

Digital Forensics Exercises

Maximilian Heim

December 11, 2023

Contents

1	Exercise 1.1	3
1.1	Useful links	3
1.2	a) Which timestamps do Linux and Windows provide?	3
2	Exercise 1.3: Read only	3
2.1	a)	3
2.2	b)	3
3	Exercise 2.1: Forensic backup	4
4	Exercise 2.2: Hash validation	4
5	Exercise 2.7: In depth analysis SK	4
5.1	a)	4
5.2	b)	5
5.3	c)	10

1 Exercise 1.1

1.1 Useful links

Time for Truth: Forensic Analysis of NTFS Timestamps - <https://eprints.cs.univie.ac.at/7091/>

1.2 a) Which timestamps do Linux and Windows provide?

Linux

1. atime - Access time
2. mtime - Modification time
3. ctime - Creation time

Windows <https://eprints.cs.univie.ac.at/7091/1/3465481.3470016.pdf>

1. Modified - Modification time
2. Accessed - Access time
3. Changed - Change of file metadata via the MFT entry
4. Birth - Creation of file via the MFT entry

2 Exercise 1.3: Read only

Exercise description Ein USB-Stick und ein virtuelles Laufwerk sollen beim Anschließen an einen Rechner nicht vollständig gemountet werden, sondern im „nur-lesen“ Modus eingebunden werden. Beschreiben Sie die notwendigen Konfigurationen und erstellen Sie jeweils ein Script um den Vorgang zu automatisieren.

2.1 a)

To mount a block device read in read only mode `mount -o ro <drive> /mnt` may be used

2.2 b)

First automount has to be disabled via `mountvol.exe /N`. After that the media can be connected to the system. The next step is to enable the read only flag for the volume via `attributes volume set readonly`. Then the volume may be mounted. <https://superuser.com/questions/213005/how-to-mount-an-ntfs-partition-read-only-i>

3 Exercise 2.1: Forensic backup

Exercise description Erstellen Sie ein mit dem Tool dd [Howto] ein 50MB großes virtuelles Laufwerk und formatieren Sie diese mit FAT. Verwenden Sie hierfür das Tool mkfs [Howto]. Binden Sie das Laufwerk ein und kopieren Sie anschließend verschiedene, beliebige Dateien auf die Partition. Erstellen Sie eine forensische Kopie der Partition - verwenden Sie hierfür das Programm dcfldd [Website][Howto]. Beachten Sie dabei, dass das Laufwerk nicht eingebunden sein darf. Der MD5-Hash-Wert der Kopie soll dabei in eine Datei geschrieben werden. Beschreiben Sie Ihr Vorgehen und die verwendeten Befehle. Wie sieht der Befehl aus, wenn Sie das Image in 10 MB große Dateien aufsplitten?

Backup partition as image The following command dumps the partition into an image file and saves the hash into a file

```
dcfldd if=imagetest.img of=imagedump.img hashlog=imagedumphash.md5 hash=md5
```

Backup partition splitted This command additionally splits the contents into 10 MB files

```
dcfldd if=imagetest.img split=10000000 of=imagedump.img hashlog=imagedumphash.md5 hash=md5
```

4 Exercise 2.2: Hash validation

Exercise Validieren Sie den Hash-Wert aus Übung 2.1, indem Sie einen Hashwert des virtuellen Laufwerkes mit dem Programm md5sum erzeugen. Führen Sie anschließend einen automatisierten Vergleich der beiden Hash-Werte durch. Beschreiben Sie Ihr Vorgehen und die verwendeten Befehle.

Generation of hash md5sum imagetest.img > original.md5

Comparison of hashes cmp -n 32 -ignore-initial 0:14 original.md5 imagedumphash.md5

References

5 Exercise 2.7: In depth analysis SK

5.1 a)

Partition table mmls uebung_2-7.dd

```
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: _____	0000000000	0000002047	0000002048
Unallocated			
002: 000:000	0000002048	0000053247	0000051200
Linux (0x83)			
003: _____	0000053248	0000104447	0000051200
Unallocated			
004: 000:002	0000104448	0000155647	0000051200
Win95 FAT32 (0x0b)			
005: Meta	0000155648	0000204799	0000049152
DOS Extended (0x05)			
006: Meta	0000155648	0000155648	0000000001
Extended Table (#1)			
007: _____	0000155648	0000157695	0000002048
Unallocated			
008: 001:000	0000157696	0000204799	0000047104
Linux (0x83)			

5.2 b)

Partition 2 `fsstat -f ext3 -o 2048 uebung_2-7.dd`

FILE SYSTEM INFORMATION

File System Type: Ext3

Volume Name:

Volume ID: 627cea8be986a5a3b94e761f598eab5a

Last Written at: 2012-03-12 13:40:49 (CET)

Last Checked at: 2012-03-21 15:01:53 (CET)

Last Mounted at: 2012-03-24 14:33:01 (CET)

Unmounted properly

Source OS: Linux

Dynamic Structure

Compat Features: Journal, Ext Attributes, Resize Inode, Dir Index

InCompat Features: Filetype,

Read Only Compat Features: Sparse Super,

Journal ID: 00

Journal Inode: 8

METADATA INFORMATION

Inode Range: 1 - 6401

Root Directory: 2

Free Inodes: 6379

CONTENT INFORMATION

Block Range: 0 — 25599
Block Size: 1024
Reserved Blocks Before Block Groups: 1
Free Blocks: 12343

BLOCK GROUP INFORMATION

Number of Block Groups: 4
Inodes per group: 1600
Blocks per group: 8192

Group: 0:
 Inode Range: 1 — 1600
 Block Range: 1 — 8192
 Layout:
 Super Block: 1 — 1
 Group Descriptor Table: 2 — 2
 Data bitmap: 102 — 102
 Inode bitmap: 103 — 103
 Inode Table: 104 — 303
 Data Blocks: 304 — 8192
 Free Inodes: 1579 (98%)
 Free Blocks: 707 (8%)
 Total Directories: 2

Group: 1:
 Inode Range: 1601 — 3200
 Block Range: 8193 — 16384
 Layout:
 Super Block: 8193 — 8193
 Group Descriptor Table: 8194 — 8194
 Data bitmap: 8294 — 8294
 Inode bitmap: 8295 — 8295
 Inode Table: 8296 — 8495
 Data Blocks: 8496 — 16384
 Free Inodes: 1600 (100%)
 Free Blocks: 3955 (48%)
 Total Directories: 0

Group: 2:
 Inode Range: 3201 — 4800
 Block Range: 16385 — 24576
 Layout:
 Data bitmap: 16385 — 16385
 Inode bitmap: 16386 — 16386
 Inode Table: 16387 — 16586
 Data Blocks: 16387 — 16386, 16587 — 24576
 Free Inodes: 1600 (100%)
 Free Blocks: 6961 (84%)
 Total Directories: 0

Group: 3:
Inode Range: 4801 — 6400
Block Range: 24577 — 25599
Layout:
Super Block: 24577 — 24577
Group Descriptor Table: 24578 — 24578
Data bitmap: 24678 — 24678
Inode bitmap: 24679 — 24679
Inode Table: 24680 — 24879
Data Blocks: 24880 — 25599
Free Inodes: 1600 (100%)
Free Blocks: 720 (70%)
Total Directories: 0

Partition 4 `fsstat -f fat -o 104448 uebung_2-7.dd`

FILE SYSTEM INFORMATION

File System Type: FAT16

OEM Name: mkdosfs
Volume ID: 0x38ba908
Volume Label (Boot Sector):
Volume Label (Root Directory):
File System Type Label: FAT16

Sectors before file system: 0

File System Layout (in sectors)
Total Range: 0 — 51199
* Reserved: 0 — 3
** Boot Sector: 0
* FAT 0: 4 — 55
* FAT 1: 56 — 107
* Data Area: 108 — 51199
** Root Directory: 108 — 139
** Cluster Area: 140 — 51199

METADATA INFORMATION

Range: 2 — 817478
Root Directory: 2

CONTENT INFORMATION

Sector Size: 512
Cluster Size: 2048
Total Cluster Range: 2 — 12766

FAT CONTENTS (in sectors)

144–11287 (11144) → EOF
11288–17183 (5896) → EOF
17184–20971 (3788) → EOF
20980–20983 (4) → EOF
20984–21687 (704) → EOF
21688–22391 (704) → EOF
22392–23099 (708) → EOF
23100–23219 (120) → EOF
23220–23339 (120) → EOF
23340–23459 (120) → EOF

Partition 8 `fsstat -f ext -o 157696 uebung_2-7.dd`

FILE SYSTEM INFORMATION

File System Type: Ext4

Volume Name:

Volume ID: d787b67ed5e90caa3f4a161a87787e76

Last Written at: 2012–03–12 13:40:49 (CET)

Last Checked at: 2012–03–21 15:03:05 (CET)

Last Mounted at: 2012–03–07 14:48:20 (CET)

Unmounted properly

Last mounted on: /home/cmoch/ueb_albsig/ext4

Source OS: Linux

Dynamic Structure

Compat Features: Journal, Ext Attributes, Resize Inode, Dir Index

InCompat Features: Filetype, Extents, Flexible Block Groups,

Read Only Compat Features: Sparse Super, Huge File, Extra Inode Size

Journal ID: 00

Journal Inode: 8

METADATA INFORMATION

Inode Range: 1 – 5905

Root Directory: 2

Free Inodes: 5883

Inode Size: 128

CONTENT INFORMATION

Block Groups Per Flex Group: 16

Block Range: 0 – 23551

Block Size: 1024

Reserved Blocks Before Block Groups: 1

Free Blocks: 12620

BLOCK GROUP INFORMATION

Number of Block Groups: 3

Inodes per group: 1968

Blocks per group: 8192

Group: 0:

Inode Range: 1 – 1968

Block Range: 1 – 8192

Layout:

Super Block: 1 – 1

Group Descriptor Table: 2 – 2

Group Descriptor Growth Blocks: 3 – 93

Data bitmap: 94 – 94

Inode bitmap: 110 – 110

Inode Table: 126 – 371

Data Blocks: 372 – 8192

Free Inodes: 1947 (98%)

Free Blocks: 7341 (89%)

Total Directories: 2

Group: 1:

Inode Range: 1969 – 3936

Block Range: 8193 – 16384

Layout:

Super Block: 8193 – 8193

Group Descriptor Table: 8194 – 8194

Group Descriptor Growth Blocks: 8195 – 8285

Data bitmap: 95 – 95

Inode bitmap: 111 – 111

Inode Table: 372 – 617

Data Blocks: 618 – 16384

Free Inodes: 1968 (100%)

Free Blocks: 358 (4%)

Total Directories: 0

Group: 2:

Inode Range: 3937 – 5904

Block Range: 16385 – 23551

Layout:

Data bitmap: 96 – 96

Inode bitmap: 112 – 112

Inode Table: 618 – 863

Data Blocks: 864 – 23551

Free Inodes: 1968 (100%)

Free Blocks: 4921 (68%)

Total Directories: 0

5.3 c)

Partition 2 fls -o 2048 -f ext uebung_2-7.dd

```
d/d 11: lost+found
r/r 12: ebc36c92-3886-11e1-af06-5c260a3d892a.mp3
r/r 13: ebc89640-3886-11e1-af06-5c260a3d892a.mp3
r/r 14: 4cafbc82-389a-11e1-af06-5c260a3d892a.mp3
r/r 15: a5df3984-38bf-11e1-af06-5c260a3d892a.txt
r/r * 16: 7e070e10-38ff-11e1-af06-5c260a3d892a.txt
r/r * 17: 64349836-3936-11e1-af06-5c260a3d892a.txt
r/r 18: 2f2e3c1c-3942-11e1-af06-5c260a3d892a.jpg
r/r 19: d0b8aa62-3948-11e1-af06-5c260a3d892a.jpg
r/r 20: 4f20572c-395f-11e1-af06-5c260a3d892a.bmp
r/r 21: ded49934-397b-11e1-af06-5c260a3d892a.txt
r/r 22: e8ca3836-39bc-11e1-af06-5c260a3d892a.txt
r/r 23: cb04b604-39cc-11e1-af06-5c260a3d892a.txt
V/V 6401: $OrphanFiles
```

Partition 4 fls -o 104448 -f fat uebung_2-7.dd

```
r/r 7: d1f7f3b0-6891-11e1-af06-5c260a3d892a.mp3
r/r 12: d1faea98-6891-11e1-af06-5c260a3d892a.mp3
r/r 17: cb8b76b8-68d3-11e1-af06-5c260a3d892a.mp3
r/r * 22: 7dc51cc0-68da-11e1-af06-5c260a3d892a.txt
r/r * 27: 3d211488-68eb-11e1-af06-5c260a3d892a.txt
r/r 32: 8b103ba4-6938-11e1-af06-5c260a3d892a.txt
r/r 37: 18fb26fe-6957-11e1-af06-5c260a3d892a.bmp
r/r 42: 3cd504f6-6983-11e1-af06-5c260a3d892a.bmp
r/r 47: 9cf197c2-69cf-11e1-af06-5c260a3d892a.bmp
r/r 52: 237896fa-6a0d-11e1-af06-5c260a3d892a.txt
r/r 57: 3d86f27e-6a50-11e1-af06-5c260a3d892a.txt
r/r 62: 4df1a002-6a5d-11e1-af06-5c260a3d892a.txt
v/v 817475: $MBR
v/v 817476: $FAT1
v/v 817477: $FAT2
V/V 817478: $OrphanFiles
```

Partition 8 fls -o 157696 -f ext uebung_2-7.dd

```
d/d 11: lost+found
r/r 12: 77336f00-6b38-11e1-af06-5c260a3d892a.mp3
r/r 13: 7735ba62-6b38-11e1-af06-5c260a3d892a.mp3
r/r 14: 5562411e-6b86-11e1-af06-5c260a3d892a.mp3
r/r 15: 24012126-6bb8-11e1-af06-5c260a3d892a.txt
r/r * 16: 4e249608-6bf5-11e1-af06-5c260a3d892a.txt
r/r * 17: 876a0370-6c0f-11e1-af06-5c260a3d892a.txt
r/r 18: 2c430cc4-6c25-11e1-af06-5c260a3d892a.bmp
r/r 19: 6e8056fa-6c4d-11e1-af06-5c260a3d892a.jpg
r/r 20: 8ae5a6c6-6c78-11e1-af06-5c260a3d892a.gif
```

r/r 21: 901b0e08-6cc1-11e1-af06-5c260a3d892a.txt
r/r 22: e26033d4-6cc9-11e1-af06-5c260a3d892a.txt
r/r 23: 4e499582-6cd8-11e1-af06-5c260a3d892a.txt
V/V 5905: \$OrphanFiles