

## Activity 3: Linux Data Acquisition

### Introduction

Data Acquisition is the process of copying data. For digital forensics, it's the task of collecting digital evidence from electronic media. There are two types of data acquisition: static acquisition and live acquisition. In this practice, you'll perform a static data acquisition using the Linux `dd` and `dcfldd` commands. Operations in this guide are performed in CAINE Linux. You may also use Kali Linux for the same practice.

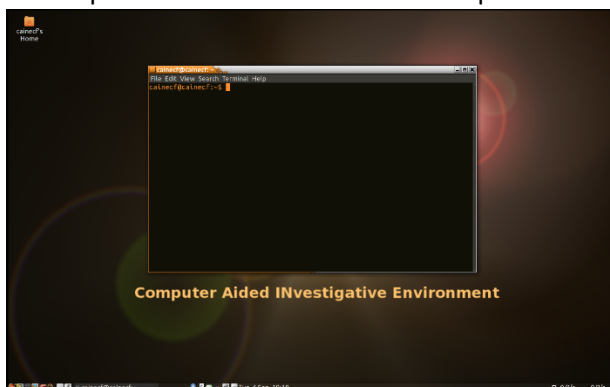
### Objectives

- Get familiar with the data acquisition process.
- Prepare a target drive for data acquisition.
- Use Linux data acquisition tool to acquire data from a USB drive.
- Validate the acquired data.

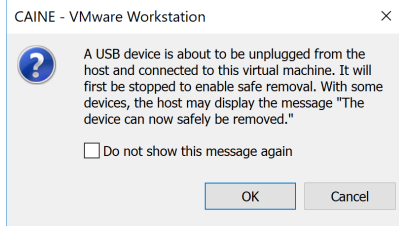
### Tasks

#### Part1. Prepare the target drive

1. In CAINE, Find terminal in Main->system tools->MATE Terminal. In Kali Linux, Find the terminal on the panel to the left of the screen. Open the terminal.



2. To change to root account, type command ***su***  
And provide the password.
3. Type ***fdisk -l*** to show the current disks.
4. Plug the USB (target disk where the image will be stored) to the host.
5. Then you need to connect the USB to the Linux virtual machine. In vmware workstation, click on VM->Removable Devices->Flash Disk->Connect



6. Type ***fdisk -l*** again. The extra drive showed now (`/dev/sdb` in this case) is the target disk drive where the image of original disk will be stored on.

```
root@cainecf:/home/cainecf# fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xa209faae

Device Boot Start End Sectors Size Id Type
/dev/sda1 2048 41943039 41940992 20G 83 Linux
root@cainecf:/home/cainecf# fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xa209faae

Device Boot Start End Sectors Size Id Type
/dev/sda1 2048 41943039 41940992 20G 83 Linux

Disk /dev/sdb: 961 MiB, 1007681536 bytes, 1968128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x00000000

Device Boot Start End Sectors Size Id Type
/dev/sdb1 32 1968127 1968096 961M 6 FAT16
root@cainecf:/home/cainecf#
```

7. Zero out the target drive:

***dd if=/dev/zero of=/dev/sdb***

This will take some time depending on the virtual machine configuration and the usb data transfer speed. Please be patient as it is writing bit by bit. The time can vary from seconds to half an hour. *Make sure you are zeroing out the target drive, not the original drive or other drives!*

```
root@cainecf:/home/cainecf# dd if=/dev/zero of=/dev/sdb
dd: writing to '/dev/sdb': No space left on device
1968128+0 records in
1968128+0 records out
1007681536 bytes (1.0 GB, 961 MiB) copied, 2736.91 s, 368 kB/s
root@cainecf:/home/cainecf#
```

8. Type ***fdisk /dev/sdb*** to begin creating a partition on the target drive.

```
root@cainecf:/home/cainecf# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.27.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognised partition table.
Created a new DOS disklabel with disk identifier 0x735a3bc4.

Command (m for help):
```

- a. Then type ***m*** to show the menu
- b. Type ***p*** to print the partition table and see if there are any partitions on `/dev/sdb`  
If there are no partitions on target drive, the output will sometimes be similar to:

Disk */dev/sdb* doesn't contain a valid partition table

Or the output simply doesn't show any partition information.

Then you'll need to create a new partition following the steps below.

- c. Type **n** and hit enter, to create a new partition. It lists two partition types: primary and extended.
- d. Type **p** and hit enter, to choose a primary partition table.
- e. Type **1** and hit enter, to select the first partition
- f. After the new partition is created, type **p** again to show current partitions on */dev/sdb* and you'll see the newly created partition.

```
root@calnecf:/home/calnecf# fdisk
File Edit View Search Terminal Help
O dump disk layout to sfdisk script file

Save & Exit
w write table to disk and exit
q quit without saving changes

Create a new label
g create a new empty GPT partition table
G create a new empty SGI (IRIX) partition table
o create a new empty DOS partition table
s create a new empty Sun partition table

Command (m for help): p
Disk /dev/sdb: 361 MiB, 3687681536 bytes, 1968128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x735a3b0d

Command (m for help): n
Partition type
p primary (0 primary, 0 extended, 4 free)
e extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-1968127, default 2048):
Last sector, +sectors or +size(K,M,G,T,P) (2048-1968127, default 1968127):
Created a new partition 1 of type 'Linux' and of size 960 MiB.

Command (m for help): p
Disk /dev/sdb: 361 MiB, 3687681536 bytes, 1968128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x735a3b0d

Device Boot Start End Sectors Size Id Type
/dev/sdb1 2048 1968127 1966080 960M 83 Linux

Command (m for help):
```

9. Last step creates a new linux partition */dev/sdb1*. This step changes the newly created partition to windows 95 FAT32 file system.
  - a. Type **m** to show the menu
  - b. Type **t** to change the partition type
  - c. Type **l** (lowercase L) to show available file systems and their code values

```
root@calnecf:/home/calnecf# fdisk
File Edit View Search Terminal Help
l Load disk layout from sfdisk script file
O dump disk layout to sfdisk script file

Save & Exit
w write table to disk and exit
q quit without saving changes

Create a new label
g create a new empty GPT partition table
G create a new empty SGI (IRIX) partition table
o create a new empty DOS partition table
s create a new empty Sun partition table

Command (m for help): t
Selected partition 1
Partition type (type L to list all types): l

0 Empty 24 NEC DOS 81 Mlnix / old Ltn bf Solaris
1 FAT12 27 Hidden NTFS Win 82 Linux swap / So c1 DRDOS/sec (FAT-
2 XENIX root 30 Plan 9 83 Linux c4 DRDOS/sec (FAT-
3 XENIX usr 3c PartitionMagic 84 OS/2 hidden or c6 DRDOS/sec (FAT-
4 FAT16 <32M 40 Venix 80286 85 Linux extended c7 Syrnix
5 Extended 41 PPC PReP Boot 86 NTFS volume set da Non-FS data
6 FAT16 42 SFS 87 NTFS volume set db CP/M / CTOS / .
7 HPFS/NTFS/exFAT 4d QNX4.x 88 Linux plaintext de Dell Utility
8 AIX 4e QNX4.x 2nd part 8e Linux LVM df BootIt
9 AIX bootable 4f QNX4.x 3rd part 93 Amoeba e1 DOS access
a OS/2 Boot Manag 50 OnTrack DM 94 Amoeba BBT e3 DOS R/O
b W95 FAT32 51 OnTrack DM6 Aux 9f BSD/OS e4 SpeedStor
c W95 FAT32 (LBA) 52 CP/M a0 IBM Thinkpad hi ea Rufus alignment
e W95 FAT16 (LBA) 53 OnTrack DM6 Aux a5 FreeBSD eb BeOS fs
f W95 Ext'd (LBA) 54 OnTrackDM6 a6 OpenBSD ee GPT
10 OPUS 55 EZ-Drive a7 NeXTSTEP ef EFI (FAT-12/16/
11 Hidden FAT12 56 Golden Bow a8 Darwin UFS f0 Linux/PA-RISC b
12 Compaq diagnost 5c Priam Edisk a9 NetBSD f1 SpeedStor
14 Hidden FAT16 <3 61 SpeedStor ab Darwin boot f4 SpeedStor
16 Hidden FAT16 63 GNU HURD or Sys af HFS / HFS+ f2 DOS secondary
17 Hidden HPFS/NTF 64 Novell Netware b7 BSDI fs fb VMware VMFS
18 AST SmartSleep 65 Novell Netware b8 BSDI swap fc VMware VMKCORE
1b Hidden W95 FAT3 70 DiskSecure Mult bb Boot Wizard hid fd Linux RAID auto
1c Hidden W95 FAT3 75 PC/IX bc Acronis FAT32 L fe LANstep
1e Hidden W95 FAT1 80 Old Minix be Solaris boot ff BBT

Partition type (type L to list all types):
```

- d. Type **c** to change the partition to Windows 95 FAT32(LBA)
- e. Type **p** to display the newly changed drive

- f. Type **w** to save/write the newly created partition to the **/dev/sdb** drive.

```
Partition type (type L to list all types): c
Changed type of partition 'Linux' to 'W95 FAT32 (LBA)'.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Synching disks.

root@cainecf:/home/cainecf#
```

10. Type **fdisk -l** to see the drives again.

11. To format a FAT file system from Linux, type **mkfs.msdos -vF32 /dev/sdb1**

- a. **-v** means verbose execution
- b. **F32** means the type of file allocation table used is 32 bit (it can be 12, 16 or 32 bit).

```
root@cainecf:/home/cainecf# fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xa209faae

Device      Boot Start      End  Sectors  Size Id Type
/dev/sda1                2048  41943039  41940992   20G 83 Linux

Disk /dev/sdb: 961 MiB, 1007681536 bytes, 1968128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x735a3bc4

Device      Boot Start      End  Sectors  Size Id Type
/dev/sdb1                2048 1968127 1966080   960M  c W95 FAT32 (LBA)
root@cainecf:/home/cainecf# mkfs.msdos -vF32 /dev/sdb1
mkfs.fat 3.0.28 (2015-05-16)
/dev/sdb1 has 31 heads and 62 sectors per track,
hidden sectors 0x0800;
logical sector size is 512;
using 0xf8 media descriptor, with 1966080 sectors;
drive number 0x80;
filesystem has 2 32-bit FATs and 8 sectors per cluster.
FAT size is 1917 sectors, and provides 245276 clusters.
There are 32 reserved sectors.
Volume ID is ebfbddcf, no volume label.
root@cainecf:/home/cainecf#
```

## Part 2. Perform Data Acquisition

12. Insert the second USB drive as the *source drive*, type **fdisk -l** to show all disks and identify which one is the source drive and which one is target drive. The next step is to perform data acquisition towards the source drive. In this case, **/dev/sdc1** is the source drive.

```

root@cainecf:/home/cainecf# fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xa209faae

Device      Boot Start      End  Sectors  Size Id Type
/dev/sda1                2048 41943039 41940992  20G 83 Linux

Disk /dev/sdb: 961 MiB, 1007681536 bytes, 1968128 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x735a3bc4

Device      Boot Start      End  Sectors  Size Id Type
/dev/sdb1                2048 1968127 1966080  960M  c W95 FAT32 (LBA)

Disk /dev/sdc: 501.5 MiB, 525860864 bytes, 1027072 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x000d27cc

Device      Boot Start      End  Sectors  Size Id Type
/dev/sdc1                32 1026559 1026528 501.2M  6 FAT16
root@cainecf:/home/cainecf#

```

13. Type **ls /** to show files under /.
14. Type **ls /mnt** to show files under /mnt.
15. Type **mkdir /mnt/sdb1** to create a folder under /mnt.
16. Type **ls /mnt/sdb1** again to show files under /mnt/sdb1.
17. Type **mount -t vfat /dev/sdb1 /mnt/sdb1** to mount the target drive partition.
  - a. -t means the file system type is vfat.
18. Type **cd /mnt/sdb1** to change to default directory to target drive
19. Type **ls -al** to show the contents of the target drive's root level.
20. Type **mkdir case1** to create a target directory.
21. Type **ls** to confirm that the new directory has been created.
22. Type **md5sum /dev/sdc1 |tee /mnt/sdb1/case1/pre-imagesource.md5.txt**
  - a. Md5sum calculate the hash of the source drive using MD5 algorithm
  - b. |tee means the output is added to the txt file and also displayed in terminal

```

root@cainecf:/home/cainecf# ls /
bin boot cdrom dev etc home initrd.img lib lib64 lost+found media mnt opt proc root run sbin snap srv sys usr var vmlinuz
root@cainecf:/home/cainecf# ls /mnt
root@cainecf:/home/cainecf# mkdir /mnt/sdb1
root@cainecf:/home/cainecf# ls /mnt
sdb1
root@cainecf:/home/cainecf# mount -t vfat /dev/sdb1 /mnt/sdb1
root@cainecf:/mnt/sdb1# cd /mnt/sdb1
root@cainecf:/mnt/sdb1# ls -al
total 8
drwxr-xr-x 2 root root 4096 Jan  1 1970 .
drwxr-xr-x 3 root root 4096 Sep  5 05:36 ..
root@cainecf:/mnt/sdb1# ls
root@cainecf:/mnt/sdb1# mkdir case1
root@cainecf:/mnt/sdb1# ls
case1
root@cainecf:/mnt/sdb1# md5sum /dev/sdc1 |tee /mnt/sdb1/case1/pre-imagesource.md5.txt
c92f5d69278d4413871f027f0613be69 /dev/sdc1
root@cainecf:/mnt/sdb1#

```

23. To acquire data from the source drive /dev/sdc1, type
 

```
dcflddd if=/dev/sdc1 of=/mnt/sdb1/case1/image1.dd conv=noerror,sync hash=md5 hashwindow=0 hashlog=/mnt/sdb1/case1/post-imagesource.md5.txt
```

```
root@caineef:/mnt/sdb1# md5sum /dev/sdc1 |tee /mnt/sdb1/case1/pre-imagesource.md5.txt
c92f5d69278d4413871f027f0613be69 /dev/sdc1
root@caineef:/mnt/sdb1# dcfldd if=/dev/sdc1 of=/mnt/sdb1/case1/image1.dd conv=noerror,sync hash=md5 hashwindow=0 hashlog=/mnt/sdb1/case1/post-imagesource.md5.txt
15872 blocks (496Mb) written.
16039+1 records in
16040+0 records out
root@caineef:/mnt/sdb1# cat /mnt/sdb1/case1/pre-imagesource.md5.txt
c92f5d69278d4413871f027f0613be69 /dev/sdc1
root@caineef:/mnt/sdb1# cat /mnt/sdb1/case1/post-imagesource.md5.txt
Total (md5): c92f5d69278d4413871f027f0613be69
root@caineef:/mnt/sdb1#
```

- a. To create segmented volumes of 2MB each, you may use  
***dcfldd if=/dev/sdc1 split=2M of=/mnt/sdb1/case1/image1.dd conv=noerror,sync hash=md5 hashwindow=0 hashlog=/media/sdb1/case1/post-imagesource.md5.txt***
- b. If use dd command for data acquisition, the command should be  
***dd if=/dev/sdc1 |split -b 2m - image\_sdc***  
if you want to create multiple segments.  
Otherwise you can simply use  
***dd if=/dev/sdc1 of=/mnt/sdb1/case1/image1-dd.dd***  
In my demo, my target drive does not have enough disk storage to store the target image. You can read the number of blocks and bytes copied.

```
root@caineef:/mnt/sdb1/case1# dd if=/dev/sdc1 of=/mnt/sdb1/case1/image1-dd.dd
dd: writing to '/mnt/sdb1/case1/image1-dd.dd': No space left on device
935617+0 records in
935616+0 records out
479035392 bytes (479 MB, 457 MiB) copied, 890.542 s, 538 kB/s
root@caineef:/mnt/sdb1/case1#
```

### Part 3. Validate the acquired data

24. To validate the acquired data, there are two ways if using *dcfldd* for data acquisition:
  - a. Type  
***md5sum /dev/sdc1 |tee /mnt/sdb1/case1/pre-imagesource.md5.txt***  
to generate the hash of the original source drive;  
Type  
***cat /mnt/sdb1/case1/pre-imagesource.md5.txt***  
to show the hash value of original source drive;  
The hash of the acquired image is already computed and put into post-imagesource.md5.txt. Type  
***cat /mnt/sdb1/case1/post-imagesource.md5.txt***  
to show the hash value and then compare the value in pre-imagesource.md5.txt
  - b. Type ***dcfldd if=/dev/sdc1 vf=/mnt/sdb1/case1/image1.dd*** (Note: this only applies to the nonsegmented image file).
25. To validate the acquired data if using *dd* for data acquisition:
  - a. Type ***md5sum /dev/sdc1 |tee /mnt/sdb1/case1/pre-imagesource.md5.txt***
  - b. Type ***cat image\_sdc.\*|md5sum > post-imagesource.md5.txt***

And then compare the values in both files to see if they are the same. This applies to the segmented image file too.

***Post-Activity questions.***

***Please complete this exercise and submit the PDF to Canvas.***

1. What are the two broad categories of acquisition?
2. What is a live storage acquisition and when is it used?
3. Which command should be used to check the disks available on the current system? You only need to state the command name, not the entire command string.
4. The mkfs -t command does what?
5. Which drive should be 'zeroed out', the source evidence drive or the target drive?
6. What is the purpose of 'zeroing out' before a storage acquisition is performed?
7. When you issue the command the command  
***dd if=/dev/zero of=/dev/sdb***  
What does the string "/dev/sdb" represent?
8. The md5sum /dev/sda command does what? Why is it used?
9. How many times should the ***md5sum*** command be used at least in one acquisition?
10. Instead of using "dd", what other commands can you use to perform data acquisition in Linux?