34
e2fsck 1.45.5 (07-Jan-2020)
Superblock has an invalid journal (inode 8).
Clear<y>? cancelled!
fsck.ext4: The journal superblock is corrupt while checking journal for /dev/sd
b5
e2fsck: Cannot proceed with file system check

/dev/sdb5: ***** WARNING: Filesystem still has errors *****

user@TargetLinux01:/$ sudo fsck /dev/sdb5 -p
fsck from util-linux 2.34
/dev/sdb5: Superblock has an invalid journal (inode 8).
CLEARED.
*** journal has been deleted ***

/dev/sdb5: Journal inode is not in use, but contains data.  CLEARED.
/dev/sdb5: Recreate journal.
Creating journal (16384 blocks):  Done.

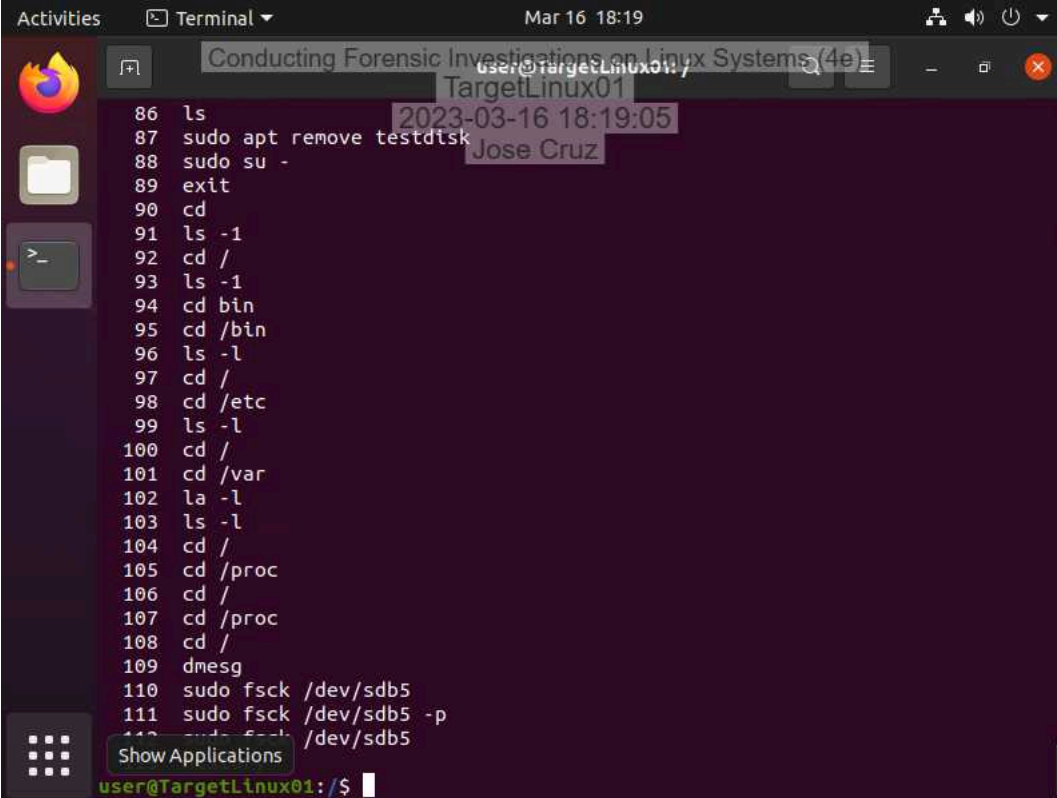
*** journal has been regenerated ***
/dev/sdb5: 163682/622592 files (0.1% non-contiguous), 1357738/2489600 blocks
user@TargetLinux01:/$ sudo fsck /dev/sdb5
fsck from util-linux 2.34
e2fsck 1.45.5 (07-Jan-2020)
/ Show Applications , 163682/622592 files, 1357738/2489600 blocks
user@TargetLinux01:/$
```

## Conducting Forensic Investigations on Linux Systems (4e)

Digital Forensics, Investigation, and Response, Fourth Edition - Lab 06

---

9. Make a screen capture showing the results of the history command.

A screenshot of a Linux terminal window. The window title is "Terminal" and the date/time is "Mar 16 18:19". The terminal shows a list of commands and their outputs, numbered 86 to 112. The commands include 'ls', 'sudo apt remove testdisk', 'sudo su -', 'exit', 'cd', 'ls -l', 'cd /', 'cd bin', 'cd /bin', 'cd /etc', 'cd /var', 'cd /proc', 'dmesg', and 'sudo fsck /dev/sdb5'. The output for 'ls' shows the contents of the root directory. The output for 'dmesg' shows the boot log. The prompt at the bottom is "user@TargetLinux01:/\$".

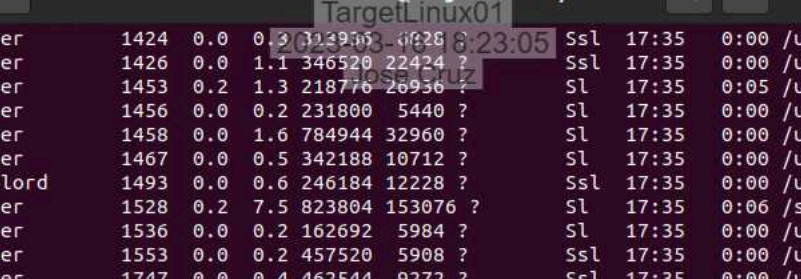
```
86 ls
87 sudo apt remove testdisk
88 sudo su -
89 exit
90 cd
91 ls -l
92 cd /
93 ls -l
94 cd bin
95 cd /bin
96 ls -l
97 cd /
98 cd /etc
99 ls -l
100 cd /
101 cd /var
102 ls -l
103 cd /
104 cd /proc
105 cd /
106 cd /proc
107 cd /
108 dmesg
109 sudo fsck /dev/sdb5
110 sudo fsck /dev/sdb5 -p
111 sudo fsck /dev/sdb5
112
```

user@TargetLinux01:/\$

## Digital Forensics, Investigation, and Response, Fourth Edition - Lab 06

Digital Forensics, Investigation, and Response, Fourth Edition - Lab 06

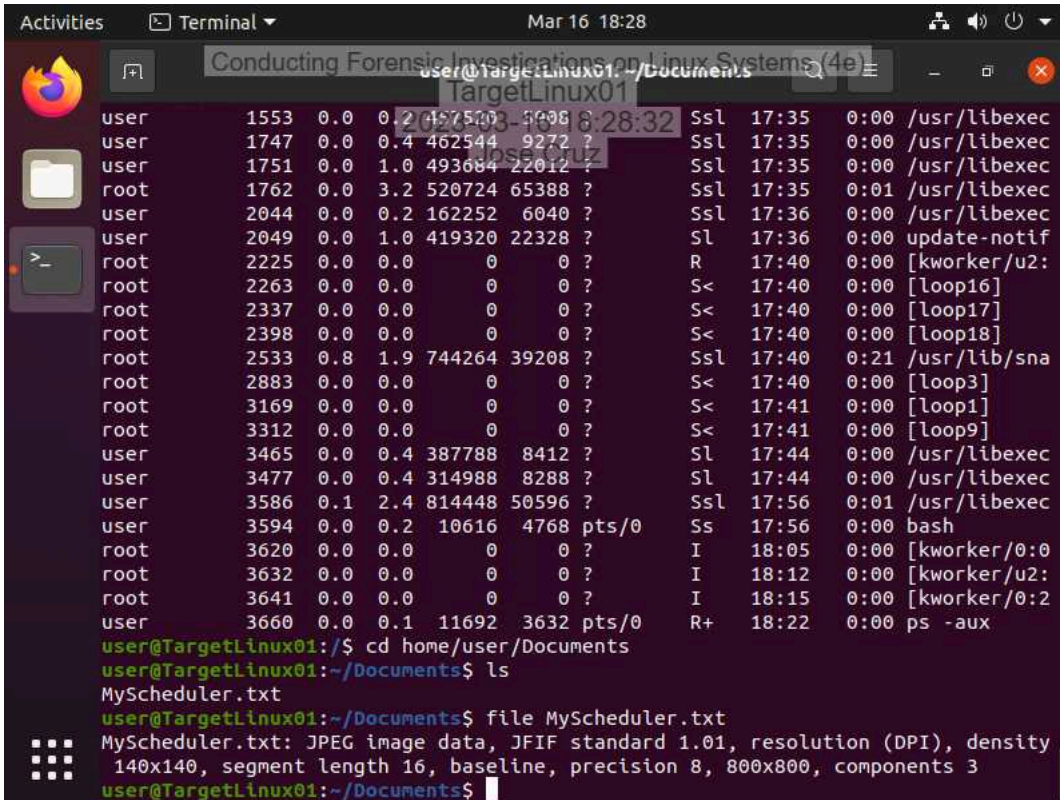
11. **Make a screen capture** showing the running processes.



```
Activities Terminal Mar 16 18:23
Conducting Forensic Investigations on Linux Systems (4e)
TargetLinux01
user@targetlinux01:~$ netstat -tlnp
tcp        0 0 0.0.0.0:22 0.0.0.0:22 LISTENING 6:23:05 Ssl 17:35 0:00 /usr/libexec
user        1426 0.0 1.1 346520 22424 ? Ssl 17:35 0:00 /usr/libexec
user        1453 0.2 1.3 218776 26936 ? SL 17:35 0:05 /usr/bin/vmt
user        1456 0.0 0.2 231800 5440 ? SL 17:35 0:00 /usr/libexec
user        1458 0.0 1.6 784944 32960 ? SL 17:35 0:00 /usr/libexec
user        1467 0.0 0.5 342188 10712 ? SL 17:35 0:00 /usr/libexec
colord      1493 0.0 0.6 246184 12228 ? Ssl 17:35 0:00 /usr/libexec
user        1528 0.2 7.5 823804 153076 ? SL 17:35 0:06 /snap/snap-s
user        1536 0.0 0.2 162692 5984 ? SL 17:35 0:00 /usr/libexec
user        1553 0.0 0.2 457520 5908 ? Ssl 17:35 0:00 /usr/libexec
user        1747 0.0 0.4 462544 9272 ? Ssl 17:35 0:00 /usr/libexec
user        1751 0.0 1.0 493684 22012 ? Ssl 17:35 0:00 /usr/libexec
root        1762 0.0 3.2 520724 65388 ? Ssl 17:35 0:01 /usr/libexec
user        2044 0.0 0.2 162252 6040 ? Ssl 17:36 0:00 /usr/libexec
user        2049 0.0 1.0 419320 22328 ? SL 17:36 0:00 update-notif
root        2225 0.0 0.0 0 0 ? R 17:40 0:00 [kworker/u2:
root        2263 0.0 0.0 0 0 ? S< 17:40 0:00 [loop16]
root        2337 0.0 0.0 0 0 ? S< 17:40 0:00 [loop17]
root        2398 0.0 0.0 0 0 ? S< 17:40 0:00 [loop18]
root        2533 0.8 1.9 744264 39208 ? Ssl 17:40 0:21 /usr/lib/sna
root        2883 0.0 0.0 0 0 ? S< 17:40 0:00 [loop3]
root        3169 0.0 0.0 0 0 ? S< 17:41 0:00 [loop1]
root        3312 0.0 0.0 0 0 ? S< 17:41 0:00 [loop9]
user        3465 0.0 0.4 387788 8412 ? SL 17:44 0:00 /usr/libexec
user        3477 0.0 0.4 314988 8288 ? SL 17:44 0:00 /usr/libexec
user        3586 0.1 2.4 814448 50596 ? Ssl 17:56 0:01 /usr/libexec
user        3591 0.0 0.2 10616 4768 pts/0 Ss 17:56 0:00 bash
root        3632 0.0 0.0 0 0 ? I 18:05 0:00 [kworker/0:0
root        3632 0.0 0.0 0 0 ? I 18:12 0:00 [kworker/u2:
Show Applications
```



15. Make a screen capture showing the results of the file command.



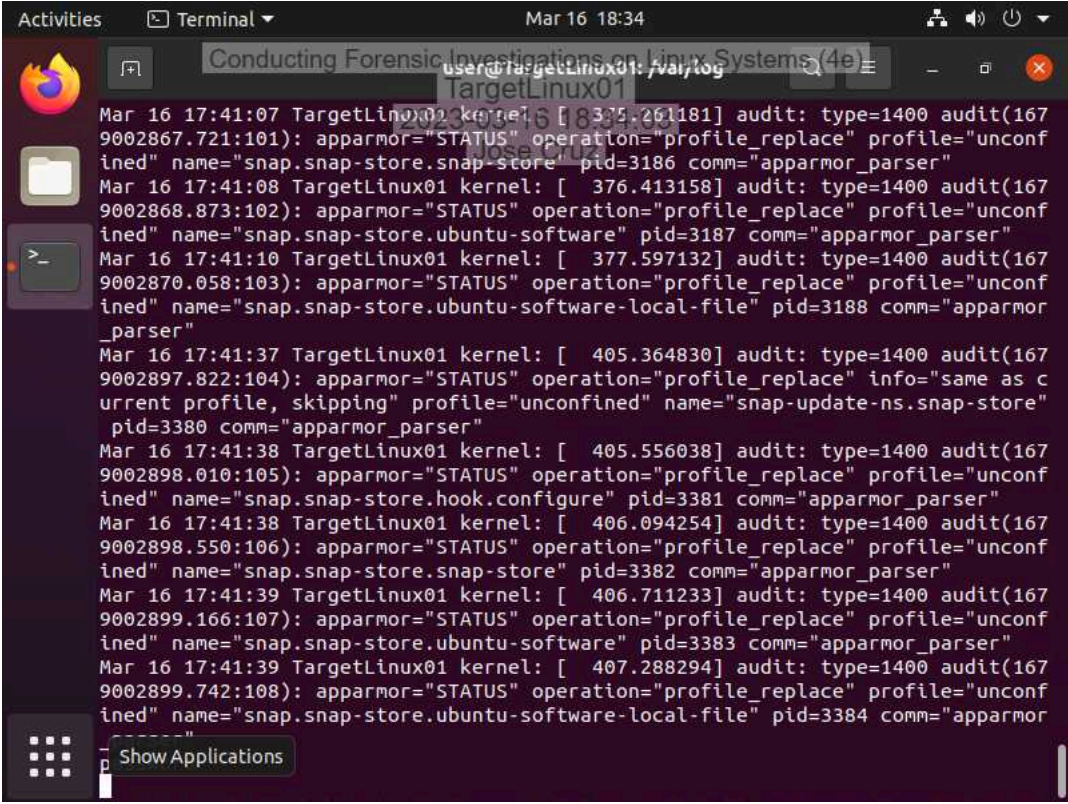
The screenshot shows a terminal window titled "Terminal" with the date and time "Mar 16 18:28". The terminal output displays the results of the `ps -aux` command, showing a list of processes running on the system. The output includes columns for user, PID, PPID, CPU, MEM, VSZ, RSS, T, S, and COMMAND. The processes listed include `user`, `root`, and `[kworker/u2:0]`, `[loop16]`, `[loop17]`, `[loop18]`, `[loop3]`, `[loop1]`, `[loop9]`, `bash`, and `ps -aux`. The terminal also shows the output of the `file` command, which identifies `MyScheduler.txt` as a JPEG image data file.

```
user@TargetLinux01:~/Documents$ ps -aux
user      1553  0.0  0.2 157500 5908 ?        Ssl  17:35   0:00 /usr/libexec
user      1747  0.0  0.4 462544 9272 ?        Ssl  17:35   0:00 /usr/libexec
user      1751  0.0  1.0 493684 22012 ?       Ssl  17:35   0:00 /usr/libexec
root      1762  0.0  3.2 520724 65388 ?        Ssl  17:35   0:01 /usr/libexec
user      2044  0.0  0.2 162252 6040 ?        Ssl  17:36   0:00 /usr/libexec
user      2049  0.0  1.0 419320 22328 ?        Sl   17:36   0:00 update-notif
root      2225  0.0  0.0 0 0 ?        R    17:40   0:00 [kworker/u2:0]
root      2263  0.0  0.0 0 0 ?        S<   17:40   0:00 [loop16]
root      2337  0.0  0.0 0 0 ?        S<   17:40   0:00 [loop17]
root      2398  0.0  0.0 0 0 ?        S<   17:40   0:00 [loop18]
root      2533  0.8  1.9 744264 39208 ?       Ssl  17:40   0:21 /usr/lib/sna
root      2883  0.0  0.0 0 0 ?        S<   17:40   0:00 [loop3]
root      3169  0.0  0.0 0 0 ?        S<   17:41   0:00 [loop1]
root      3312  0.0  0.0 0 0 ?        S<   17:41   0:00 [loop9]
user      3465  0.0  0.4 387788 8412 ?        Sl   17:44   0:00 /usr/libexec
user      3477  0.0  0.4 314988 8288 ?        Sl   17:44   0:00 /usr/libexec
user      3586  0.1  2.4 814448 50596 ?       Ssl  17:56   0:01 /usr/libexec
user      3594  0.0  0.2 10616 4768 pts/0    Ss   17:56   0:00 bash
root      3620  0.0  0.0 0 0 ?        I    18:05   0:00 [kworker/0:0]
root      3632  0.0  0.0 0 0 ?        I    18:12   0:00 [kworker/u2:0]
root      3641  0.0  0.0 0 0 ?        I    18:15   0:00 [kworker/0:2]
user      3660  0.0  0.1 11692 3632 pts/0    R+   18:22   0:00 ps -aux

user@TargetLinux01:/$ cd home/user/Documents
user@TargetLinux01:~/Documents$ ls
MyScheduler.txt
user@TargetLinux01:~/Documents$ file MyScheduler.txt
MyScheduler.txt: JPEG image data, JFIF standard 1.01, resolution (DPI), density
140x140, segment length 16, baseline, precision 8, 800x800, components 3
user@TargetLinux01:~/Documents$
```

### Part 3: Retrieve Logs Files on a Live Linux System

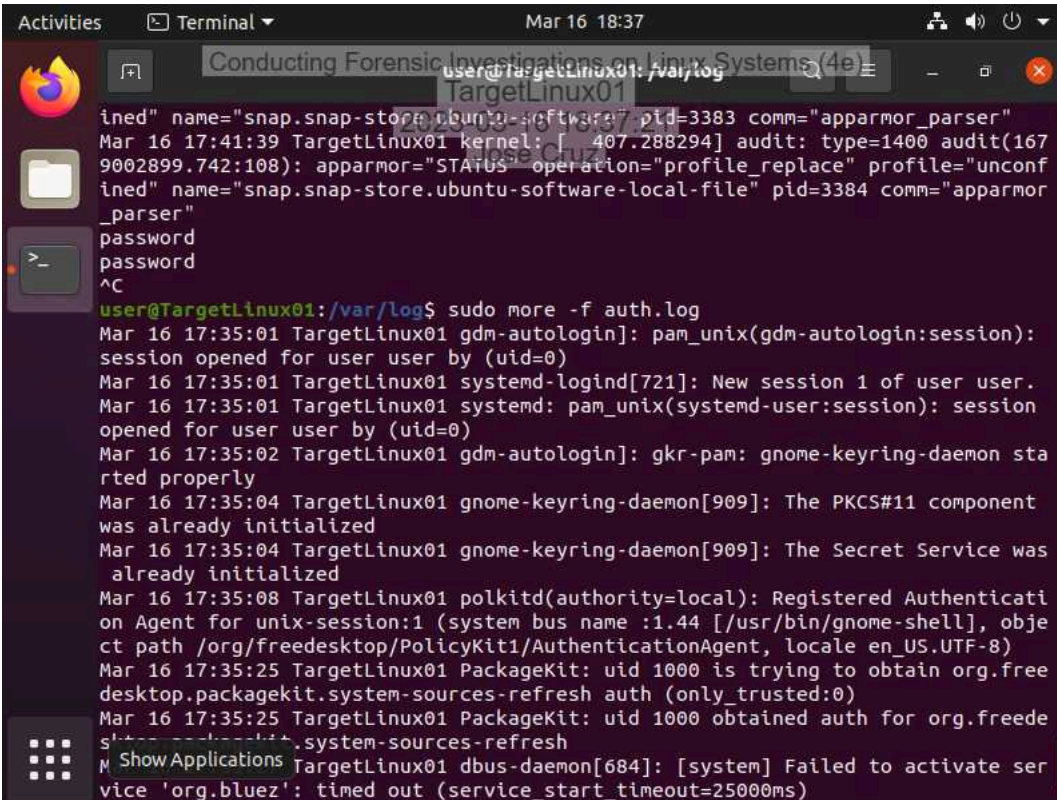
### 4. Make a screen capture showing the records in the kern.log file.



The screenshot shows a terminal window titled "Terminal" with the date and time "Mar 16 18:34". The terminal displays the contents of the file `/var/log/kern.log`, which contains kernel audit records. The records show various system events, including apparmor profile replacements and snap-store operations. The terminal output is as follows:

```
Mar 16 17:41:07 TargetLinux01 kernel: [ 335.261181] audit: type=1400 audit(1679002867.721:101): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.snap-store" pid=3186 comm="apparmor_parser"
Mar 16 17:41:08 TargetLinux01 kernel: [ 376.413158] audit: type=1400 audit(1679002868.873:102): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=3187 comm="apparmor_parser"
Mar 16 17:41:10 TargetLinux01 kernel: [ 377.597132] audit: type=1400 audit(1679002870.058:103): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=3188 comm="apparmor_parser"
Mar 16 17:41:37 TargetLinux01 kernel: [ 405.364830] audit: type=1400 audit(1679002897.822:104): apparmor="STATUS" operation="profile_replace" info="same as current profile, skipping" profile="unconfined" name="snap-update-ns.snap-store" pid=3380 comm="apparmor_parser"
Mar 16 17:41:38 TargetLinux01 kernel: [ 405.556038] audit: type=1400 audit(1679002898.010:105): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.hook.configure" pid=3381 comm="apparmor_parser"
Mar 16 17:41:38 TargetLinux01 kernel: [ 406.094254] audit: type=1400 audit(1679002898.550:106): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.snap-store" pid=3382 comm="apparmor_parser"
Mar 16 17:41:39 TargetLinux01 kernel: [ 406.711233] audit: type=1400 audit(1679002899.166:107): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=3383 comm="apparmor_parser"
Mar 16 17:41:39 TargetLinux01 kernel: [ 407.288294] audit: type=1400 audit(1679002899.742:108): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=3384 comm="apparmor_parser"
```

### 7. Make a screen capture showing the records in the auth.log file.



```
user@TargetLinux01: /var/log
TargetLinux01
Mar 16 17:41:39 TargetLinux01 kernel: [ 407.288294] audit: type=1400 audit(167
9002899.742:108): apparmor="STATUS" operation="profile_replace" profile="unconf
ined" name="snap.snap-store.ubuntu-software-local-file" pid=3384 comm="apparmor
_parser"
password
password
^C
user@TargetLinux01:/var/log$ sudo more -f auth.log
Mar 16 17:35:01 TargetLinux01 gdm-autologin]: pam_unix(gdm-autologin:session):
session opened for user user by (uid=0)
Mar 16 17:35:01 TargetLinux01 systemd-logind[721]: New session 1 of user user.
Mar 16 17:35:01 TargetLinux01 systemd: pam_unix(systemd-user:session): session
opened for user user by (uid=0)
Mar 16 17:35:02 TargetLinux01 gdm-autologin]: gkr-pam: gnome-keyring-daemon sta
rted properly
Mar 16 17:35:04 TargetLinux01 gnome-keyring-daemon[909]: The PKCS#11 component
was already initialized
Mar 16 17:35:04 TargetLinux01 gnome-keyring-daemon[909]: The Secret Service was
already initialized
Mar 16 17:35:08 TargetLinux01 polkitd(authority=local): Registered Authenticati
on Agent for unix-session:1 (system bus name :1.44 [/usr/bin/gnome-shell], obje
ct path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8)
Mar 16 17:35:25 TargetLinux01 PackageKit: uid 1000 is trying to obtain org.free
desktop.packagekit.system-sources-refresh auth (only_trusted:0)
Mar 16 17:35:25 TargetLinux01 PackageKit: uid 1000 obtained auth for org.freede
sktop.packagekit.system-sources-refresh
Show Applications TargetLinux01 dbus-daemon[684]: [system] Failed to activate ser
vice 'org.bluez': timed out (service_start_timeout=25000ms)
```

## Section 2: Applied Learning

### Part 1: Identify Login Attempts on a Linux Drive Image

15. **Document** the names of the two non-root users that attempted to log in, the number of attempts detected, the date/time range of the attempts, the source IP address for the login attempts, and the port.

The first unauthorized username was "gdm" and "pam\_unix". The number of attempts for gdm was 6 max entries and for pam\_unix was 5 attempts. The date and time for gdm was Jun 11 at 4:47:50. For user pam\_unix date and time was Jun 11 at 5:05:01. Seems the Ip address for both users are 192.168.78.1. The ports being used are 22, 4663, and 3521.

17. **Document** the date and time the most recent successful login for the user(s) that you previously identified in step 15.

User gdm date and time of opened session was in June 11 at 06:08:53. For user pam\_unix the date and time is June 11 at 06:09:01.

### Part 2: Identify Software Installations on a Linux Drive Image

3. **Document** the applications that were installed using apt-get, then use the Internet to identify the ones that might be considered suspicious.

Seems in the advance search the software "logkeys" was installed on June 10 at 10:44:57.

### Part 3: Identify External Drive Attachments on a Linux Drive Image

4. **Document** when the USB storage device was connected and its serial number.

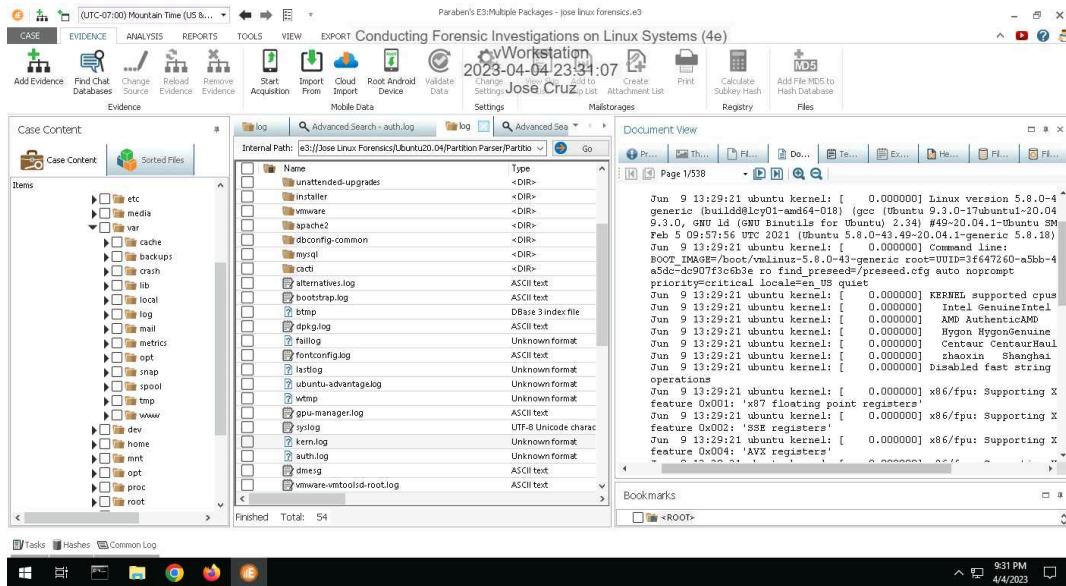
The current USB was connected on June 10 at 10:10:24:42, in addition the serial number for the usb is = 3.



### Section 3: Challenge and Analysis

#### Part 1: Identify Recently Printed Files on a Linux Drive Image

Make a screen capture showing the contents of the printer log file.



#### Part 2: Identify Disk Imaging on a Linux Drive Image

Make a screen capture showing the record of the dd command in the Text View.

