

Conducting Forensic Investigations on Linux Systems (4e)

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

Student:

Karan Atul Vora

Email:

kxv230021@utdallas.edu

Time on Task:

5 hours, 6 minutes

Progress:

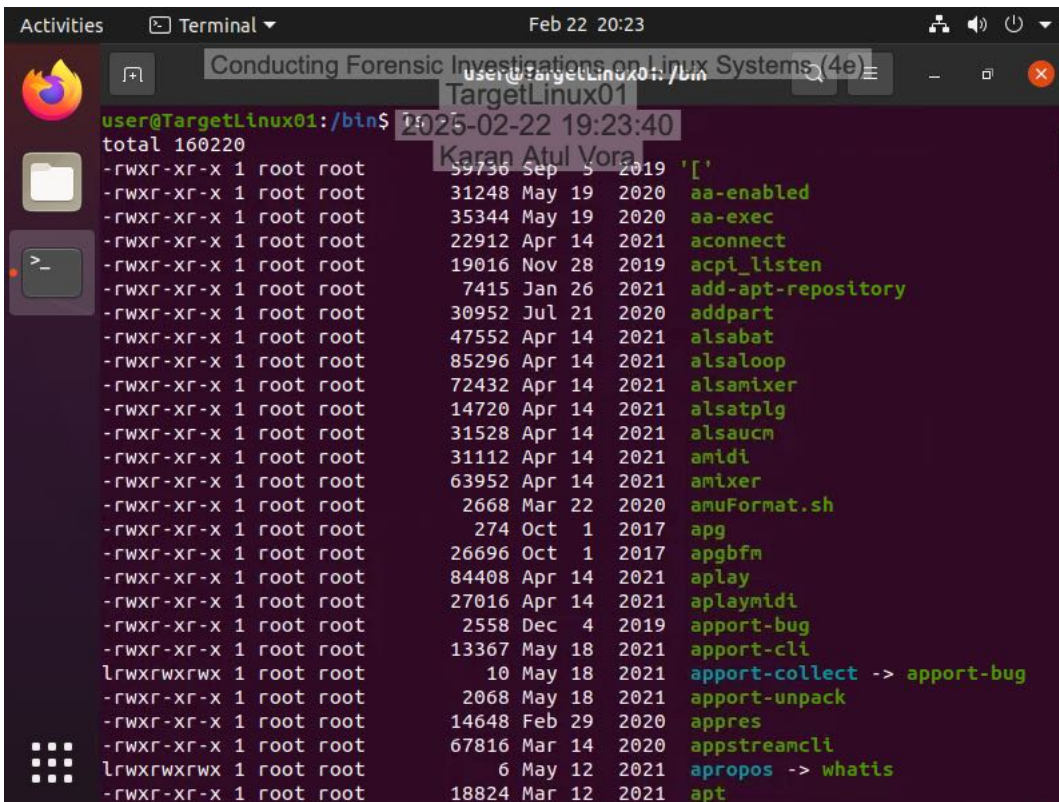
100%

Report Generated: Monday, February 24, 2025 at 4:29 PM

Section 1: Hands-On Demonstration

Part 1: Explore a Live Linux System

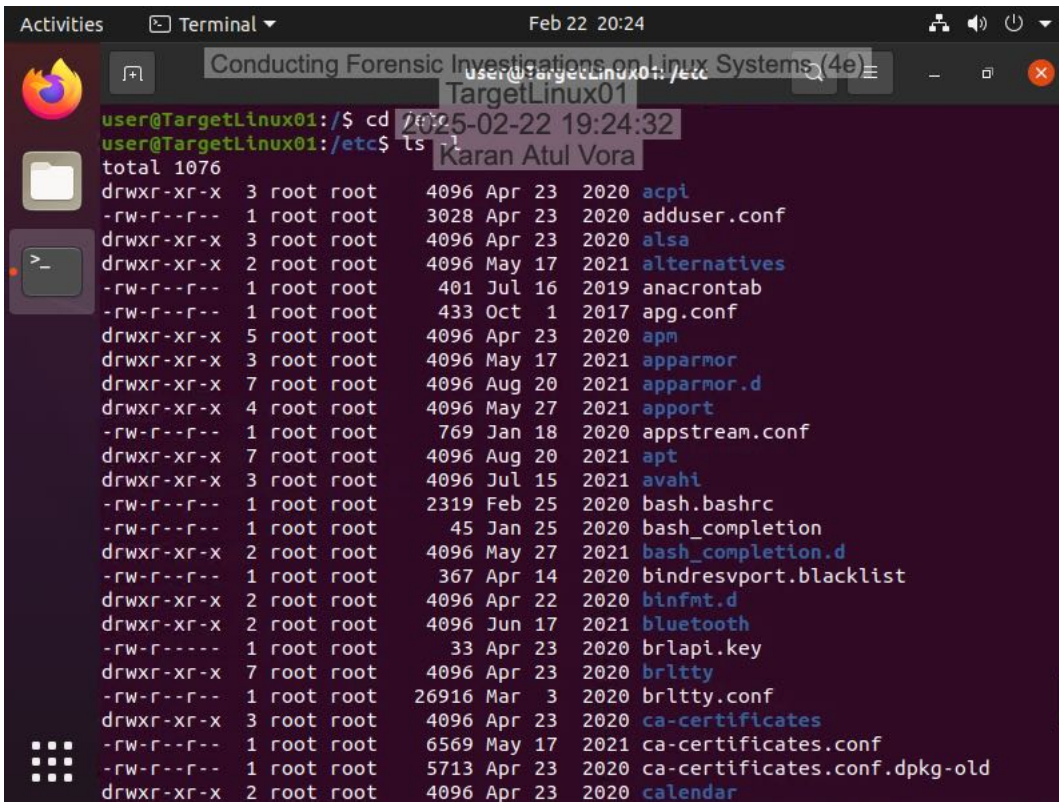
17. Make a screen capture showing the contents of the `/bin` directory.



A terminal window titled "Terminal" showing the command `ls -l /bin` and its output. The output lists various executables in the `/bin` directory, including `aa-enabled`, `aa-exec`, `aconect`, `acpi_listen`, `add-apt-repository`, `addpart`, `alsabat`, `alsaloop`, `alsamixer`, `alsatplg`, `alsaucm`, `amidi`, `amixer`, `amuFormat.sh`, `apg`, `apgbfm`, `aplay`, `aplaymidi`, `apport-bug`, `apport-cli`, `apport-collect`, `apport-unpack`, `appres`, `appstreamcli`, `apropos`, and `apt`. The terminal window also shows the user's name "Karan Atul Vora" and the date "Feb 22 20:23".

```
user@TargetLinux01: /bin$ ls -l /bin
total 160220
-rwxr-xr-x 1 root root 39736 Sep  5 2019 '['
-rwxr-xr-x 1 root root 31248 May 19 2020 aa-enabled
-rwxr-xr-x 1 root root 35344 May 19 2020 aa-exec
-rwxr-xr-x 1 root root 22912 Apr 14 2021 aconect
-rwxr-xr-x 1 root root 19016 Nov 28 2019 acpi_listen
-rwxr-xr-x 1 root root 7415 Jan 26 2021 add-apt-repository
-rwxr-xr-x 1 root root 30952 Jul 21 2020 addpart
-rwxr-xr-x 1 root root 47552 Apr 14 2021 alsabat
-rwxr-xr-x 1 root root 85296 Apr 14 2021 alsaloop
-rwxr-xr-x 1 root root 72432 Apr 14 2021 alsamixer
-rwxr-xr-x 1 root root 14720 Apr 14 2021 alsatplg
-rwxr-xr-x 1 root root 31528 Apr 14 2021 alsaucm
-rwxr-xr-x 1 root root 31112 Apr 14 2021 amidi
-rwxr-xr-x 1 root root 63952 Apr 14 2021 amixer
-rwxr-xr-x 1 root root 2668 Mar 22 2020 amuFormat.sh
-rwxr-xr-x 1 root root 274 Oct  1 2017 apg
-rwxr-xr-x 1 root root 26696 Oct  1 2017 apgbfm
-rwxr-xr-x 1 root root 84408 Apr 14 2021 aplay
-rwxr-xr-x 1 root root 27016 Apr 14 2021 aplaymidi
-rwxr-xr-x 1 root root 2558 Dec  4 2019 apport-bug
-rwxr-xr-x 1 root root 13367 May 18 2021 apport-cli
lrwxrwxrwx 1 root root 10 May 18 2021 apport-collect -> apport-bug
-rwxr-xr-x 1 root root 2068 May 18 2021 apport-unpack
-rwxr-xr-x 1 root root 14648 Feb 29 2020 appres
-rwxr-xr-x 1 root root 67816 Mar 14 2020 appstreamcli
lrwxrwxrwx 1 root root 6 May 12 2021 apropos -> whatis
-rwxr-xr-x 1 root root 18824 Mar 12 2021 apt
```

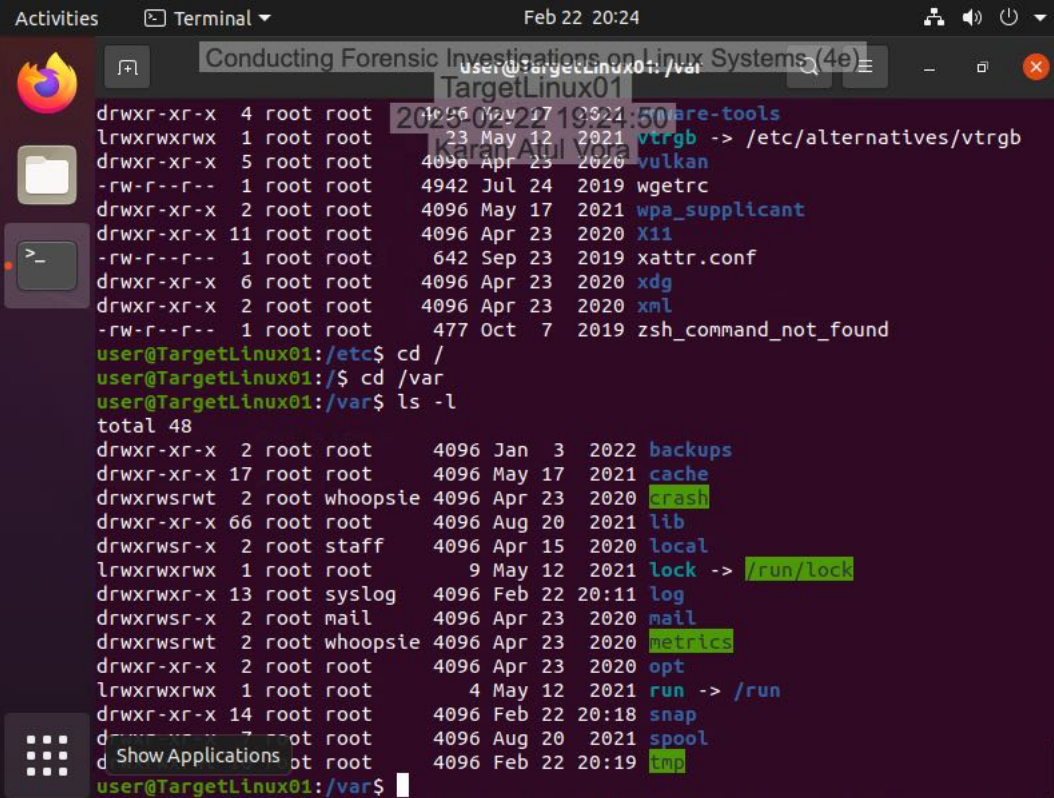
20. Make a screen capture showing the contents of the `/etc` directory.



A screenshot of a Linux terminal window titled "Terminal" with a date and time of "Feb 22 20:24". The terminal shows a user at "TargetLinux01" with the prompt "user@TargetLinux01:/\$". The user has navigated to the `/etc` directory, as indicated by the prompt "user@TargetLinux01:/etc\$". The user has then executed the `ls -l` command, which displays a long listing of files and directories in the `/etc` directory. The output shows various system configuration files and directories, including `acpi`, `adduser.conf`, `alsa`, `alternatives`, `anacrontab`, `apg.conf`, `apm`, `apparmor`, `apparmor.d`, `appport`, `appstream.conf`, `apt`, `avahi`, `bash.bashrc`, `bash_completion`, `bash_completion.d`, `bindresvport.blacklist`, `binfmt.d`, `bluetooth`, `brlapi.key`, `brltty`, `brltty.conf`, `ca-certificates`, `ca-certificates.conf`, `ca-certificates.conf.dpkg-old`, and `calendar`. The files are listed with their permissions, owner, group, size, and modification date.

```
total 1076
drwxr-xr-x 3 root root 4096 Apr 23 2020 acpi
-rw-r--r-- 1 root root 3028 Apr 23 2020 adduser.conf
drwxr-xr-x 3 root root 4096 Apr 23 2020 alsa
drwxr-xr-x 2 root root 4096 May 17 2021 alternatives
-rw-r--r-- 1 root root 401 Jul 16 2019 anacrontab
-rw-r--r-- 1 root root 433 Oct 1 2017 apg.conf
drwxr-xr-x 5 root root 4096 Apr 23 2020 apm
drwxr-xr-x 3 root root 4096 May 17 2021 apparmor
drwxr-xr-x 7 root root 4096 Aug 20 2021 apparmor.d
drwxr-xr-x 4 root root 4096 May 27 2021 appport
-rw-r--r-- 1 root root 769 Jan 18 2020 appstream.conf
drwxr-xr-x 7 root root 4096 Aug 20 2021 apt
drwxr-xr-x 3 root root 4096 Jul 15 2021 avahi
-rw-r--r-- 1 root root 2319 Feb 25 2020 bash.bashrc
-rw-r--r-- 1 root root 45 Jan 25 2020 bash_completion
drwxr-xr-x 2 root root 4096 May 27 2021 bash_completion.d
-rw-r--r-- 1 root root 367 Apr 14 2020 bindresvport.blacklist
drwxr-xr-x 2 root root 4096 Apr 22 2020 binfmt.d
drwxr-xr-x 2 root root 4096 Jun 17 2021 bluetooth
-rw-r--r-- 1 root root 33 Apr 23 2020 brlapi.key
drwxr-xr-x 7 root root 4096 Apr 23 2020 brltty
-rw-r--r-- 1 root root 26916 Mar 3 2020 brltty.conf
drwxr-xr-x 3 root root 4096 Apr 23 2020 ca-certificates
-rw-r--r-- 1 root root 6569 May 17 2021 ca-certificates.conf
-rw-r--r-- 1 root root 5713 Apr 23 2020 ca-certificates.conf.dpkg-old
drwxr-xr-x 2 root root 4096 Apr 23 2020 calendar
```

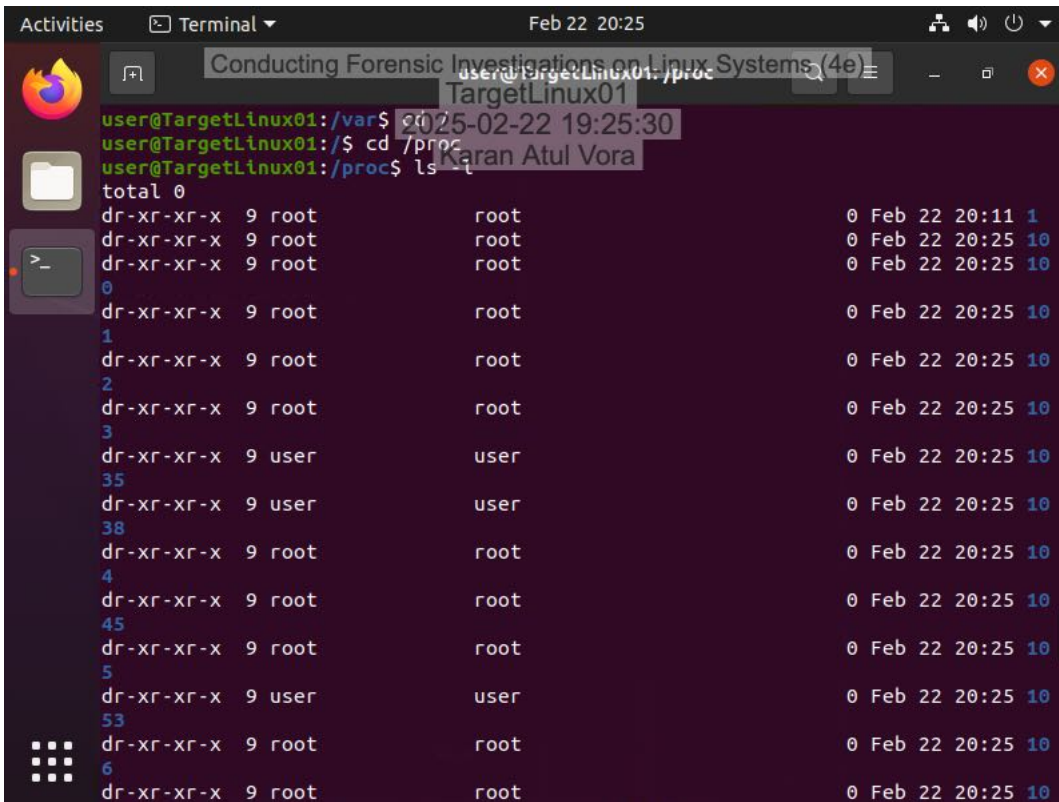
21. Make a screen capture showing the contents of the /var directory.



A terminal window titled "Terminal" with a date and time of "Feb 22 20:24". The window shows a user at "TargetLinux01" in the "/var" directory. The user has executed the command "ls -l", which displays a long listing of files and directories in /var. The output shows various files with permissions, owner, group, size, date, and name. Some files are highlighted in green in the original image. The user then navigates to the /etc directory and runs "cd /", then "cd /var", and finally "ls -l".

```
user@TargetLinux01: /var$ ls -l
total 48
drwxr-xr-x  4 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 Feb 22  20:11 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 14 root root    4096 Feb 22  20:18 snap
drwxr-xr-x  7 root root    4096 Aug 20  2021 spool
drwxr-xr-x  2 root root    4096 Feb 22  20:19 tmp
```

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



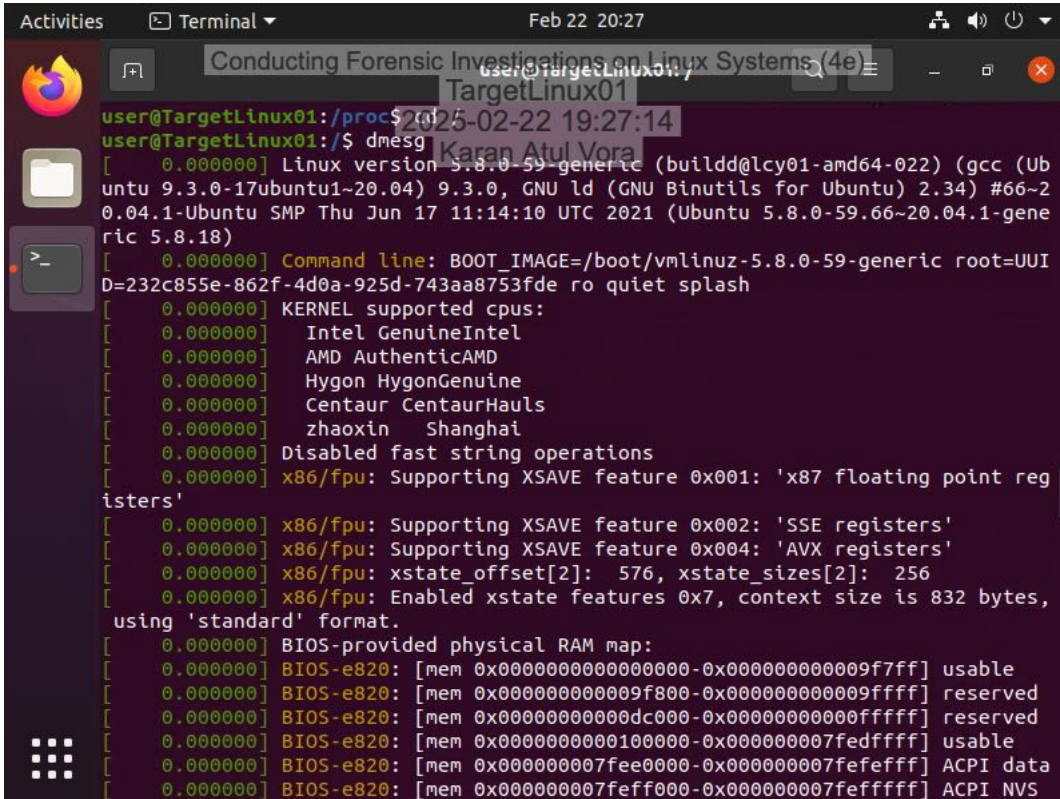
A terminal window titled "Terminal" showing the command prompt `user@TargetLinux01:/var$`. The user enters `cd /proc` and then `ls -lt`. The output shows a list of files in the `/proc` directory, including `total 0`, `dr-xr-xr-x 9 root root 0 Feb 22 20:11 1`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `0`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `1`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `2`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `3`, `dr-xr-xr-x 9 user user 0 Feb 22 20:25 10`, `35`, `dr-xr-xr-x 9 user user 0 Feb 22 20:25 10`, `38`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `4`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `45`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `5`, `dr-xr-xr-x 9 user user 0 Feb 22 20:25 10`, `53`, `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`, `6`, and `dr-xr-xr-x 9 root root 0 Feb 22 20:25 10`.

Part 2: Use Linux Shell Commands for Forensic Investigations

Conducting Forensic Investigations on Linux Systems (4e)

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

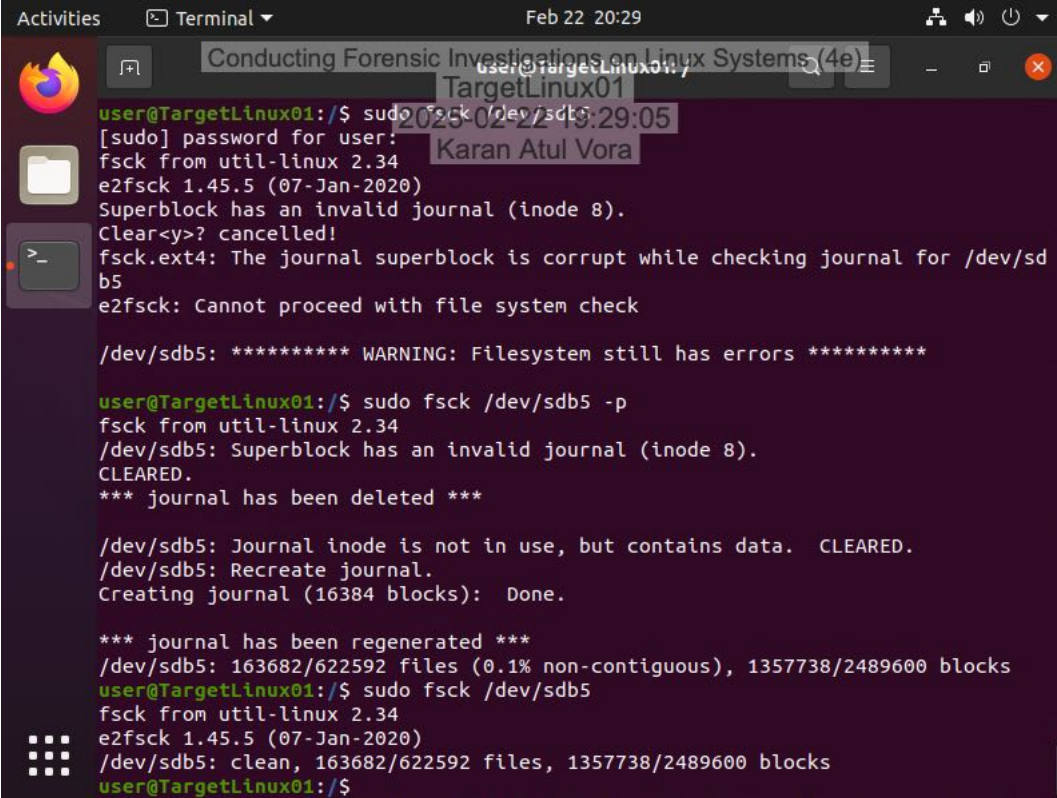
2. Make a screen capture showing the results of the `dmesg` command.



The screenshot shows a terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" with the subtitle "TargetLinux01". The terminal displays the output of the `dmesg` command, which shows system boot logs. The output includes the Linux version (5.8.0-59-generic), the command line, kernel supported CPUs (Intel GenuineIntel, AMD AuthenticAMD, Hygon HygonGenuine, Centaur CentaurHauls, zhaoxin Shanghai), disabled fast string operations, x86/fpu supporting XSAVE features (SSE registers, AVX registers), and BIOS-provided physical RAM map.

```
user@TargetLinux01:/proc$ cat /proc/cmdline
user@TargetLinux01:/$ dmesg
[ 0.000000] Linux version 5.8.0-59-generic (buildd@lcy01-amd64-022) (gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0, GNU ld (GNU Binutils for Ubuntu) 2.34) #66~20.04.1-Ubuntu SMP Thu Jun 17 11:14:10 UTC 2021 (Ubuntu 5.8.0-59.66~20.04.1-generic 5.8.18)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.8.0-59-generic root=UUID=232c855e-862f-4d0a-925d-743aa8753fde ro quiet splash
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Hygon HygonGenuine
[ 0.000000] Centaur CentaurHauls
[ 0.000000] zhaoxin Shanghai
[ 0.000000] Disabled fast string operations
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256
[ 0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009f7ff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009f800-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000dc000-0x00000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000007fedffff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000007fee0000-0x0000000007fefeffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x0000000007feff000-0x0000000007feffff] ACPI NVS
```

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

A terminal window titled "Terminal" with a date and time of "Feb 22 20:29". The window 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. A second fsck command with the -p flag is shown, which successfully recreates the journal and regenerates it. The final output shows the filesystem is clean.

```
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

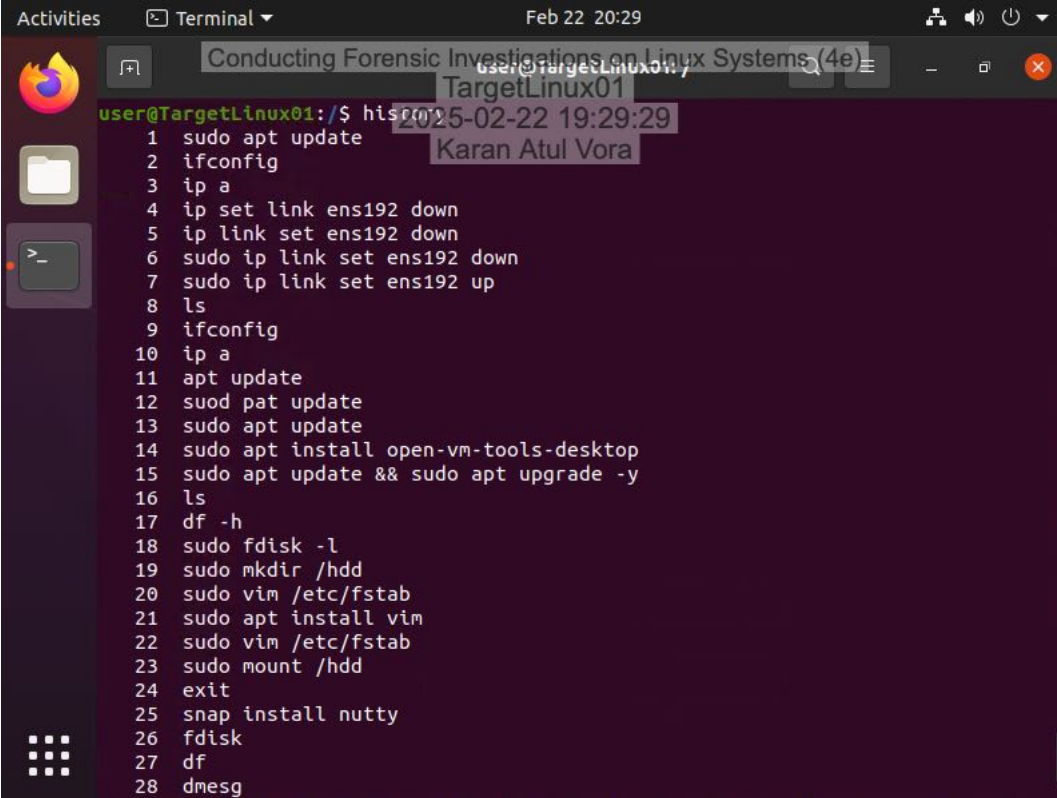
/dev/sdb5: ***** WARNING: Filesystem still has errors *****

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)
/dev/sdb5: clean, 163682/622592 files, 1357738/2489600 blocks
user@TargetLinux01:/$
```

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



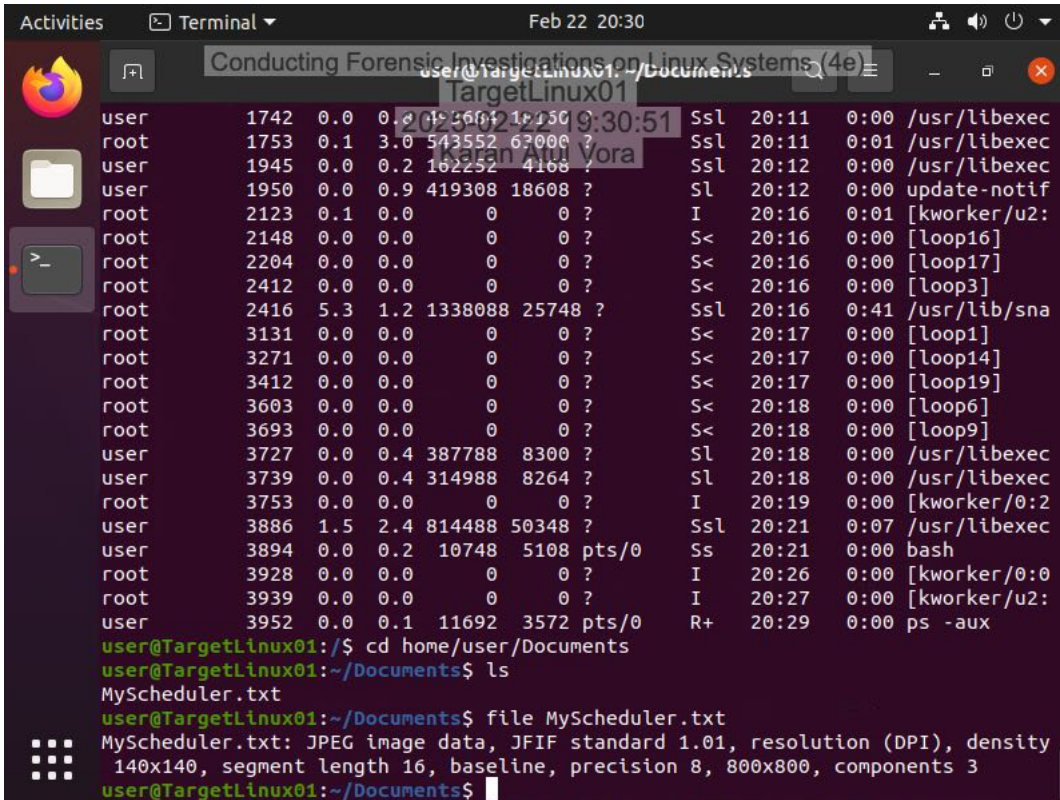
The screenshot shows a terminal window titled "Terminal" with the date and time "Feb 22 20:29". The terminal prompt is "user@TargetLinux01:". The user has entered the command "history", and the output is a list of 28 commands. The terminal window has a dark background and a light-colored text. The window title bar includes "Activities", "Terminal", and "Feb 22 20:29". The terminal content is as follows:

```
user@TargetLinux01:/$ history
1 sudo apt update
2 ifconfig
3 ip a
4 ip set link ens192 down
5 ip link set ens192 down
6 sudo ip link set ens192 down
7 sudo ip link set ens192 up
8 ls
9 ifconfig
10 ip a
11 apt update
12 suod pat update
13 sudo apt update
14 sudo apt install open-vm-tools-desktop
15 sudo apt update && sudo apt upgrade -y
16 ls
17 df -h
18 sudo fdisk -l
19 sudo mkdir /hdd
20 sudo vim /etc/fstab
21 sudo apt install vim
22 sudo vim /etc/fstab
23 sudo mount /hdd
24 exit
25 snap install nutty
26 fdisk
27 df
28 dmesg
```

11. Make a screen capture showing the running processes.

```
user@TargetLinux01:/$ ps
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.8  0.4 168968  9756 ?        Ss   20:11   0:09 /sbin/init s
root           2  0.0  0.0      0     0 ?        S    20:11   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   20:11   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   20:11   0:00 [rcu_par_gp]
root           6  0.0  0.0      0     0 ?        I<   20:11   0:00 [kworker/0:0
root           7  0.0  0.0      0     0 ?        I    20:11   0:00 [kworker/0:1
root           8  0.1  0.0      0     0 ?        I    20:11   0:01 [kworker/u2:
root           9  0.0  0.0      0     0 ?        I<   20:11   0:00 [mm_percpu_w
root          10  0.0  0.0      0     0 ?        S    20:11   0:00 [ksoftirqd/0
root          11  0.1  0.0      0     0 ?        I    20:11   0:01 [rcu_sched]
root          12  0.0  0.0      0     0 ?        S    20:11   0:00 [migration/0
root          13  0.0  0.0      0     0 ?        S    20:11   0:00 [idle_inject
root          14  0.0  0.0      0     0 ?        S    20:11   0:00 [cpuhp/0]
root          15  0.0  0.0      0     0 ?        S    20:11   0:00 [kdevtmpfs]
root          16  0.0  0.0      0     0 ?        I<   20:11   0:00 [netns]
root          17  0.0  0.0      0     0 ?        S    20:11   0:00 [rcu_tasks_k
root          18  0.0  0.0      0     0 ?        S    20:11   0:00 [rcu_tasks_r
root          19  0.0  0.0      0     0 ?        S    20:11   0:00 [rcu_tasks_t
root          20  0.0  0.0      0     0 ?        S    20:11   0:00 [kauditd]
root          21  0.0  0.0      0     0 ?        S    20:11   0:00 [khungtaskd]
root          22  0.0  0.0      0     0 ?        S    20:11   0:00 [oom_reaper]
root          23  0.0  0.0      0     0 ?        I<   20:11   0:00 [writeback]
root          24  0.0  0.0      0     0 ?        S    20:11   0:00 [kcompactd0]
root          25  0.0  0.0      0     0 ?        SN   20:11   0:00 [ksmd]
root          26  0.0  0.0      0     0 ?        SN   20:11   0:00 [khugepaged]
root          72  0.0  0.0      0     0 ?        I<   20:11   0:00 [kintegrityd
root          73  0.0  0.0      0     0 ?        I<   20:11   0:00 [kblockd]
```


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



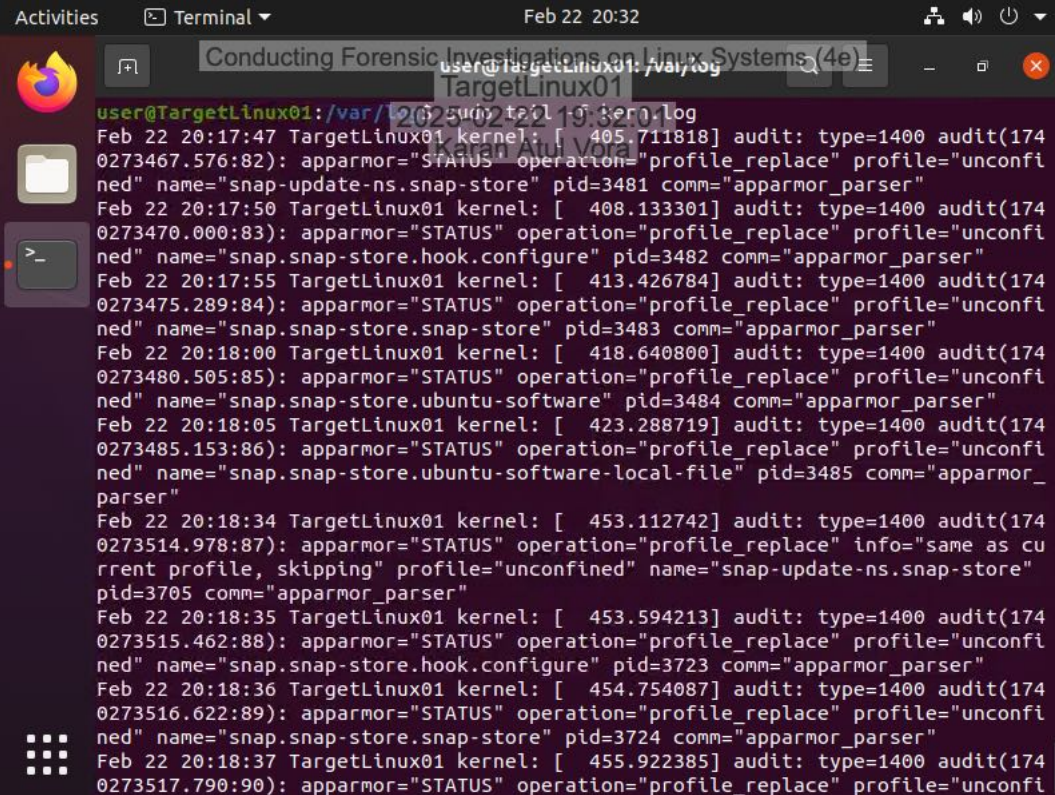
The screenshot shows a terminal window titled "Conducting Forensic Investigations on Linux Systems (4e)" with the hostname "TargetLinux01". The terminal displays the output of the 'ps -aux' command, showing various system processes. Below this, the user navigates to the directory ~/Documents and lists the files. The 'file' command is used to identify the file 'MyScheduler.txt', which is identified as a JPEG image.

```
user@TargetLinux01:~/Documents$ ps -aux
user      1742  0.0  0.2  15560 12050 ?        Ssl  20:11   0:00 /usr/libexec
root      1753  0.1  3.0  543552 63000 ?        Ssl  20:11   0:01 /usr/libexec
user      1945  0.0  0.2  162252 4168 ?        Ssl  20:12   0:00 /usr/libexec
user      1950  0.0  0.9  419308 18608 ?        Sl   20:12   0:00 update-notif
root      2123  0.1  0.0  0 0 ?        I    20:16   0:01 [kworker/u2:
root      2148  0.0  0.0  0 0 ?        S<   20:16   0:00 [loop16]
root      2204  0.0  0.0  0 0 ?        S<   20:16   0:00 [loop17]
root      2412  0.0  0.0  0 0 ?        S<   20:16   0:00 [loop3]
root      2416  5.3  1.2 1338088 25748 ?        Ssl  20:16   0:41 /usr/lib/sna
root      3131  0.0  0.0  0 0 ?        S<   20:17   0:00 [loop1]
root      3271  0.0  0.0  0 0 ?        S<   20:17   0:00 [loop14]
root      3412  0.0  0.0  0 0 ?        S<   20:17   0:00 [loop19]
root      3603  0.0  0.0  0 0 ?        S<   20:18   0:00 [loop6]
root      3693  0.0  0.0  0 0 ?        S<   20:18   0:00 [loop9]
user      3727  0.0  0.4  387788 8300 ?        Sl   20:18   0:00 /usr/libexec
user      3739  0.0  0.4  314988 8264 ?        Sl   20:18   0:00 /usr/libexec
root      3753  0.0  0.0  0 0 ?        I    20:19   0:00 [kworker/0:2
user      3886  1.5  2.4  814488 50348 ?        Ssl  20:21   0:07 /usr/libexec
user      3894  0.0  0.2  10748 5108 pts/0    Ss   20:21   0:00 bash
root      3928  0.0  0.0  0 0 ?        I    20:26   0:00 [kworker/0:0
root      3939  0.0  0.0  0 0 ?        I    20:27   0:00 [kworker/u2:
user      3952  0.0  0.1  11692 3572 pts/0    R+   20:29   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$
```

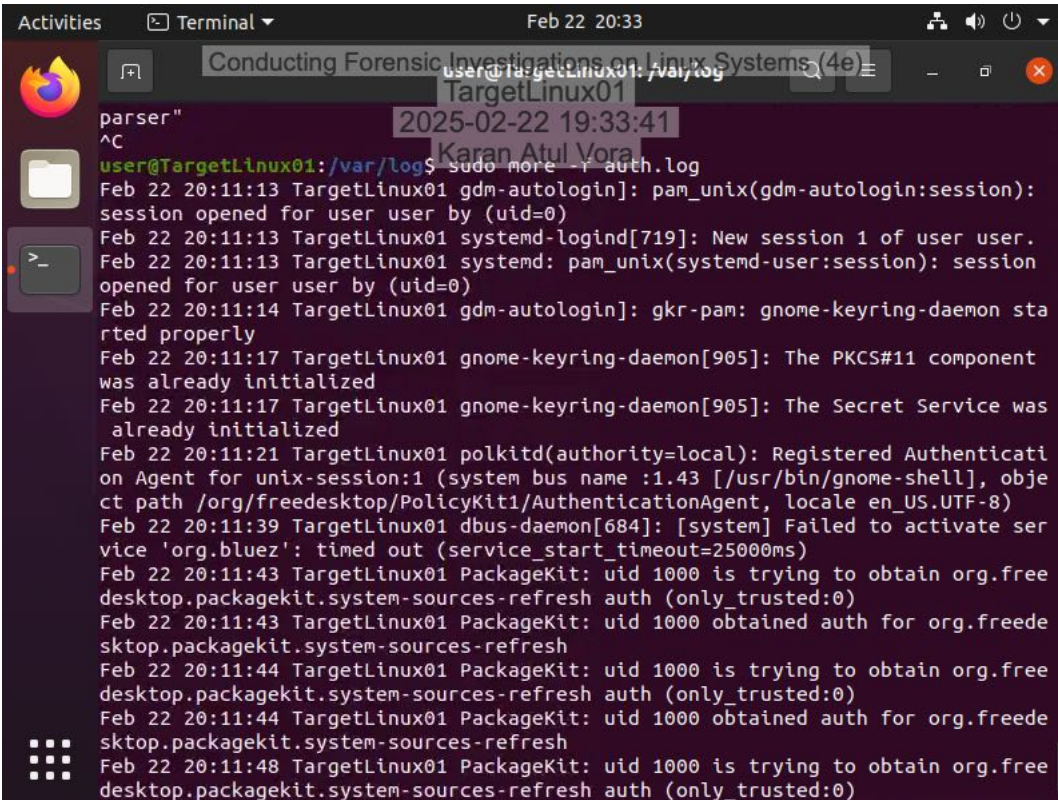
Part 3: Retrieve Logs Files on a Live Linux System

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



```
user@TargetLinux01: /var/log
cat /var/log/kern.log
Feb 22 20:17:47 TargetLinux01 kernel: [ 405.711818] audit: type=1400 audit(174
0273467.576:82): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap-update-ns.snap-store" pid=3481 comm="apparmor_parser"
Feb 22 20:17:50 TargetLinux01 kernel: [ 408.133301] audit: type=1400 audit(174
0273470.000:83): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.hook.configure" pid=3482 comm="apparmor_parser"
Feb 22 20:17:55 TargetLinux01 kernel: [ 413.426784] audit: type=1400 audit(174
0273475.289:84): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.snap-store" pid=3483 comm="apparmor_parser"
Feb 22 20:18:00 TargetLinux01 kernel: [ 418.640800] audit: type=1400 audit(174
0273480.505:85): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software" pid=3484 comm="apparmor_parser"
Feb 22 20:18:05 TargetLinux01 kernel: [ 423.288719] audit: type=1400 audit(174
0273485.153:86): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.ubuntu-software-local-file" pid=3485 comm="apparmor_
parser"
Feb 22 20:18:34 TargetLinux01 kernel: [ 453.112742] audit: type=1400 audit(174
0273514.978:87): apparmor="STATUS" operation="profile_replace" info="same as cu
rrent profile, skipping" profile="unconfined" name="snap-update-ns.snap-store"
pid=3705 comm="apparmor_parser"
Feb 22 20:18:35 TargetLinux01 kernel: [ 453.594213] audit: type=1400 audit(174
0273515.462:88): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.hook.configure" pid=3723 comm="apparmor_parser"
Feb 22 20:18:36 TargetLinux01 kernel: [ 454.754087] audit: type=1400 audit(174
0273516.622:89): apparmor="STATUS" operation="profile_replace" profile="unconfi
ned" name="snap.snap-store.snap-store" pid=3724 comm="apparmor_parser"
Feb 22 20:18:37 TargetLinux01 kernel: [ 455.922385] audit: type=1400 audit(174
0273517.790:90): apparmor="STATUS" operation="profile_replace" profile="unconfi
```

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



A terminal window titled "Terminal" with a date and time of "Feb 22 20:33". The window shows the command `user@TargetLinux01: /var/log$ sudo more -f auth.log` and its output. The output displays system logs from the `auth.log` file, including session openings for user 'user', systemd-logind session creation, gdm-autologin messages, gnome-keyring-daemon initialization, polkitd authentication agent registration, and PackageKit authentication attempts. The logs are timestamped with "Feb 22 20:11:13", "Feb 22 20:11:14", "Feb 22 20:11:17", "Feb 22 20:11:21", "Feb 22 20:11:39", "Feb 22 20:11:43", and "Feb 22 20:11:44".

```
parser"
^C
user@TargetLinux01: /var/log$ sudo more -f auth.log
2025-02-22 19:33:41
Karan Atul Vora
Feb 22 20:11:13 TargetLinux01 gdm-autologin]: pam_unix(gdm-autologin:session):
session opened for user user by (uid=0)
Feb 22 20:11:13 TargetLinux01 systemd-logind[719]: New session 1 of user user.
Feb 22 20:11:13 TargetLinux01 systemd: pam_unix(systemd-user:session): session
opened for user user by (uid=0)
Feb 22 20:11:14 TargetLinux01 gdm-autologin]: gkr-pam: gnome-keyring-daemon sta
rted properly
Feb 22 20:11:17 TargetLinux01 gnome-keyring-daemon[905]: The PKCS#11 component
was already initialized
Feb 22 20:11:17 TargetLinux01 gnome-keyring-daemon[905]: The Secret Service was
already initialized
Feb 22 20:11:21 TargetLinux01 polkitd(authority=local): Registered Authenticati
on Agent for unix-session:1 (system bus name :1.43 [/usr/bin/gnome-shell], obje
ct path /org/freedesktop/PolicyKit1/AuthenticationAgent, locale en_US.UTF-8)
Feb 22 20:11:39 TargetLinux01 dbus-daemon[684]: [system] Failed to activate ser
vice 'org.bluez': timed out (service_start_timeout=25000ms)
Feb 22 20:11:43 TargetLinux01 PackageKit: uid 1000 is trying to obtain org.free
desktop.packagekit.system-sources-refresh auth (only_trusted:0)
Feb 22 20:11:43 TargetLinux01 PackageKit: uid 1000 obtained auth for org.freede
sktop.packagekit.system-sources-refresh
Feb 22 20:11:44 TargetLinux01 PackageKit: uid 1000 is trying to obtain org.free
desktop.packagekit.system-sources-refresh auth (only_trusted:0)
Feb 22 20:11:44 TargetLinux01 PackageKit: uid 1000 obtained auth for org.freede
sktop.packagekit.system-sources-refresh
Feb 22 20:11:48 TargetLinux01 PackageKit: uid 1000 is trying to obtain org.free
desktop.packagekit.system-sources-refresh auth (only_trusted:0)
```

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.

neol, 12, Jun 11 00:57:17 - 05:06:50, 192.168.78.1, 14444Dominic, 11, Jun 11 05:07:29 - 05:38:32, 192.168.78.1,3417

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

Jan 11 05:23:03 for user Dominic

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.

logkeys

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

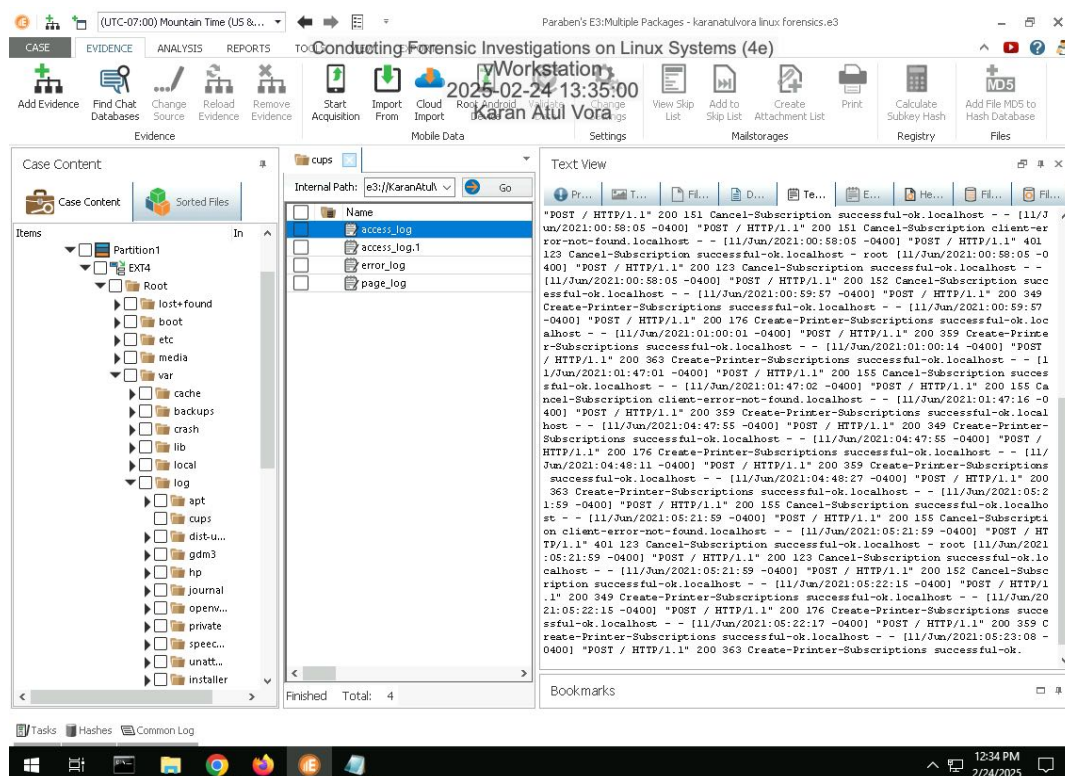
4. **Document** when the USB storage device was connected and its serial number.

Jun 10 10:24:12 FBI1405291710344

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.

