Gregg Gunsch
Dec 2010

A Digital Forensics Challenge

Imagine that someone has imaged some floppy disks and stored their image files on a flashdrive. Next they imaged the flashdrive and stored its image file on a harddrive. You only have the flashdrive's image file, ***flashdrive.img***. Your task is to analyze ***flashdrive.img*** to determine where a specific floppy's image file was stored, and carved the floppy's image file out of ***flashdrive.img*** using dd. Yes, you could easily use automated tools, but that isn't the point of this exercise. You should be able to solve every bit of this by hand, using Brian Carrier's book, "File System Forensic Analysis," as a guide. All of the specifics you need are provided in the ***fsstat*** output and WinHex screenshots below.

Below is the top part of the output of the command: ***fsstat –o 32 flashdrive.img***

```
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: FAT16

OEM Name: MSDOS5.0
Volume ID: 0xa00c263e
Volume Label (Boot Sector): NO NAME
Volume Label (Root Directory):
File System Type Label: FAT16

Sectors before file system: 32

File System Layout (in sectors)
Total Range: 0 - 4095967
* Reserved: 0 - 3
** Boot Sector: 0
* FAT 0: 4 - 253
* FAT 1: 254 - 503
* Data Area: 504 - 4095967
** Root Directory: 504 - 535
** Cluster Area: 536 - 4095959
** Non-clustered: 4095960 - 4095967

METADATA INFORMATION
--------------------------------------------
Range: 2 - 65279590
Root Directory: 2

CONTENT INFORMATION
--------------------------------------------
Sector Size: 512
Cluster Size: 32768
Total Cluster Range: 2 - 63992
```

Below is a WinHex snapshot of a portion of the Root Directory, covering the area that contains the directory entry of the floppy's image file:

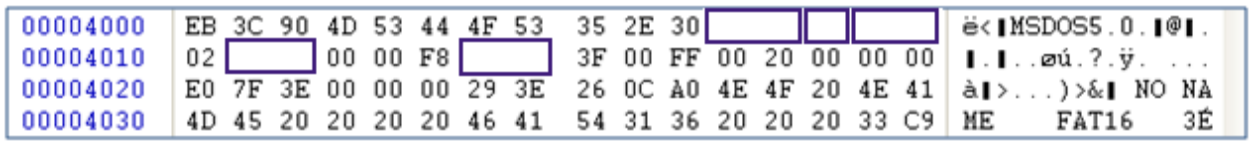
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00044640	42	30	00	30	00	31	00	00	00	FF	FF	0F	00	F5	FF	FF	B0.0.1...ÿÿ▌.õÿÿ
00044650	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿ..ÿÿÿÿ
00044660	01	43	00	46	00	32	00	33	00	30	00	0F	00	F5	6D	00	▌C.F.2.3.0.▌.õm.
00044670	69	00	64	00	74	00	65	00	72	00	00	00	6D	00	2E	00	i.d.t.e.r...m...
00044680	43	46	32	33	30	4D	7E	31	30	30	31	21	00	16	75	B0	CF230M~1001!.▌u°
00044690	6F	3B	4A	3C	00	00	2F	A2	51	37	38	3E	00	80	16	00	o;J<..⁄¢Q78>.▌▌.
000446A0	4A	4F	45	4A	41	43	4B	20	49	4D	47	20	00	55	6A	B0	JOEJACK IMG .Uj°
000446B0	6F	3B	88	3D	00	00	3B	3A	D7	2C	0B	3E	00	80	16	00	o;▌=..;:×,▌>.▌▌.
000446C0	43	61	00	73	00	74	00	65	00	72	00	0F	00	44	2E	00	Ca.s.t.e.r.▌.D..
000446D0	70	00	70	00	74	00	78	00	00	00	00	00	FF	FF	FF	FF	p.p.t.x.....ÿÿÿÿ
000446E0	02	73	00	69	00	63	00	20	00	53	00	0F	00	44	63	00	▌s.i.c. .S.▌.Dc.
000446F0	69	00	65	00	6E	00	63	00	65	00	00	00	20	00	4D	00	i.e.n.c.e... .M.
00044700	01	44	00	69	00	67	00	69	00	74	00	0F	00	44	61	00	▌D.i.g.i.t.▌.Da.
00044710	6C	00	20	00	46	00	6F	00	72	00	00	00	65	00	6E	00	l. .F.o.r...e.n.
00044720	44	49	47	49	54	41	7E	31	50	50	54	20	00	12	18	BB	DIGITA~1PPT .▌▌»
00044730	7E	3B	88	3D	00	00	A2	B8	7E	3B	DA	54	83	75	C4	00	~;▌=..¢¸~;ÚT▌uÄ.
00044740	4E	49	53	44	4F	57	7E	31	45	58	45	20	00	AE	35	0C	NISDOW~1EXE .®5▌
00044750	3D	3C	88	3D	00	00	28	0A	37	3C	07	5E	F8	39	06	00	=<▌=..(▌7<▌^ø9▌.
00044760	4D	45	4D	54	45	53	54	20	44	4B	20	20	18	C3	E2	9D	MEMTEST DK ▌Ãâ▌
00044770	88	3D	88	3D	00	00	E2	9D	88	3D	BF	5F	00	80	16	00	▌=▌=..â▌▌=¿_.▌▌.
00044780	42	30	00	31	00	00	00	FF	FF	FF	FF	0F	00	E0	FF	FF	B0.1...ÿÿÿÿ▌.àÿÿ
00044790	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿ..ÿÿÿÿ
000447A0	01	43	00	46	00	33	00	30	00	35	00	0F	00	E0	66	00	▌C.F.3.0.5.▌.àf.
000447B0	6C	00	6F	00	70	00	70	00	79	00	00	00	2E	00	30	00	l.o.p.p.y.....0.
000447C0	43	46	33	30	35	46	7E	31	30	30	31	20	00	40	75	B0	CF305F~1001 .@u°
000447D0	6F	3B	88	3D	00	00	2F	A2	51	37	65	3E	00	80	16	00	o;▌=../¢Q7e>.▌▌.

- What is the absolute start location (sector number, and byte offset in hexadecimal) of the <u>whole</u> Root Directory? _____
- How many files altogether are represented in this section of the Root Directory? ____
- How many of them use long file names? ____
- How many of them could be floppy drive images? ____
  (Hint: Exactly how big are the floppy disks we typically use these days?)
- Of these, which one is the floppy image created on 8 Dec 2010?
  - What is its 8.3 filename? _____
  - Does it have a long filename, and if so, what is it? _____
  - What is the start cluster number (base 10)? _____
  - What is the absolute start sector number (base 10)? _____
  - What is the logical size of this file in bytes? _____
  - What is the physical size of this file in clusters? _____
  - How much file slack is there (RAM + disk)? _____

- Fill in the blanks for the command to carve this floppy from the physical device:

  dd if=flashdrive.img of=carvedfloppy bs=_____ count=_____ skip=_____

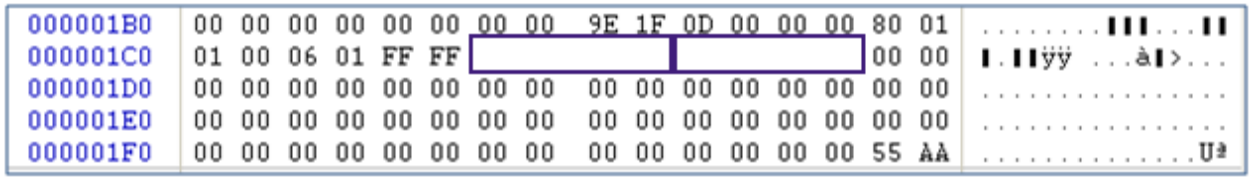
Below are the first 64 bytes of absolute sector 32 of the flashdrive.img file.



00004000	EB 3C 90 4D 53 44 4F 53 35 2E 30 [ ] [ ] [ ]	ë<▮MSDOS5.0.▮@▮.
00004010	02 [ ] 00 00 F8 [ ] 3F 00 FF 00 20 00 00 00	▮.▮..øú.?.ÿ. ...
00004020	E0 7F 3E 00 00 00 29 3E 26 0C A0 4E 4F 20 4E 41	à▮>...)>&▮ NO NA
00004030	4D 45 20 20 20 20 46 41 54 31 36 20 20 20 33 C9	ME FAT16 3É

- What is this sector called? _____
- Fill in the blanks in the figure above. For answers requiring multiple bytes per field, be sure to record your answer in little endian form and zero-fill as appropriate: make the sector look like it should.
  - byte offset 11-12: _____
  - byte offset 13: _____
  - byte offset 14-15: _____
  - byte offset 17-18: _____
  - byte offset 22-23: _____

Below is the tail end of the first sector of the flashdrive.img file.



000001B0	00 00 00 00 00 00 00 00 9E 1F 0D 00 00 00 80 01	........▮▮▮...▮▮
000001C0	01 00 06 01 FF FF [ ] [ ] 00 00	▮.▮▮ÿÿ ...à▮>...
000001D0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
000001E0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
000001F0	00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA	..............Uª

- What is this sector called? _____
- Fill in the blanks in the figure above. For answers requiring multiple bytes per field, be sure to record your answer in little endian form and zero-fill as appropriate: make the sector look like it should.
  - bytes 1C6-1C9: _____
  - bytes 1CA-1CD: _____