

Conducting Forensic Investigations on Linux Systems (4e)

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

Student:

muhammad zahid

Email:

zahid1mz@cmich.edu

Time on Task:

3 hours, 4 minutes

Progress:

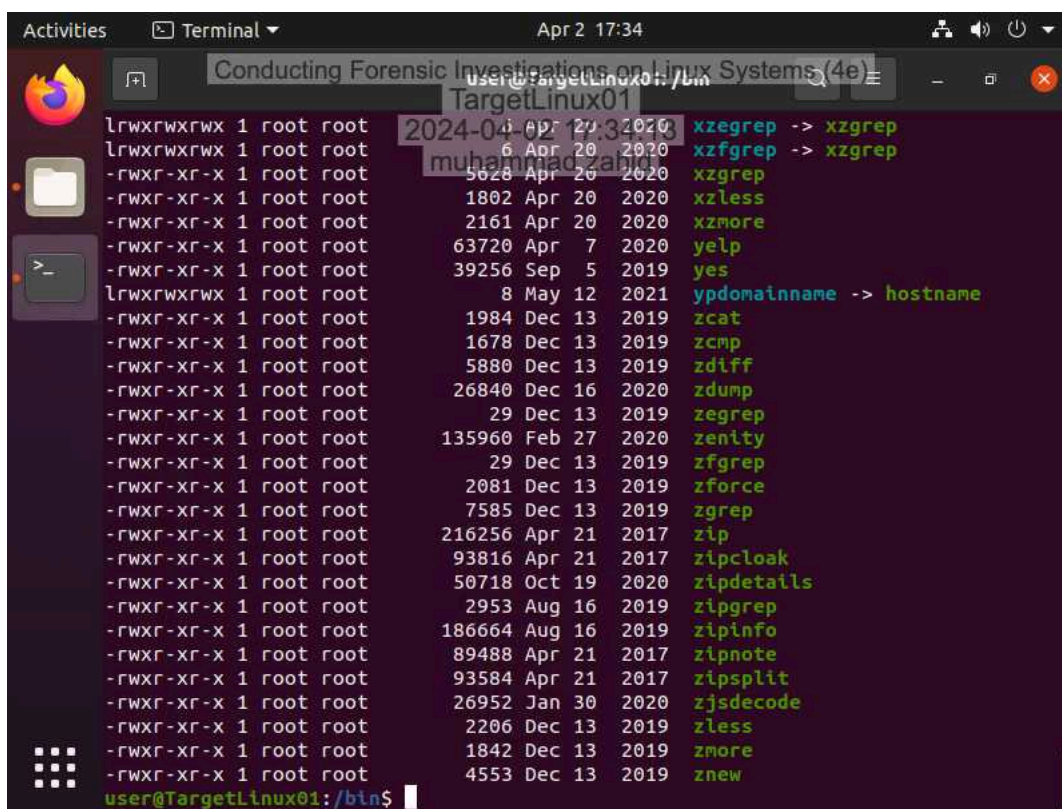
100%

Report Generated: Tuesday, April 2, 2024 at 6:40 PM

Section 1: Hands-On Demonstration

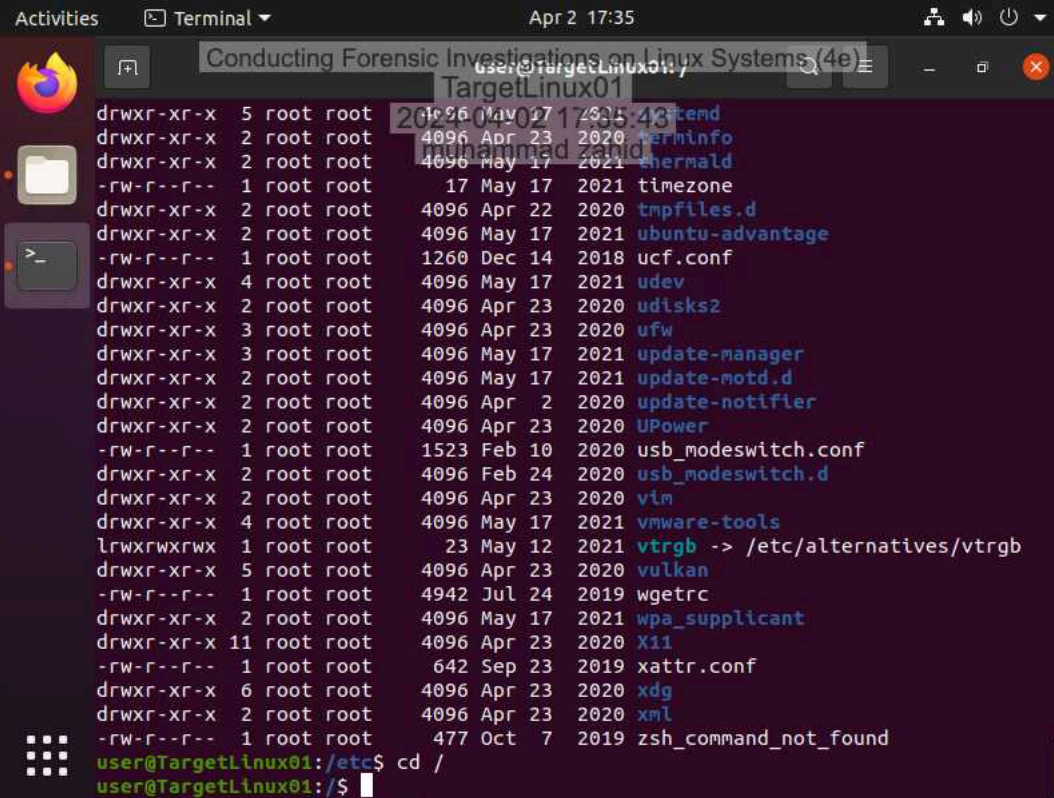
Part 1: Explore a Live Linux System

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



```
user@TargetLinux01: /bin$ ls -l
lrwxrwxrwx 1 root root 4096 Apr 27 2020 xzgrep -> xzgrep
lrwxrwxrwx 1 root root 4096 Apr 20 2020 xzfgrep -> xzfgrep
-rwxr-xr-x 1 root root 5628 Apr 20 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$
```

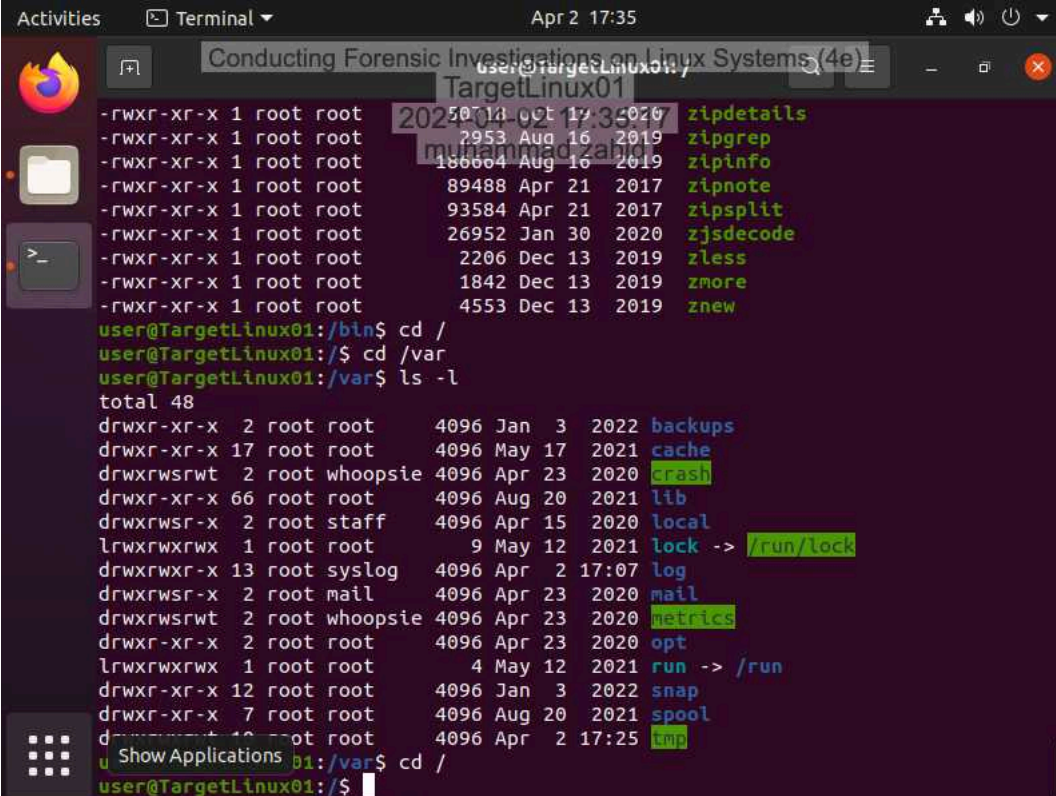
20. Make a screen capture showing the contents of the /etc directory.



```
user@TargetLinux01: /etc$ ls -la
drwxr-xr-x 5 root root 4096 Apr 23 2020 .
drwxr-xr-x 2 root root 4096 Apr 23 2020 ..
drwxr-xr-x 2 root root 4096 May 17 2021 alternatives
-rw-r--r-- 1 root root 17 May 17 2021 timezone
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$ cd /
user@TargetLinux01:/$
```

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



The screenshot shows a terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" with the prompt "user@TargetLinux01:". The user has entered the following commands:

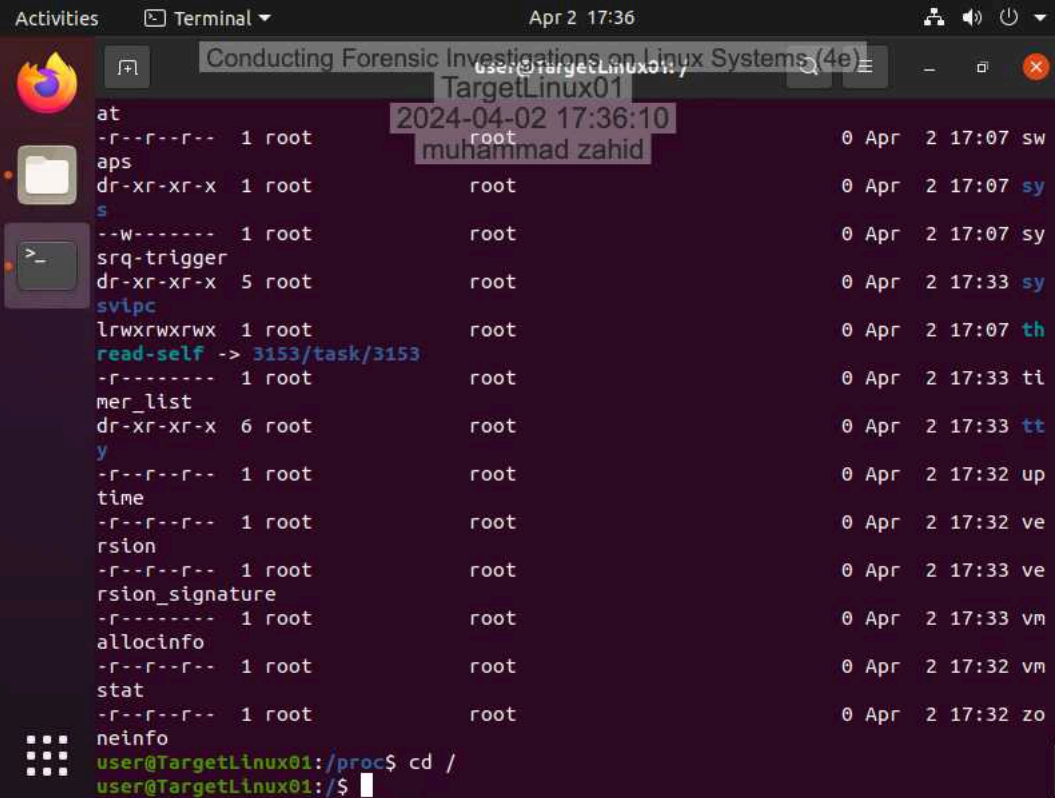
```
user@TargetLinux01:/bin$ cd /
user@TargetLinux01:/$ cd /var
user@TargetLinux01:/var$ ls -l
```

The output of the `ls -l` command is as follows:

```
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 Apr  2  17:07 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
drwxr-xr-x 10 root root     4096 Apr  2  17:25 tmp
```

The user then enters the command `cd /`, and the prompt returns to `user@TargetLinux01:/`.

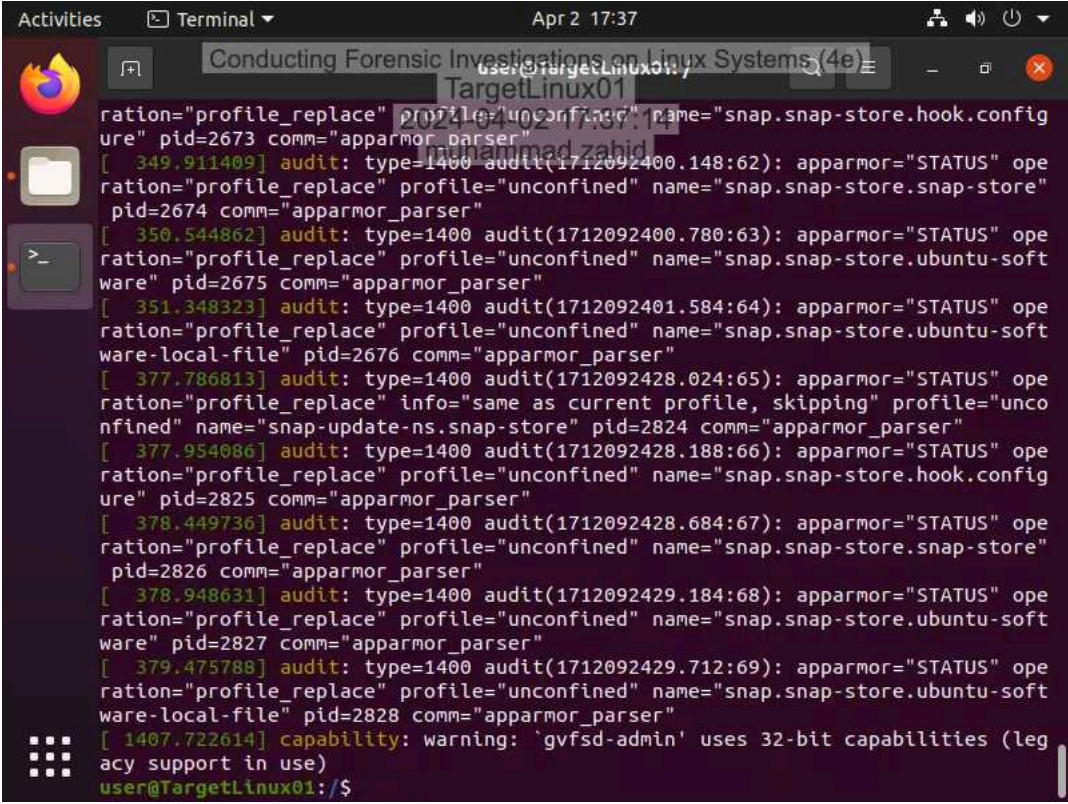
22. Make a screen capture showing the contents of the /proc directory.



```
user@TargetLinux01:/$ ls -la /proc
total 0
dr-xr-xr-x  1 root    root           0 Apr  2 17:07 sw
dr-xr-xr-x  1 root    root           0 Apr  2 17:07 sy
--w-----  1 root    root           0 Apr  2 17:07 sy
srq-trigger  5 root    root           0 Apr  2 17:33 sy
svipc        1 root    root           0 Apr  2 17:07 th
lrwxrwxrwx  1 root    root           0 Apr  2 17:33 ti
-r-----  1 root    root           0 Apr  2 17:33 tt
mer_list     6 root    root           0 Apr  2 17:32 up
y            1 root    root           0 Apr  2 17:32 ve
time         1 root    root           0 Apr  2 17:33 ve
rsion        1 root    root           0 Apr  2 17:33 vm
rsion_signature 1 root    root           0 Apr  2 17:33 vm
allocinfo    1 root    root           0 Apr  2 17:32 zo
stat         1 root    root           0 Apr  2 17:32 zo
neinfo       1 root    root           0 Apr  2 17:32 zo
user@TargetLinux01:/proc$ cd /
user@TargetLinux01:/$
```

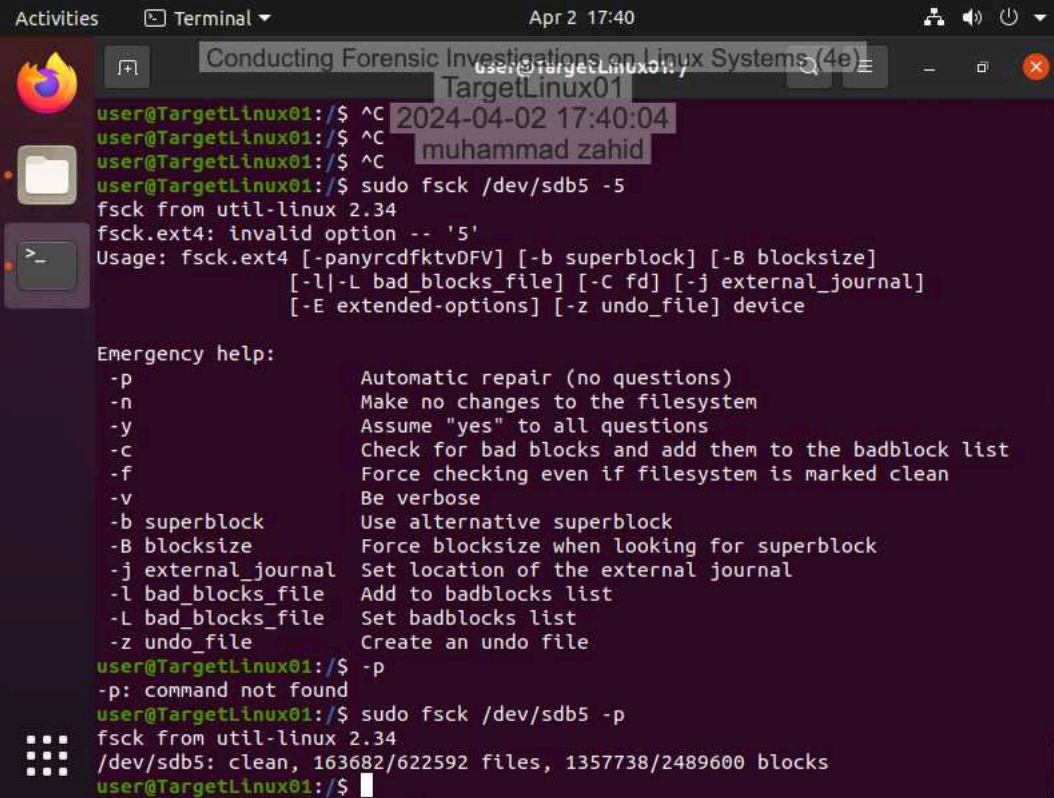
Part 2: Use Linux Shell Commands for Forensic Investigations

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)". The terminal shows the output of the `dmesg` command, displaying various audit messages. The messages include timestamps in brackets, audit types (e.g., `audit: type=1400`), and details about profile replacements and apparmor operations. The terminal prompt is `user@TargetLinux01:/$`.

```
ration="profile_replace" profile="unconfined" name="snap.snap-store.hook.config
ure" pid=2673 comm="apparmor_parser"
[ 349.911409] audit: type=1400 audit(1712092400.148:62): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.snap-store"
pid=2674 comm="apparmor_parser"
[ 350.544862] audit: type=1400 audit(1712092400.780:63): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-soft
ware" pid=2675 comm="apparmor_parser"
[ 351.348323] audit: type=1400 audit(1712092401.584:64): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-soft
ware-local-file" pid=2676 comm="apparmor_parser"
[ 377.786813] audit: type=1400 audit(1712092428.024:65): apparmor="STATUS" ope
ration="profile_replace" info="same as current profile, skipping" profile="unco
nfined" name="snap-update-ns.snap-store" pid=2824 comm="apparmor_parser"
[ 377.954086] audit: type=1400 audit(1712092428.188:66): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.hook.config
ure" pid=2825 comm="apparmor_parser"
[ 378.449736] audit: type=1400 audit(1712092428.684:67): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.snap-store"
pid=2826 comm="apparmor_parser"
[ 378.948631] audit: type=1400 audit(1712092429.184:68): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-soft
ware" pid=2827 comm="apparmor_parser"
[ 379.475788] audit: type=1400 audit(1712092429.712:69): apparmor="STATUS" ope
ration="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-soft
ware-local-file" pid=2828 comm="apparmor_parser"
[ 1407.722614] capability: warning: 'gvfsd-admin' uses 32-bit capabilities (leg
acy support in use)
user@TargetLinux01:/$
```

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

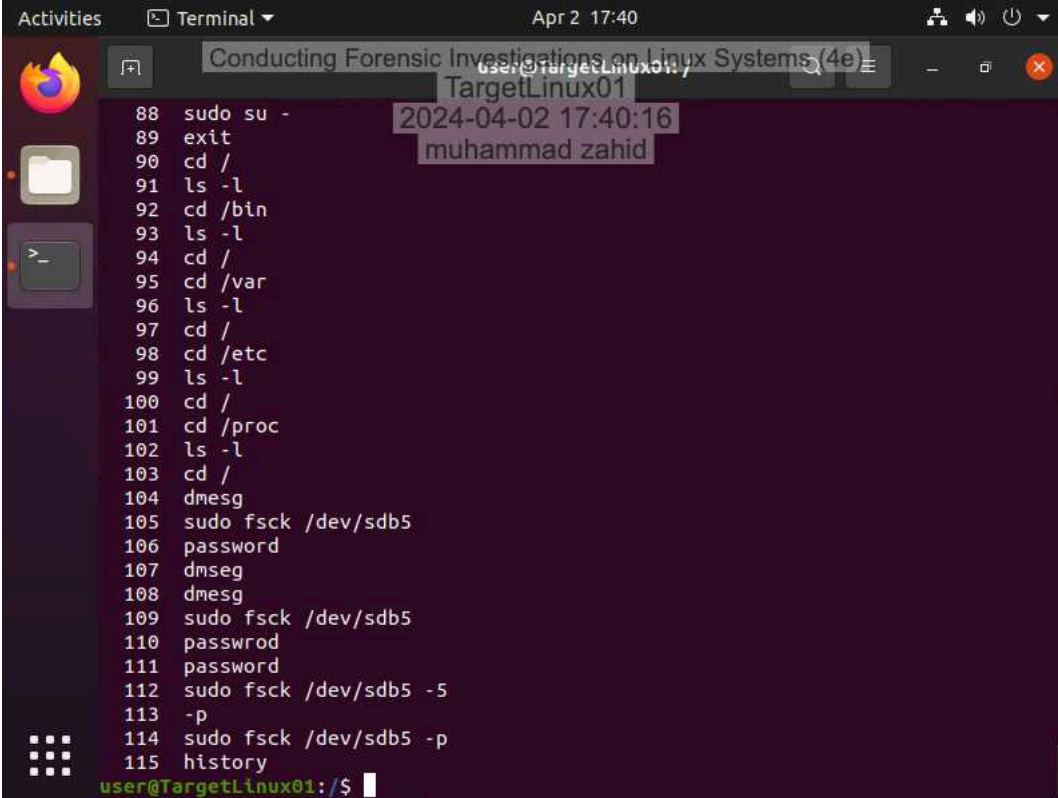


A terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" is shown. The window has a dark background with a terminal icon on the left. The terminal displays the following commands and output:

```
user@TargetLinux01:/$ ^C
user@TargetLinux01:/$ ^C
user@TargetLinux01:/$ ^C
user@TargetLinux01:/$ sudo fsck /dev/sdb5 -5
fsck from util-linux 2.34
fsck.ext4: invalid option -- '5'
Usage: fsck.ext4 [-panyrcdfktvDFV] [-b superblock] [-B blocksizes]
               [-l|-L bad_blocks_file] [-C fd] [-j external_journal]
               [-E extended-options] [-z undo_file] device

Emergency help:
-p           Automatic repair (no questions)
-n           Make no changes to the filesystem
-y           Assume "yes" to all questions
-c           Check for bad blocks and add them to the badblock list
-f           Force checking even if filesystem is marked clean
-v           Be verbose
-b superblock Use alternative superblock
-B blocksizes Force blocksizes when looking for superblock
-j external_journal Set location of the external journal
-l bad_blocks_file Add to badblocks list
-L bad_blocks_file Set badblocks list
-z undo_file  Create an undo file
user@TargetLinux01:/$ -p
-p: command not found
user@TargetLinux01:/$ sudo fsck /dev/sdb5 -p
fsck from util-linux 2.34
/dev/sdb5: clean, 163682/622592 files, 1357738/2489600 blocks
user@TargetLinux01:/$
```

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

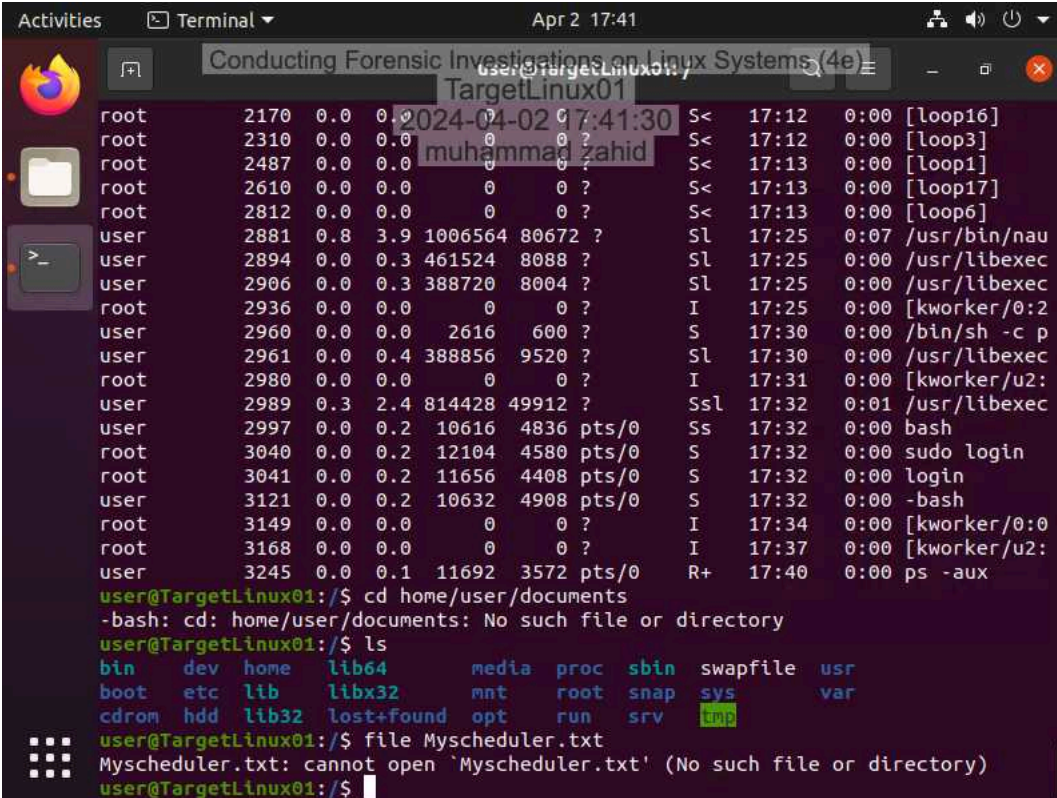
A screenshot of a Linux terminal window. The window title is "Terminal" and the system clock shows "Apr 2 17:40". The terminal displays a list of commands and their outputs, numbered 88 to 115. The commands include directory navigation, file listing, and disk checks. The output for the "history" command is visible at the bottom. The terminal prompt is "user@TargetLinux01:/\$".

```
88 sudo su -
89 exit
90 cd /
91 ls -l
92 cd /bin
93 ls -l
94 cd /
95 cd /var
96 ls -l
97 cd /
98 cd /etc
99 ls -l
100 cd /
101 cd /proc
102 ls -l
103 cd /
104 dmesg
105 sudo fsck /dev/sdb5
106 password
107 dmesg
108 dmesg
109 sudo fsck /dev/sdb5
110 password
111 password
112 sudo fsck /dev/sdb5 -5
113 -p
114 sudo fsck /dev/sdb5 -p
115 history
user@TargetLinux01:/$
```

11. Make a screen capture showing the running processes.

```
user@TargetLinux01:~$ ps -aux
user      1481  0.0  0.2 162507 5712 ?        Ssl  17:07   0:00 /usr/libexec
user      1509  0.0  0.2 457520 5728 ?        Ssl  17:07   0:00 /usr/libexec
user      1697  0.0  0.4 462536 8368 ?        Ssl  17:07   0:00 /usr/libexec
user      1701  0.0  1.0 493684 20596 ?        Ssl  17:07   0:00 /usr/libexec
root      1712  0.0  3.2 529340 66668 ?        Ssl  17:07   0:01 /usr/libexec
user      1996  0.0  0.2 162252 5692 ?        Ssl  17:08   0:00 /usr/libexec
user      2001  0.0  1.0 493040 22192 ?        SL   17:08   0:00 update-notif
root      2144  0.0  0.0 0 0 ?        I    17:12   0:01 [kworker/u2:]
root      2170  0.0  0.0 0 0 ?        S<   17:12   0:00 [loop16]
root      2310  0.0  0.0 0 0 ?        S<   17:12   0:00 [loop3]
root      2487  0.0  0.0 0 0 ?        S<   17:13   0:00 [loop1]
root      2610  0.0  0.0 0 0 ?        S<   17:13   0:00 [loop17]
root      2812  0.0  0.0 0 0 ?        S<   17:13   0:00 [loop6]
user      2881  0.8  3.9 1006564 80672 ?        SL   17:25   0:07 /usr/bin/nau
user      2894  0.0  0.3 461524 8088 ?        SL   17:25   0:00 /usr/libexec
user      2906  0.0  0.3 388720 8004 ?        SL   17:25   0:00 /usr/libexec
root      2936  0.0  0.0 0 0 ?        I    17:25   0:00 [kworker/0:2]
user      2960  0.0  0.0 2616 600 ?        S    17:30   0:00 /bin/sh -c p
user      2961  0.0  0.4 388856 9520 ?        SL   17:30   0:00 /usr/libexec
root      2980  0.0  0.0 0 0 ?        I    17:31   0:00 [kworker/u2:]
user      2989  0.3  2.4 814428 49912 ?        Ssl  17:32   0:01 /usr/libexec
user      2997  0.0  0.2 10616 4836 pts/0    Ss   17:32   0:00 bash
root      3040  0.0  0.2 12104 4580 pts/0    S    17:32   0:00 sudo login
root      3041  0.0  0.2 11656 4408 pts/0    S    17:32   0:00 login
user      3121  0.0  0.2 10632 4908 pts/0    S    17:32   0:00 -bash
root      3149  0.0  0.0 0 0 ?        I    17:34   0:00 [kworker/0:0]
root      3168  0.0  0.0 0 0 ?        I    17:37   0:00 [kworker/u2:]
user      3245  0.0  0.1 11692 3572 pts/0    R+   17:40   0:00 ps -aux
user@TargetLinux01:/$
```


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



The screenshot shows a terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" with the prompt "user@TargetLinux01:". The terminal displays the output of the 'file' command on 'Myscheduler.txt', which is a list of system processes. The output is as follows:

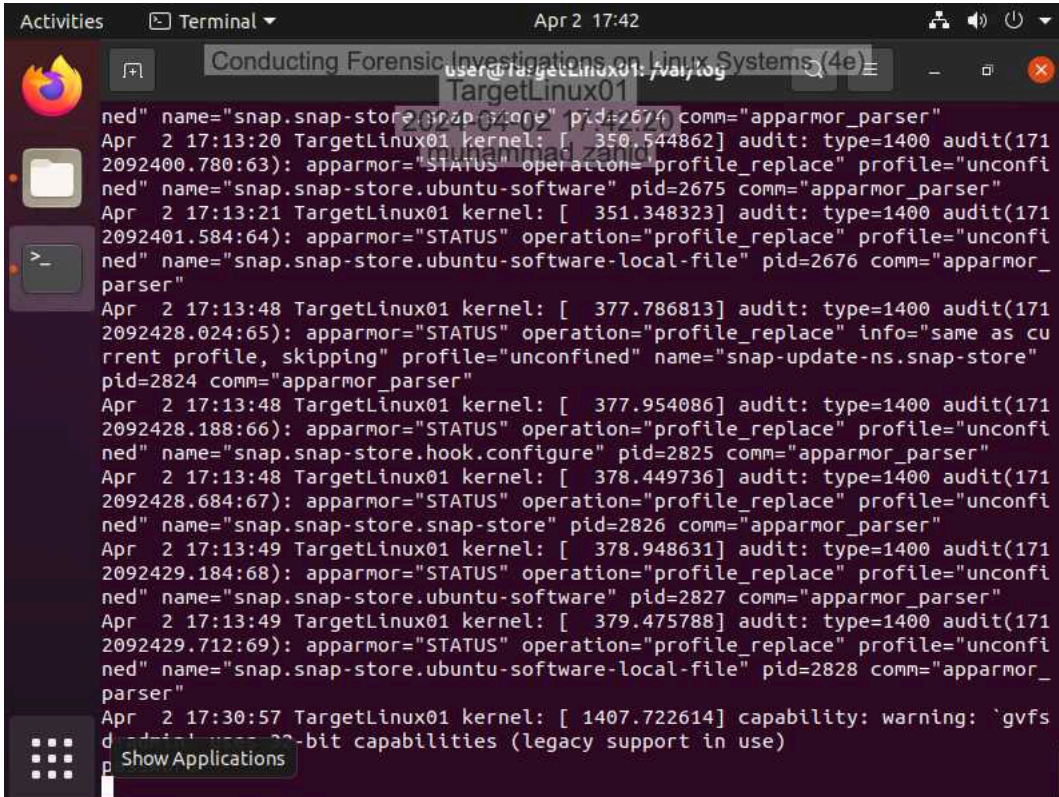
```
user@TargetLinux01:/$ file Myscheduler.txt
Myscheduler.txt: cannot open 'Myscheduler.txt' (No such file or directory)
user@TargetLinux01:/$
```

The terminal also shows the output of the 'ls' command, which lists the contents of the current directory:

```
user@TargetLinux01:/$ ls
bin  dev  home  lib64  media  proc  sbin  swapfile  usr
boot  etc  lib  libx32  mnt  root  snap  sys  var
cdrom  hdd  lib32  lost+found  opt  run  srv  tmp
```

Part 3: Retrieve Logs Files on a Live Linux System

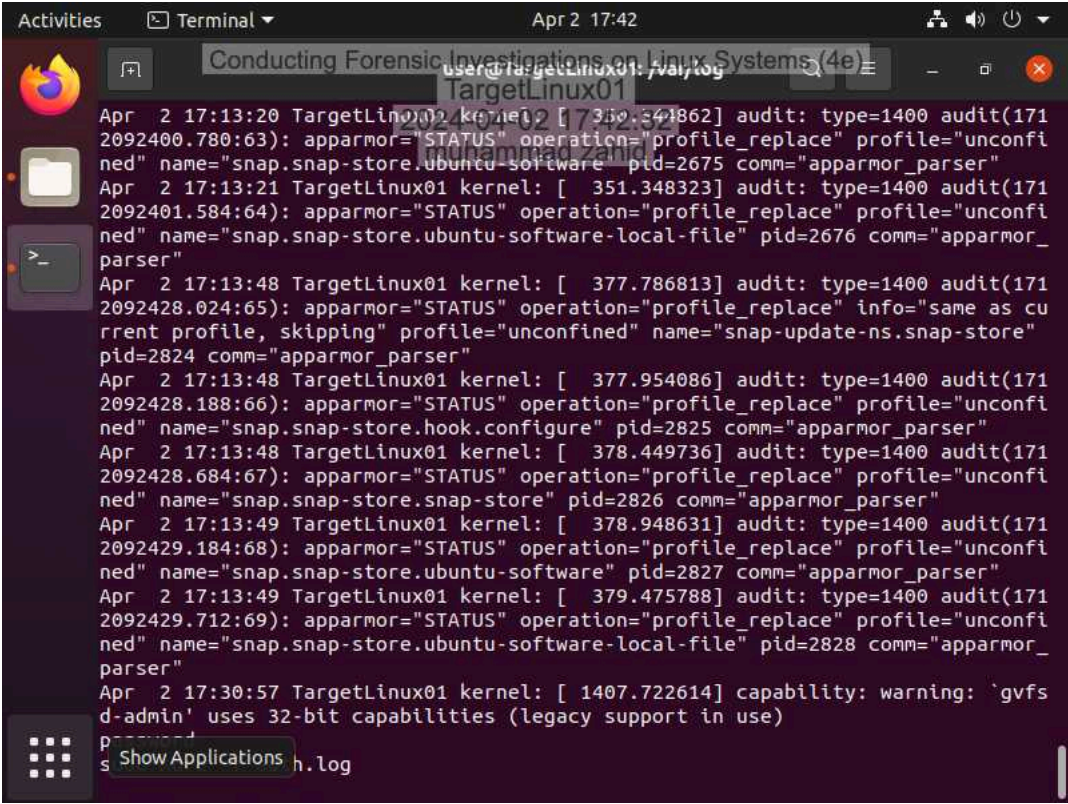
4. Make a screen capture showing the records in the kern.log file.



The screenshot shows a terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" with the command `sudo cat /var/log/kern.log` executed. The output displays a series of kernel audit records from the `TargetLinux01` system, showing `apparmor_parser` operations. The records include timestamps, kernel IDs, audit types, and details about the operations being performed on various snap packages.

```
ned" name="snap.snap-store.snap-store" pid=2674 comm="apparmor_parser"
Apr  2 17:13:20 TargetLinux01 kernel: [ 350.544862] audit: type=1400 audit(171
2092400.780:63): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software" pid=2675 comm="apparmor_parser"
Apr  2 17:13:21 TargetLinux01 kernel: [ 351.348323] audit: type=1400 audit(171
2092401.584:64): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software-local-file" pid=2676 comm="apparmor_
parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 377.786813] audit: type=1400 audit(171
2092428.024:65): apparmor="STATUS" operation="profile_replace" info="same as cu
rrent profile, skipping" profile="unconfined" name="snap-update-ns.snap-store"
pid=2824 comm="apparmor_parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 377.954086] audit: type=1400 audit(171
2092428.188:66): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.hook.configure" pid=2825 comm="apparmor_parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 378.449736] audit: type=1400 audit(171
2092428.684:67): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.snap-store" pid=2826 comm="apparmor_parser"
Apr  2 17:13:49 TargetLinux01 kernel: [ 378.948631] audit: type=1400 audit(171
2092429.184:68): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software" pid=2827 comm="apparmor_parser"
Apr  2 17:13:49 TargetLinux01 kernel: [ 379.475788] audit: type=1400 audit(171
2092429.712:69): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software-local-file" pid=2828 comm="apparmor_
parser"
Apr  2 17:30:57 TargetLinux01 kernel: [ 1407.722614] capability: warning: `gvfs
d' default uses 32-bit capabilities (legacy support in use)
p Show Applications
```

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



The screenshot shows a terminal window titled "Terminal" with the date and time "Apr 2 17:42". The terminal displays a series of audit logs from the file `/var/log/auth.log`. The logs are timestamped and include details about system events, such as profile replacements and operations performed by the `apparmor_parser` command. The logs are as follows:

```
Apr  2 17:13:20 TargetLinux01 kernel: [ 351.341862] audit: type=1400 audit(1712092400.780:63): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=2675 comm="apparmor_parser"
Apr  2 17:13:21 TargetLinux01 kernel: [ 351.348323] audit: type=1400 audit(1712092401.584:64): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=2676 comm="apparmor_parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 377.786813] audit: type=1400 audit(1712092428.024:65): apparmor="STATUS" operation="profile_replace" info="same as current profile, skipping" profile="unconfined" name="snap-update-ns.snap-store" pid=2824 comm="apparmor_parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 377.954086] audit: type=1400 audit(1712092428.188:66): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.hook.configure" pid=2825 comm="apparmor_parser"
Apr  2 17:13:48 TargetLinux01 kernel: [ 378.449736] audit: type=1400 audit(1712092428.684:67): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.snap-store" pid=2826 comm="apparmor_parser"
Apr  2 17:13:49 TargetLinux01 kernel: [ 378.948631] audit: type=1400 audit(1712092429.184:68): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software" pid=2827 comm="apparmor_parser"
Apr  2 17:13:49 TargetLinux01 kernel: [ 379.475788] audit: type=1400 audit(1712092429.712:69): apparmor="STATUS" operation="profile_replace" profile="unconfined" name="snap.snap-store.ubuntu-software-local-file" pid=2828 comm="apparmor_parser"
Apr  2 17:30:57 TargetLinux01 kernel: [ 1407.722614] capability: warning: 'gvfsd-admin' uses 32-bit capabilities (legacy support in use)
```

At the bottom of the terminal window, there is a prompt `ps` and a button labeled "Show Applications".

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.

user noel usergdm

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

june 11 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.

watching system buttons

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

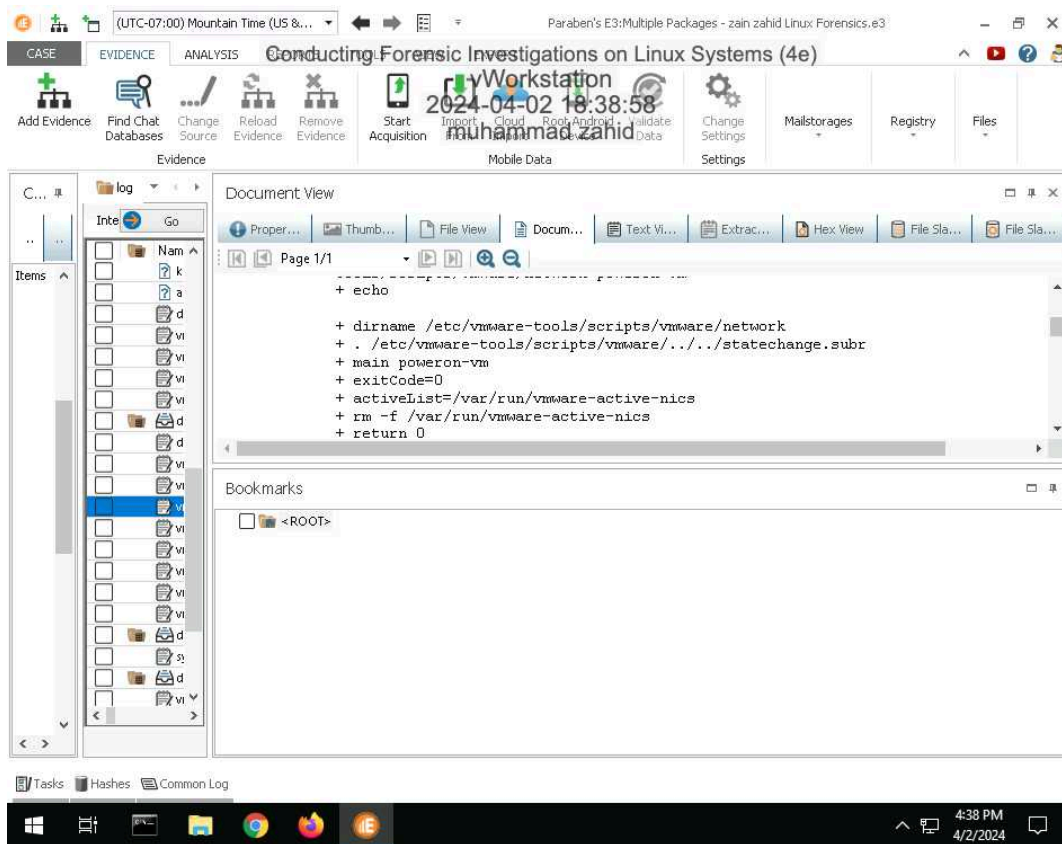
4. **Document** when the USB storage device was connected and its serial number.

Jun 9 13:29:21 9.3.0

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

Conducting Forensic Investigations on Linux Systems (4e)

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

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

