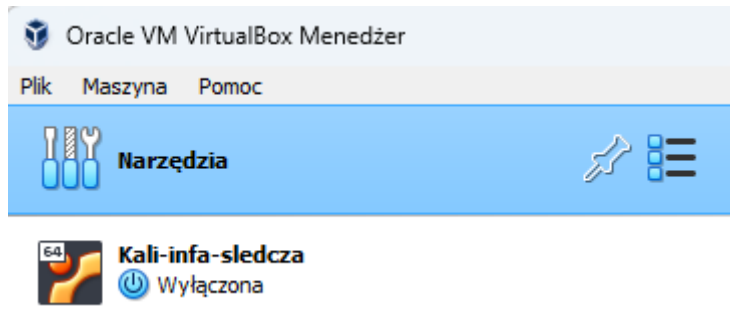


Informatyka śledcza Laboratorium nr 1

Raport – Nikodem Jakubowski

Zadanie 1 – Instalacja środowiska wirtualnego (Kali/SIFT lub innej dystrybucji Linux) oraz pobranie pliku z rozszerzeniem E01.



Na powyższym zrzucie ekranu widać maszynę wirtualną Kali Linux.

Zadanie 2 – Analiza pobranego obrazu (plik .E01).

1. Jaka jest wartość skrótu dla funkcji haszującej md5 i sha-1?

```
(user@user) - [~/Desktop]
$ md5sum USB_4GB_Kingston.E01
b879553c628b3308d624372398d8302a  USB_4GB_Kingston.E01
```

Wartość funkcji skrótu dla md5.

```
(user@user) - [~/Desktop]
$ sha1sum USB_4GB_Kingston.E01
344aa2b0179e18ad94ddcc0e5cbfa0af663faba3  USB_4GB_Kingston.E01
```

Wartość funkcji skrótu dla sha-1.

Proszę przy pomocy polecenia „mmls” o wyświetlenie i podanie odpowiedzi na pytania:

```
(user@user) - [~/Desktop]
$ mmls USB_4GB_Kingston.E01
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

   Slot   Start      End      Length  Description
000:  Meta    00000000000  00000000000  00000000001  Primary Table (#0)
001:  _____ 00000000000  00000000127  00000000128  Unallocated
002:  000:000  00000000128  0007581695   0007581568  Win95 FAT32 (0x0c)
```

Rysunek 1 funkcja mmls - USB

Użycie funkcji mmls.

1. W jakim przedziale sektorów znajduje się niealokowana pamięć?

```
(user@user)-[~/Desktop]
$ mmls -A USB_4GB_Kingston.E01

DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

Slot      Start      End      Length    Description
001:      0000000000 0000000127 0000000128 Unallocated
```

Niealokowana pamięć znajduje się w sektorze: od 0000000000 do 0000000127.

2. W której partycji znajdują się pliki systemowe?

Pliki systemowe znajdują się w partycji 002, o czym świadczy opis: Win95 FAT32 (0x0c).
(Rysunek 1 funkcja mmls - USB).

3. Proszę o podanie początku i końca sektora należącego do partycji Win95?

Początek sektora partycji Win 95 to 0000000128, a koniec 0007581695.

Przy pomocy narzędzia „fsstat” proszę o wyświetlenie i odpowiedź na pytania:

1. Jaki system plików zaczyna się w sektorze 0000000128 analizowanego pliku?

```
(user@user)-[~/Desktop]
$ fsstat -o 128 USB_4GB_Kingston.E01
FILE SYSTEM INFORMATION

File System Type: FAT32
```

Jest to FAT32.

2. Jaka jest wielkość sektora oraz klastra w badanym obszarze?

```
(user@user)-[~/Desktop]
$ fsstat -o 128 USB_4GB_Kingston.E01
FILE SYSTEM INFORMATION

File System Type: FAT32

OEM Name: MSDOS5.0
Volume ID: 0x779c953c
Volume Label (Boot Sector): USB DISK
Volume Label (Root Directory):
File System Type Label: FAT32
Next Free Sector (FS Info): 11392
Free Sector Count (FS Info): 7504624

Sectors before file system: 128

File System Layout (in sectors)
Total Range: 0 - 7581567
* Reserved: 0 - 47
** Boot Sector: 0
** FS Info Sector: 1
** Backup Boot Sector: 8
* FAT 0: 48 - 3751
* FAT 1: 3752 - 7455
* Data Area: 7456 - 7581567
** Cluster Area: 7456 - 7581567
*** Root Directory: 7456 - 7471
```

Wielkość sektora wynosi: 7581568 oraz wielkość klastra w badanym obszarze to: 7574111.

Narzędzie fls posiada funkcje umożliwiającą wyświetlanie informacji o plikach znajdujących się w partycji Win95.

1. Wypisz wszystkie pliki głównego katalogu USB_4GB_Kingston.E01.

```
(user@user)-[~/Desktop]
$ fls -a -o 0000000128 USB_4GB_Kingston.E01
r/r 3:  USB DISK      (Volume Label Entry)
d/d 6:  .Spotlight-V100
d/d * 8:      .fseventsd
d/d 9:  1
r/r 10: IMG_5609.JPG
r/r * 13:      .IMG_5609.JPG
r/r 14: IMG_5627.JPG
r/r * 17:      .IMG_5627.JPG
r/r 18: IMG_5753.JPG
r/r * 21:      .IMG_5753.JPG
r/r 22: IMG_6002.JPG
r/r * 25:      .IMG_6002.JPG
r/r 26: IMG_8064.JPG
r/r * 29:      .IMG_8064.JPG
r/r 30: text2.rar
r/r * 32:      .text2.rar
r/r * 34:      .1
v/v 121185795: $MBR
v/v 121185796: $FAT1
v/v 121185797: $FAT2
V/V 121185798: $OrphanFiles
```

Są to: IMG_5609.JPG, IMG_5627.JPG, IMG_5753.JPG, IMG_6002.JPG, IMG_8064.JPG, text2.rar.

2. Wypisz wszystkie plik znajdujące się w folderze „1”.

```
(user@user)-[~/Desktop]
$ fls -r -o 0000000128 USB_4GB_Kingston.E01 9
r/r 62725:      IMG_6110.JPG
r/r 62726:      IMG_5592.JPG
r/r 62727:      text.txt
```

Są to: IMG_6110.JPG, IMG_5592.JPG, text.txt.

Przy użyciu funkcji znajdujących się w EWFTools/ewfinfo wyświetl informacje o pliku oraz odpowiedz na poniższe pytania.

```
(user@user)-[~/Desktop]
$ ewfinfo USB_4GB_Kingston.E01
ewfinfo 20140814

Acquiry information
Case number:          001
Examiner name:        Kali
Evidence number:      001
Acquisition date:     Sun Oct  3 16:31:05 2021
System date:          Sun Oct  3 16:31:05 2021
Operating system used: Linux
Software version used: 20140807
Password:             N/A
Model:                USB DISK 2.0
Serial number:        0D7117891080

EWF information
File format:          EnCase 6
Sectors per chunk:    64
Error granularity:    64
Compression method:   deflate
Compression level:    good (fast) compression

Media information
Media type:            removable disk
Is physical:          yes
Bytes per sector:      512
Number of sectors:     7581696
Media size:            3.6 GiB (3881828352 bytes)

Digest hash information
MD5:                  5df8f604967c556c810d21dd664ceae4
```

1. Pod jaki numer sprawy podlega badany nośnik?

Pod numer 001.

2. Jaka jest nazwa osoby tworzącej obraz dysku?

Nazwa osoby to: Kali.

3. Kiedy plik został utworzony?

Został utworzony w Sun Oct 3 16:31:05 2021 (Niedziela, 3 października 2021 o 16:31:05).

4. Numer seryjny fizycznego dysku oraz nazwa modelu?

Model: USB DISK 2.0. Numer seryjny: 0D7117891080.

5. Wskaż format plików?

Format pliku to: EnCase 6 (format dla przejętych danych).

6. Proszę o podanie metody kompresji pliku?

Użyta metoda kompresji to: deflate.

7. Jaka jest pełna wielkość badanego nośnika (w bajtach)?

Wielkość medium: 3.6 GiB (3881828352 bajtów).

8. Jaki poziom kompresji został wskazany przy tworzeniu pliku?

Poziom kompresji to: good (fast).

Zadanie 3 – Analiza pobranego obrazu (LAB_1.img).

1. Wczytaj za pomocą polecenia mmls i podaj liczbę sektorów

gpt_load_table.

```
(user@user)-[~/Desktop]
$ mmls -v LAB_1.img
tsk_img_open: Type: 0 NumImg: 1 Img1: LAB_1.img
aff_open: Error determining type of file: LAB_1.img
aff_open: No such file or directory
Error opening vmdk file
Error checking file signature for vhd file
tsk_img_findFiles: LAB_1.img found
tsk_img_findFiles: 1 total segments found
raw_open: segment: 0 size: 1024000000 max offset: 1024000000 path: LAB_1.img
dos_load_prim: Table Sector: 0
raw_read: byte offset: 0 len: 65536
raw_read: found in image 0 relative offset: 0 len: 65536
raw_read_segment: opening file into slot 0: LAB_1.img
dos_load_prim_table: Testing FAT/NTFS conditions
load_pri:0:0 Start: 1 Size: 1999999 Type: 238
load_pri:0:1 Start: 0 Size: 0 Type: 0
load_pri:0:2 Start: 0 Size: 0 Type: 0
load_pri:0:3 Start: 0 Size: 0 Type: 0
bsd_load_table: Table Sector: 1
gpt_load_table: Sector: 1
gpt_load: 0 Starting Sector: 2048 End: 104447 Flag: 0
gpt_load: 1 Starting Sector: 104448 End: 309247 Flag: 0
gpt_load: 2 Starting Sector: 309248 End: 718847 Flag: 0
gpt_load: 3 Starting Sector: 718848 End: 1058815 Flag: 0
gpt_load: 4 Starting Sector: 1058816 End: 1091583 Flag: 0
gpt_load: 5 Starting Sector: 1091584 End: 1173503 Flag: 0
gpt_load: 6 Starting Sector: 0 End: 0 Flag: 0
gpt_load: 7 Starting Sector: 0 End: 0 Flag: 0
gpt_load: 8 Starting Sector: 0 End: 0 Flag: 0
gpt_load: 9 Starting Sector: 0 End: 0 Flag: 0
```

Gpt_load_table jest przypisana do sektora 1.

2. Proszę o podanie sektora startowego gpt_load: 0.

Cytat ze zdjęcia: „gpt_load: 0 Starting Sector: 2048 End: 104447 Flag: 0”. Zatem 2048.

3. Ile niealokowanych sektorów znajdują się w obrazie? Podaj ich sektory

startowe oraz końcowe.

```
(user@user)-[~/Desktop]
$ mmls -A -o 0 LAB_1.img
GUID Partition Table (EFI)
Offset Sector: 0
Units are in 512-byte sectors

Slot      Start      End      Length    Description
001:      0000000000 0000002047 0000002048 Unallocated
010:      0001173504 0001999999 0000826496 Unallocated

(user@user)-[~/Desktop]
$ mmls -A -o 1 LAB_1.img
GUID Partition Table (EFI)
Offset Sector: 1
Units are in 512-byte sectors

Slot      Start      End      Length    Description
000:      0000000000 0000000000 0000000001 Unallocated
002:      0000000035 0001058815 0001058781 Unallocated
005:      0001173504 0001999999 0000826496 Unallocated
```

Wszystkie niealokowane sektory są widoczne. Zarówno z offsetu 0 jak i 1.

4. Podaj ujawnione woluminy.

```
(user@user)-[~/Desktop]
$ mmls -o 0 LAB_1.img
GUID Partition Table (EFI)
Offset Sector: 0
Units are in 512-byte sectors

  Slot      Start      End      Length    Description
000:  Meta      0000000000  0000000000  0000000001  Safety Table
001:  _____ 0000000000  0000002047  0000002048  Unallocated
002:  Meta      0000000001  0000000001  0000000001  GPT Header
003:  Meta      0000000002  0000000033  0000000032  Partition Table
004:  000       0000002048  0000104447  0000102400  fat16
005:  001       0000104448  0000309247  0000204800  fat32
006:  002       0000309248  0000718847  0000409600  ntfs
007:  003       0000718848  0001058815  0000339968  ext4
008:  004       0001058816  0001091583  0000032768  swap
009:  005       0001091584  0001173503  0000081920  minix
010:  _____ 0001173504  0001999999  0000826496  Unallocated

(user@user)-[~/Desktop]
$ mmls -o 1 LAB_1.img
GUID Partition Table (EFI)
Offset Sector: 1
Units are in 512-byte sectors

  Slot      Start      End      Length    Description
000:  _____ 0000000000  0000000000  0000000001  Unallocated
001:  124       0000000001  0000000034  0000000034  未離煌案 0 0 0 0 0 0
002:  _____ 0000000035  0001058815  0001058781  Unallocated
003:  000       0001058816  0001091583  0000032768  swap
004:  001       0001091584  0001173503  0000081920  minix
005:  _____ 0001173504  0001999999  0000826496  Unallocated
006:  Meta      0001999967  0001999998  0000000032  Partition Table
007:  Meta      0001999998  0001999998  0000000001  GPT Header
```

Wszystkie ujawnione woluminy są widoczne. Zarówno z offsetu 0 jak i 1.

5. Wykorzystując polecenie mmstat wyświetl informacje tablicy partycji.

```
(user@user)-[~/Desktop]
$ mmstat LAB_1.img
gpt
```

Informacja tablicy partycji to: gpt.

6. Przy wykorzystaniu narzędzia fsstat wyświetl informacje o woluminie

„ntfs” oraz podaj „Volume Serial Number” oraz informacje o wersji („Version”).

```
(user@user)-[~/Desktop]
$ fsstat -o 0000309248 LAB_1.img
FILE SYSTEM INFORMATION

File System Type: NTFS
Volume Serial Number: 451AF24C771A6637
OEM Name: NTFS
Volume Name: NTFS
Version: Windows XP

METADATA INFORMATION

First Cluster of MFT: 4
First Cluster of MFT Mirror: 25599
Size of MFT Entries: 1024 bytes
Size of Index Records: 4096 bytes
Range: 0 - 68
Root Directory: 5

CONTENT INFORMATION

Sector Size: 512
Cluster Size: 4096
Total Cluster Range: 0 - 51198
Total Sector Range: 0 - 409598

$AttrDef Attribute Values:
$STANDARD_INFORMATION (16) Size: 48-72 Flags: Resident
$ATTRIBUTE_LIST (32) Size: No Limit Flags: Non-resident
$FILE_NAME (48) Size: 68-578 Flags: Resident,Index
$OBJECT_ID (64) Size: 0-256 Flags: Resident
$SECURITY_DESCRIPTOR (80) Size: No Limit Flags: Non-resident
$VOLUME_NAME (96) Size: 2-256 Flags: Resident
$VOLUME_INFORMATION (112) Size: 12-12 Flags: Resident
$DATA (128) Size: No Limit Flags:
$INDEX_ROOT (144) Size: No Limit Flags: Resident
$INDEX_ALLOCATION (160) Size: No Limit Flags: Non-resident
$BITMAP (176) Size: No Limit Flags: Non-resident
$REPARSE_POINT (192) Size: 0-16384 Flags: Non-resident
$EA_INFORMATION (208) Size: 8-8 Flags: Resident
$EA (224) Size: 0-65536 Flags:
$LOGGED_UTILITY_STREAM (256) Size: 0-65536 Flags: Non-resident
```

Volume Serial Number: 451AF24C771A6637

Version: Windows XP.

Zadanie 4 – Pozyskanie obrazu nośnika przy użyciu narzędzia „ewfacquire”.

Przy wykorzystaniu dowolnego pendriva proszę o sporządzenie jego kopii binarnej przy pomocy narzędzia ewfacquire oraz wyświetlenie najistotniejszych informacji.

Krok 1: sprawdzenie podłączenia nośnika.

```
(user@user)-[~]
$ sudo fdisk -l
Disk /dev/sda: 25 GiB, 26843545600 bytes, 52428800 sectors
Disk model: VBOX HARDDISK
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: 0xeed8620f

Device      Boot    Start        End    Sectors    Size Id Type
/dev/sda1   *          2048    50427903    50425856    24G 83 Linux
/dev/sda2             50429950    52426751    1996802    975M  f W95 Ext'd (LBA)
/dev/sda5             50429952    52426751    1996800    975M 82 Linux swap / Solaris

Disk /dev/sdb: 58.59 GiB, 62914560000 bytes, 122880000 sectors
Disk model: Flash Disk
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: 0x5d742c9d

Device      Boot    Start        End    Sectors    Size Id Type
/dev/sdb1   *          128    122879999    122879872    58.6G  7 HPFS/NTFS/exFAT
```

Krok 2: konfiguracja materiału dowodowego. Do tego start procesu.

```
(user@user)-[~]
$ sudo ewfacquire /dev/sdb1
ewfacquire 20140814

Device information:
Bus type:          USB
Vendor:            Generic
Model:             Flash Disk
Serial:            C9636D88

Storage media information:
Type:              Device
Media type:        Removable
Media size:        62 GB (62914494464 bytes)
Bytes per sector:  512

Acquiry parameters required, please provide the necessary input
Image path and filename without extension: floppy
Case number: 001
Description: test
Evidence number: 1
Examiner name: user
Notes: -
```



```

The following acquire parameters were provided:
Image path and filename:      floppy.E01
Case number:                  001
Description:                  test
Evidence number:              1
Examiner name:               user
Notes:                        -
Media type:                   removable disk
Is physical:                  yes
EWF file format:              EnCase 6 (.E01)
Compression method:           deflate
Compression level:            none
Acquiry start offset:         0
Number of bytes to acquire:   1.0 GiB (1073741824 bytes)
Evidence segment file size:   1.4 GiB (1572864000 bytes)
Bytes per sector:             512
Block size:                   64 sectors
Error granularity:            64 sectors
Retries on read error:        2
Zero sectors on read error:   no

Continue acquire with these values (yes, no) [yes]:

Acquiry started at: Oct 16, 2023 18:49:44
This could take a while.

Status: at 1%.
        acquired 15 MiB (16482304 bytes) of total 1.0 GiB (1073741824 bytes).
        completion in 6 minute(s) and 36 second(s) with 2.5 MiB/s (2684354 bytes/second).

Status: at 3%.
        acquired 37 MiB (39026688 bytes) of total 1.0 GiB (1073741824 bytes).

```

Koniec etapu „acquire”.

```

Acquiry completed at: Oct 16, 2023 18:52:55

Written: 1.0 GiB (1073743140 bytes) in 3 minute(s) and 11 second(s) with 5.3 MiB/s (56
21691 bytes/second).
MD5 hash calculated over data:          bea1bd6fbcf0ef1e1dd3b0068d376b10
ewfacquire: SUCCESS

(user@user)-[~]
$ █

```

Krok 3: weryfikacja poprawności wykonanego obrazu.

```

(user@user)-[~]
$ sudo ewfverify floppy.E01
ewfverify 20140814

Verify started at: Oct 16, 2023 18:56:04
This could take a while.

Status: at 46%.
        verified 478 MiB (502169600 bytes) of total 1.0 GiB (1073741824 bytes).
        completion in 4 second(s) with 128 MiB/s (134217728 bytes/second).

Status: at 94%.
        verified 972 MiB (1020002304 bytes) of total 1.0 GiB (1073741824 bytes).
        completion in 0 second(s) with 128 MiB/s (134217728 bytes/second).

Verify completed at: Oct 16, 2023 18:56:12

Read: 1.0 GiB (1073741824 bytes) in 8 second(s) with 128 MiB/s (134217728 bytes/second).

MD5 hash stored in file:          bea1bd6fbcf0ef1e1dd3b0068d376b10
MD5 hash calculated over data:    bea1bd6fbcf0ef1e1dd3b0068d376b10

ewfverify: SUCCESS

```


Krok 4: przykładowa weryfikacja czy widać moje pliki z „PenDrive”.

```
(user@user)-[~]  
$ fls floppy.E01  
r/r 8195: $EMPTY_VOLUME_LABEL (Volume Label Entry)  
r/r 8196: $ALLOC_BITMAP  
r/r 8197: $UPCASE_TABLE  
d/d 8198: System Volume Information  
r/r 8202: Plan tren.xlsx  
r/r 8205: szkla_podstawy_nawigacji.pdf  
r/r 8209: szkla_ratownictwo.pdf  
r/r 8213: szkla_meteorologia.pdf  
r/r 8217: szkla_prace_bosmanskie.pdf  
r/r 8221: BUDAPESZT - plan.docx  
r/r 8225: LISTA DO BUDAPESZTU.docx  
v/v 33456131: $MBR  
v/v 33456132: $FAT1  
V/V 33456133: $OrphanFiles
```