

Digital forensics

Andrej Brodnik

Computer

chapter 15

- pre-knowledge:
 - architecture of computers
 - basics (BIOS)
 - operating system
 - secondary memory (disc) and its organization
 - file systems

Startup

- startup steps
- BIOS (*Basic Input Output System*)
 - Open Firmware (Mac PowerPC), EFI (Mac Intel), Open Boot PROM (Sun), ...
- POST (*Power On Self Test*)

- the operating data are stored in xROM
- sometimes the password protects the data – password is entered by the user

Startup...

- example *Moussawi*:

The computer has been shut down for a very long time and the battery on the motherboard has been emptied

- how the data is encrypted
 - ASCII, ...
 - Little / big endian
- What happens if you take disc to another computer

File format

- at the beginning all files have their unique signatures (www.garykessler.net/library/file_sigs.html)
- jpg: *FF D8 FF E0* or *FF D8 FF E3*
- gif: *47 49 46 38 37 61* or *47 49 46 38 39 61*
- doc: *D0 CF 11 E0 A1 B1 1A E1*

File format - example

- jpeg encoded exif (*Exchangeable image file format*) file

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	FF	D8	FF	E1	16	B1	45	78	69	66	00	00	4D	4D	00	2A	ÿøÿá ±Exif MM *
00000010	00	00	00	08	00	08	01	0F	00	02	00	00	00	16	00	00	'
00000020	01	B2	01	10	00	02	00	00	00	1C	00	00	01	C8	01	12	È
00000030	00	03	00	00	00	01	00	01	00	00	01	1A	00	05	00	00	ä
00000040	00	01	00	00	01	E4	01	1B	00	05	00	00	00	01	00	00	i (
00000050	01	EC	01	28	00	03	00	00	00	01	00	02	00	00	02	13	i
00000060	00	03	00	00	00	01	00	01	00	00	87	69	00	04	00	00	i
00000070	00	01	00	00	01	F4	00	00	09	34	00	00	00	00	00	00	ô 4
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	█
00000090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	45	41	EA
000001C0	53	54	4D	41	4E	20	4B	4F	44	41	4B	20	43	4F	4D	50	STMAN KODAK COMP
000001D0	41	4E	59	00	4B	4F	44	41	4B	20	44	58	34	33	33	30	ANY KODAK DX4330
000001E0	20	44	49	47	49	54	41	4C	20	43	41	4D	45	52	41	00	DIGITAL CAMERA
000001F0	00	00	00	E6	00	00	00	01	00	00	00	E6	00	00	00	01	æ æ
00000200	00	24	82	9A	00	05	00	00	00	01	00	00	03	DA	82	9D	\$ Û
00000210	00	05	00	00	00	01	00	00	03	E2	88	22	00	03	00	00	â "
00000220	00	01	00	02	00	00	90	00	00	07	00	00	00	04	30	32	02
00000230	32	30	90	03	00	02	00	00	00	14	00	00	03	EA	90	04	20 è

File format

- the file can be embedded in another file
 - find the file
 - it can be labeled and copied (*copy-paste*)
 - or use tool **dd**
- this procedure is called *carving*
- other tools:
 - scalpel (<http://www.digitalforensicssolutions.com/Scalpel/>), DataLifter (<http://www.datalifter.com/>)
 - EnCase (<http://www.guidancesoftware.com/forensic.htm>), FTK (Forensic Toolkit, <http://accessdata.com/products/computer-forensics/ftk>), X-Ways (<http://www.x-ways.net/>)

Curving

- in the end, we only get content and not metadata from the directory
- The other problem is that the data can be scattered through the disk
 - Adroit (<http://digital-assembly.com/products/adroit-photo-forensics/>)

File format - challenge

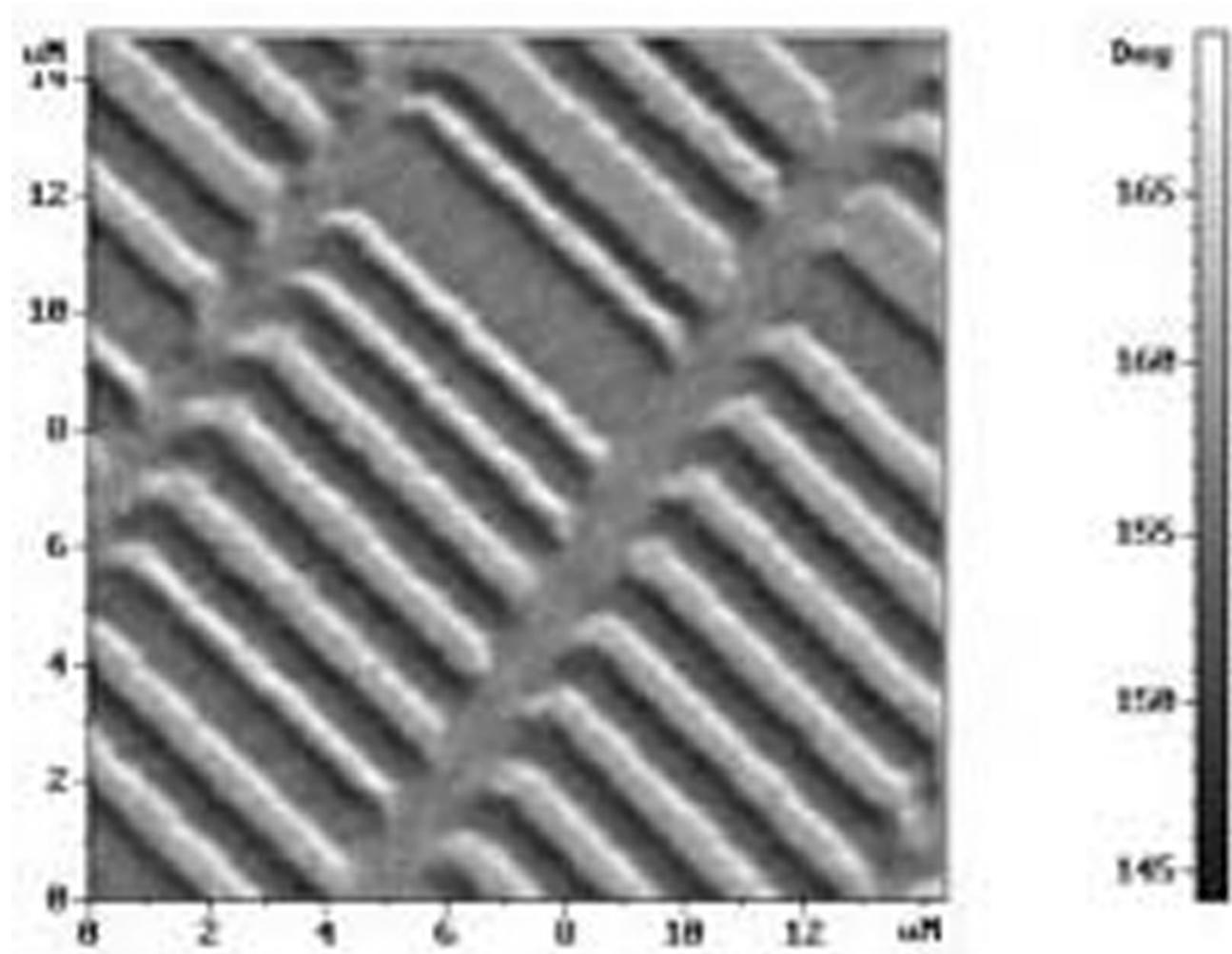
- Challenge: Embed one file in the another file and publish that on the forum. The other colleagues should find the embedded file and extract it using tools like dd or some other tools motioned it the previous slides.
- Challenge: Divide the file into more pieces and insert each one into another file and post it all in the forum. Let your colleagues reconstruct your distributed pieces.

Data storage and hiding

- the I / O units are connected to the computer via:
 - bus (IDE, ATA, SATA; SCSI, firewire)
 - interface (*controller*)
- the interfaces can also be smart
 - SMART (*Self-Monitoring, Analysis, and Reporting Technology*)
 - keep access statistics and other similar data
 - usually are not relevant for forensic research

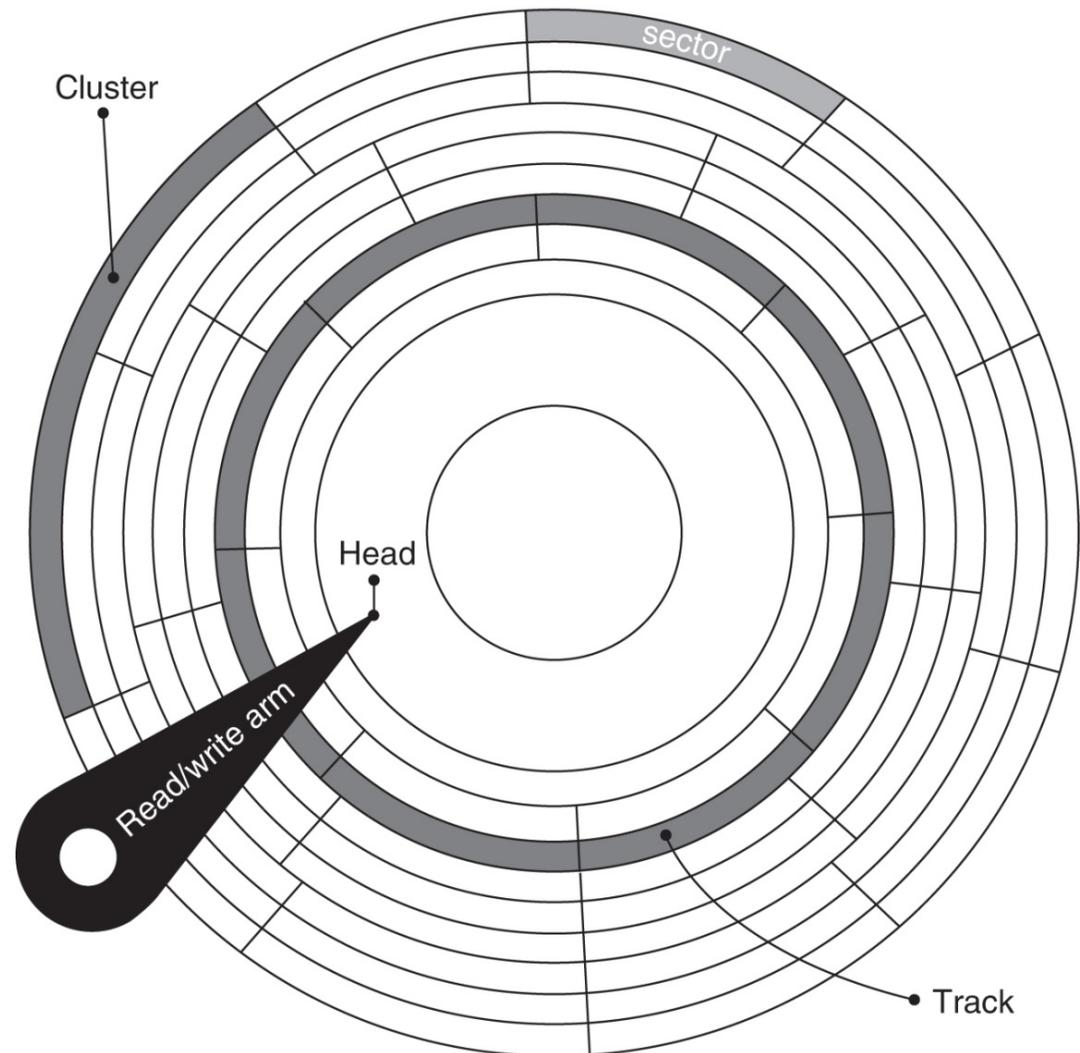
Data storage and hiding

- usually we store data permanently on a disk
- What does the hard drive look like?



Data storage and hiding

- how is the disk organized?
 - spindle, platter, cylinders, tracks, sectors, cluster
- at the first sector are control data (MBR, *master boot record*)
 - size (geometry), blocks, partitions, ...
- what organization in SSD looks like?

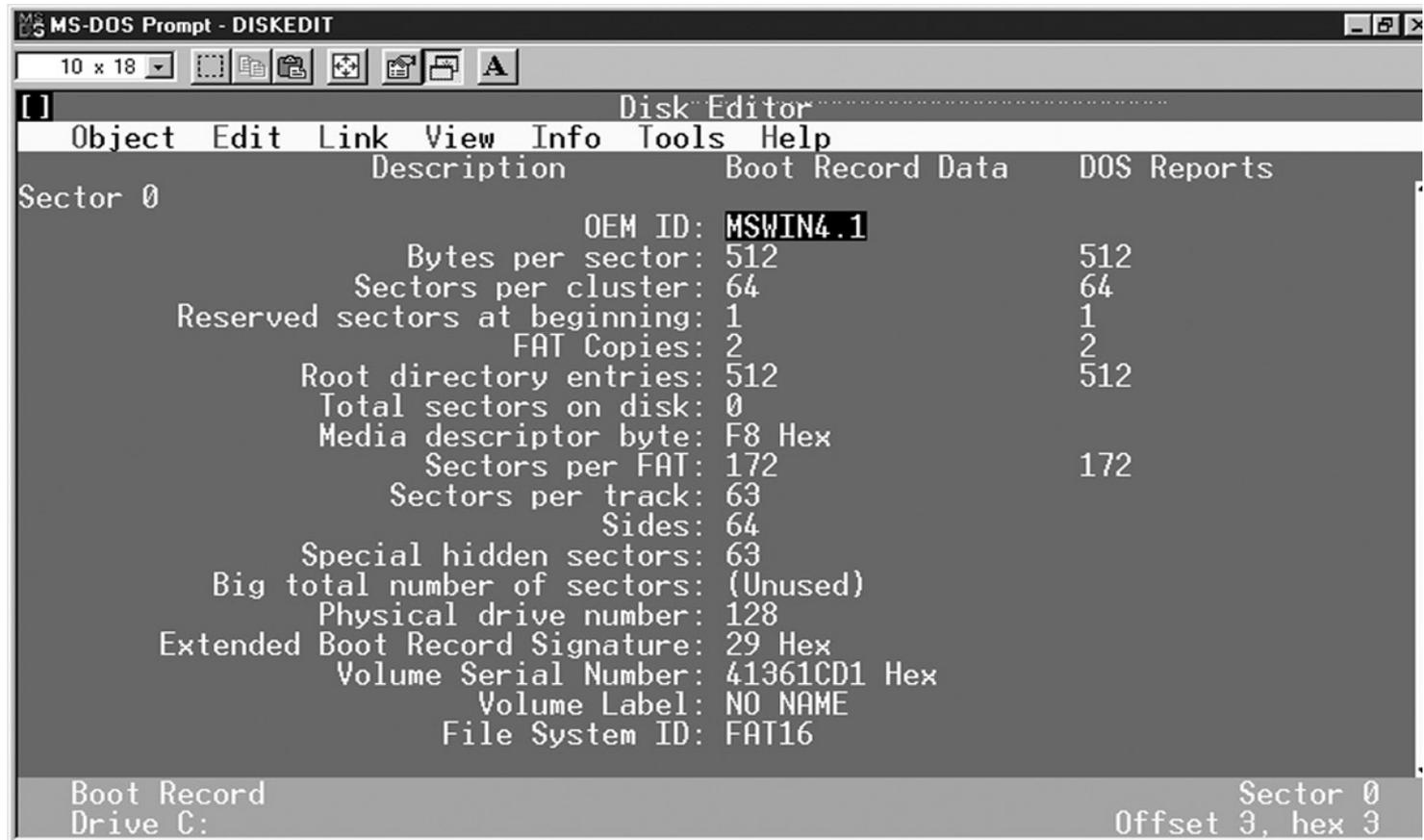


Data storage and hiding

- *Challenge: find the anadisk tool and see what it knows and can do.*
- *Challenge: what is the MBR structure? Build your MBR and post it in the forum..*

Data storage and hiding

- look at the Windows 95 boot sector with the Norton Disk Utils tool



The screenshot shows the MS-DOS Disk Editor interface. The title bar reads "MS-DOS Prompt - DISKEDIT". The menu bar includes "Object", "Edit", "Link", "View", "Info", "Tools", and "Help". The main display area shows the following information:

```

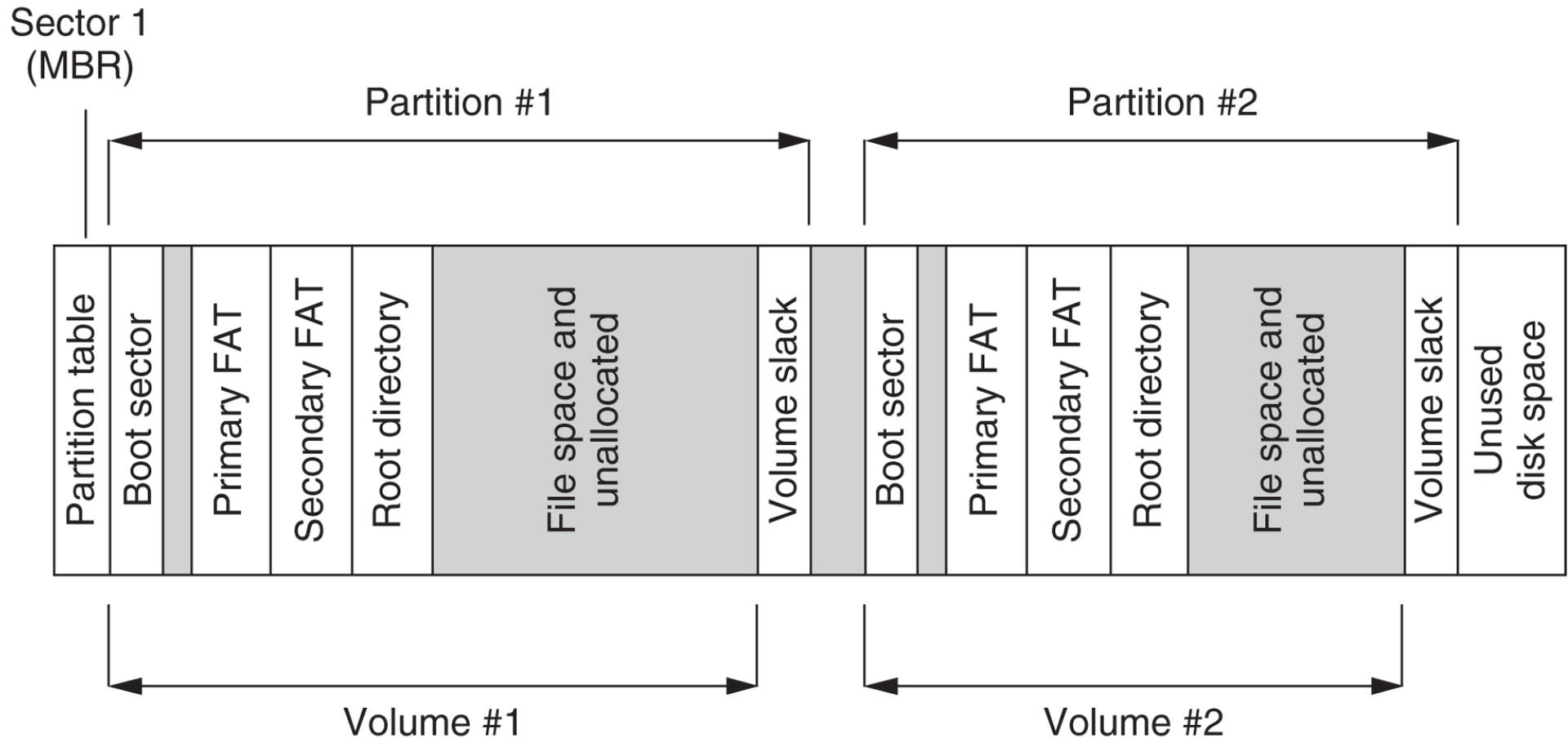
Sector 0
Description          Boot Record Data  DOS Reports
OEM ID: MSWIN4.1
Bytes per sector: 512          512
Sectors per cluster: 64       64
Reserved sectors at beginning: 1 1
FAT Copies: 2                 2
Root directory entries: 512    512
Total sectors on disk: 0
Media descriptor byte: F8 Hex
Sectors per FAT: 172          172
Sectors per track: 63
Sides: 64
Special hidden sectors: 63
Big total number of sectors: (Unused)
Physical drive number: 128
Extended Boot Record Signature: 29 Hex
Volume Serial Number: 41361CD1 Hex
Volume Label: NO NAME
File System ID: FAT16

Boot Record          Sector 0
Drive C:             Offset 3, hex 3

```

Data storage and hiding

- simplified organization of the disk with the FAT file system



Data storage and hiding

- partition, volume, sector
- inside the file system
- can also be without the file system

Data storage and hiding

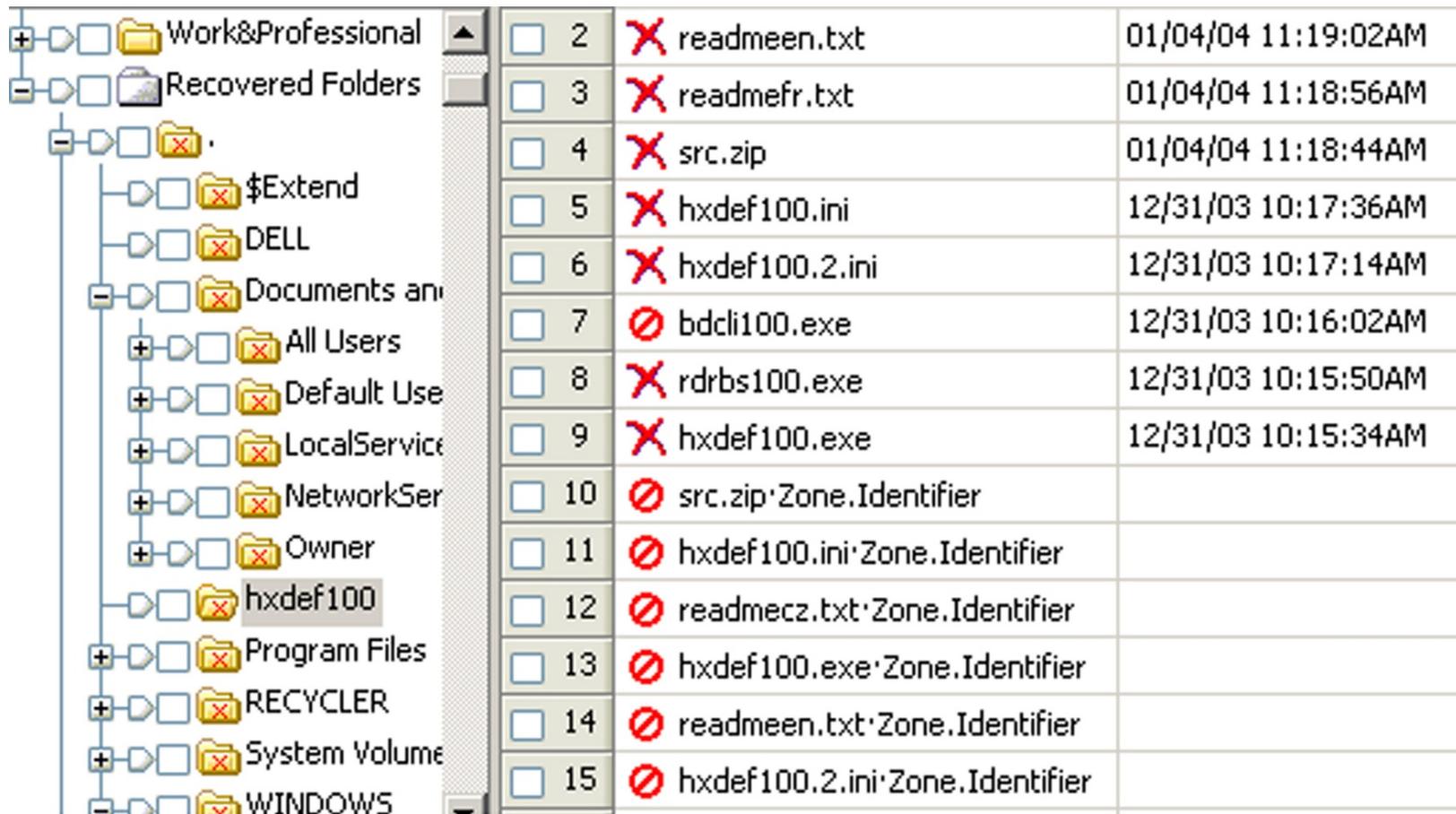
- hiding data due to internal and external fragmentation:
 - hiding within a cluster
 - hiding within the partition (partitions usually begin at the beginning of the trace)
 - hiding partition
- partition encryption
- service data: DCO (*Drive/device configuration overlay*) and HPA (*Host/hidden protected area*) –
http://www.forensicswiki.org/wiki/DCO_and_HPA

Data storage and hiding

- when file is deleted, data does not disappear
- even when we format the disk, the data does not disappear
 - take a look at the tool **fdisk**
- the result of both operations is correct file system and a cluster of empty blocks
- tools: **sleuthkit** (<http://www.sleuthkit.org/>), Norton DiskEdit, ...

Data storage and hiding

- An example of the reconstruction of files on a freshly formatted disk with the EnCase tool



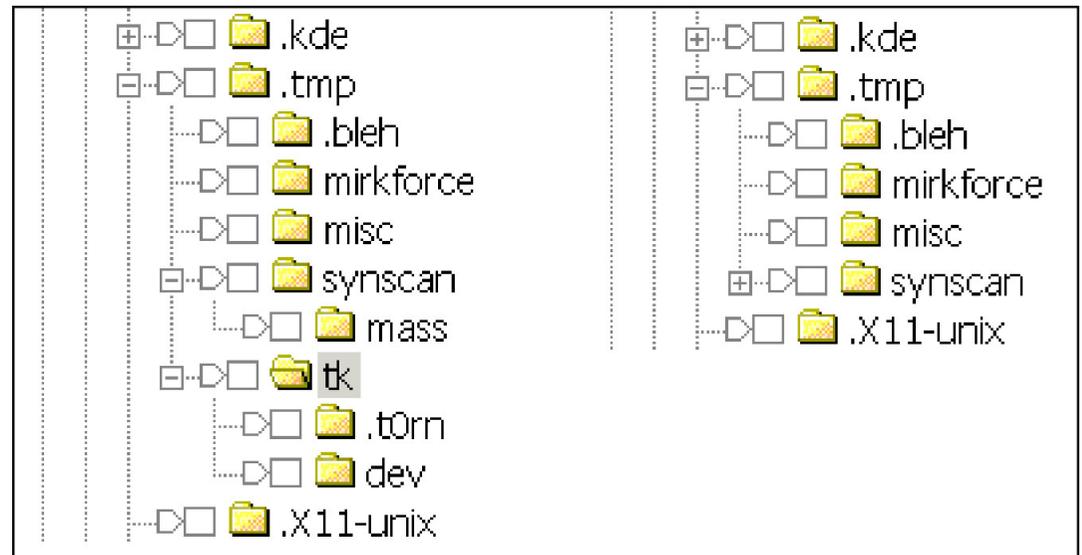
<input type="checkbox"/>	2	✗ readmeen.txt	01/04/04 11:19:02AM
<input type="checkbox"/>	3	✗ readmefr.txt	01/04/04 11:18:56AM
<input type="checkbox"/>	4	✗ src.zip	01/04/04 11:18:44AM
<input type="checkbox"/>	5	✗ hxdef100.ini	12/31/03 10:17:36AM
<input type="checkbox"/>	6	✗ hxdef100.2.ini	12/31/03 10:17:14AM
<input type="checkbox"/>	7	⊘ bdcli100.exe	12/31/03 10:16:02AM
<input type="checkbox"/>	8	✗ rdrbs100.exe	12/31/03 10:15:50AM
<input type="checkbox"/>	9	✗ hxdef100.exe	12/31/03 10:15:34AM
<input type="checkbox"/>	10	⊘ src.zip·Zone.Identifier	
<input type="checkbox"/>	11	⊘ hxdef100.ini·Zone.Identifier	
<input type="checkbox"/>	12	⊘ readmecz.txt·Zone.Identifier	
<input type="checkbox"/>	13	⊘ hxdef100.exe·Zone.Identifier	
<input type="checkbox"/>	14	⊘ readmeen.txt·Zone.Identifier	
<input type="checkbox"/>	15	⊘ hxdef100.2.ini·Zone.Identifier	

Data storage and hiding

- *Challenge: See what the MBR and boot sector on your computer looks like with an appropriate tool. Report about this on the forum.*
- *Challenge: Check the configuration of your drive.*

Data storage and hiding

- hiding partitions
 - tool Test Disk (<http://www.cgsecurity.org/>)
- at file level
 - hiding files: e.g. MS Windows: *attrib +H* in *dir/AH*
 - parliament.jpg -> test.exe
 - picture in .ppt pres.
- the latest tools



Passwords and encryption

- tools for breaking and searching passwords
 - Password Recovery Tool – PRTK in Distributed Network Attack – DNA (<http://accessdata.com/products/computer-forensics/decryption>)
 - John the Ripper (www.openwall.com/john/)
 - Cain and Abel (www.oxid.it/cain.html)
 - Advanced Archive Password Recovery (www.elcomsoft.com/azpr.html)

Passwords and encryption

- more about encryption and cryptography later
- some examples
 - tools caesar, rot13
 - support for the PGP
 - tool crypt

OS Windows

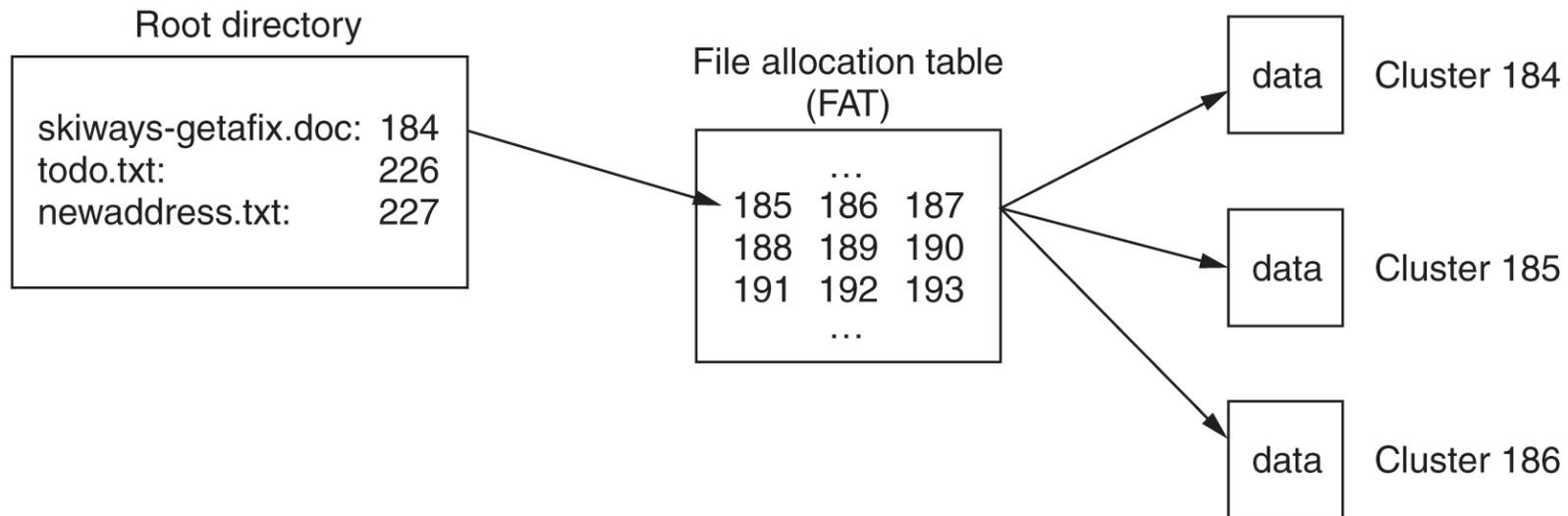
chapter 17

- file systems
- data recovery
- notes (log files)
- register
- communication trails

OS Windows –file system

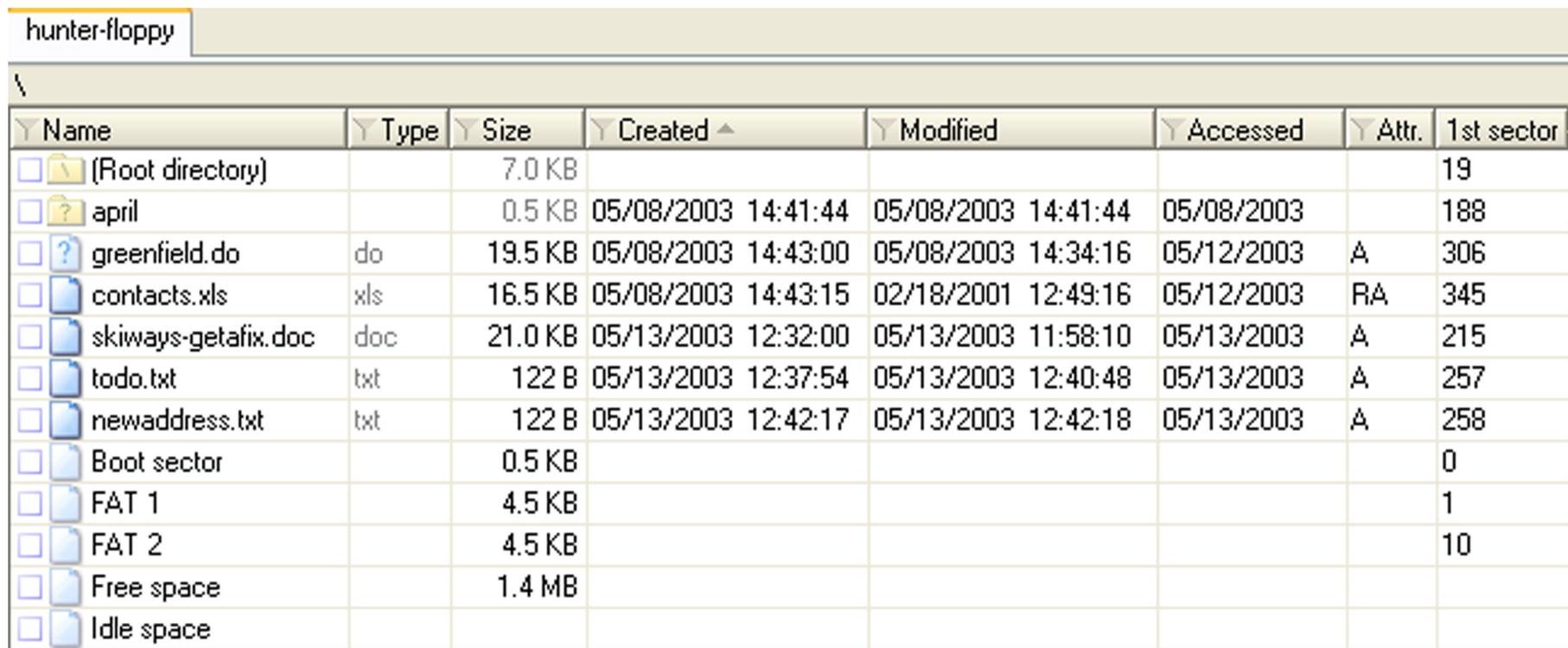
- two basic file systems FAT (*File Allocation Table*) in NTFS (*New Technology File System*)
- FAT
 - developed first for hard disks (floppy disks)
 - FAT12, FAT16, FAT32

File system FAT



- FAT_{xx} is a list of index clusters in which each file is stored
- xx means the number of bits used for the index
- 12 = 2¹² = 4096, 16 = 2¹⁶ = 65.536, 32 = 2³² = 268.435.456

File system FAT



The screenshot shows the X-Ways Forensics interface for a floppy disk named 'hunter-floppy'. The root directory is displayed as a table with columns for Name, Type, Size, Created, Modified, Accessed, Attr., and 1st sector.

Name	Type	Size	Created	Modified	Accessed	Attr.	1st sector
(Root directory)		7.0 KB					19
april		0.5 KB	05/08/2003 14:41:44	05/08/2003 14:41:44	05/08/2003		188
greenfield.do	do	19.5 KB	05/08/2003 14:43:00	05/08/2003 14:34:16	05/12/2003	A	306
contacts.xls	xls	16.5 KB	05/08/2003 14:43:15	02/18/2001 12:49:16	05/12/2003	RA	345
skiways-getafix.doc	doc	21.0 KB	05/13/2003 12:32:00	05/13/2003 11:58:10	05/13/2003	A	215
todo.txt	txt	122 B	05/13/2003 12:37:54	05/13/2003 12:40:48	05/13/2003	A	257
newaddress.txt	txt	122 B	05/13/2003 12:42:17	05/13/2003 12:42:18	05/13/2003	A	258
Boot sector		0.5 KB					0
FAT 1		4.5 KB					1
FAT 2		4.5 KB					10
Free space		1.4 MB					
Idle space							

- view the root of the file system on the hard disk using the X-Ways program
- keeps the creation time and last changes but only the last access date

FAT

The screenshot shows the WinHex application window displaying the FAT table for Drive A:. The interface includes a menu bar (File, Edit, Search, Position, View, Tools, Options, File Manager, Window, Help) and a toolbar. The main display area is a table with the following columns: Offset, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, and Access. The data rows show hexadecimal values for each byte, with some entries containing ASCII characters. The status bar at the bottom indicates 'Sector 1 of 2080', 'Offset 200', and '= 240'.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Access	
00000200	00	FF	FF	00	00	00	00	00	00	00	00	00	00	00	00	00	00	ry.....
00000210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	2B+	
00000240	C0	02	2D	E0	02	2F	00	03	31	20	03	33	40	03	35	60	À.-à./..1 .3@.5'	
00000250	03	37	80	03	39	A0	03	3B	C0	03	3D	E0	03	3F	00	04	.71.9 .;À.-à.?...	
00000260	41	20	04	43	40	04	45	60	04	47	80	04	FF	AF	04	4B	A .C@.E'.01.y~.K	
00000270	C0	04	4D	F0	FF	00	00	00	00	00	00	00	00	00	00	00	À.M5y.....	
00000280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000002F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000300	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000310	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000320	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000330	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000340	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000350	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000360	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000370	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000380	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000390	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000003F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000400	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Sector 1 of 2080 Offset 200 = 240

File system FAT

- *Challenge: See for yourself what the FAT looks like on your disk. Look in particular for those clusters that are empty - they are not part of any file system.*

File system NTFS

- a more modern file system
 - everything is in files
 - the file information is stored in the system file \$MFT
 - directory is only a file (B tree structure)
 - is journal and stores transactions over a file in the system file \$LogFile
- supports multiple file functionality
 - *ACL (Access Control List)*
- better protected, since it stores copies of file system data in multiple locations (\$MFTMirr)

File system NTFS

<i>File Record</i>	<i>Filename</i>	<i>Description</i>
0	\$MFT	Master File Table
1	\$MFTMirr	A backup copy of the first 4 records of the MFT
2	\$LogFile	Log File for CHKDSK
3	\$Volume	Volume Name, Serial Number etc...
4	\$AttrDef	Definitions of every Attribute
5	.(dot)	Root directory of the disk
6	\$Bitmap	Map of used and unused clusters
7	\$Boot	Boot record of the volume
8	\$BadClus	List of bad clusters on the partition
9	\$Secure	Security Descriptors for each file
10	\$UpCase	Table of uppercase characters used for conversion
11	\$Extend	Directory for the last four Metafiles.
12-23	UNUSED	Marked in use, or not in use, but empty.
Any	\$ObjId	Unique Object IDs given to every file
Any	\$Quota	Disk space usage quota information
Any	\$Reparse	Reparse point information
Any	\$UsnJrnl	NTFS USN Journal (for encryption)

Table 3.1.1 - NTFS 3.0+ Metafiles

File system NTFS

- *Challenge: look for journals in your NTFS journals that are empty (unused) and then look at their content.*

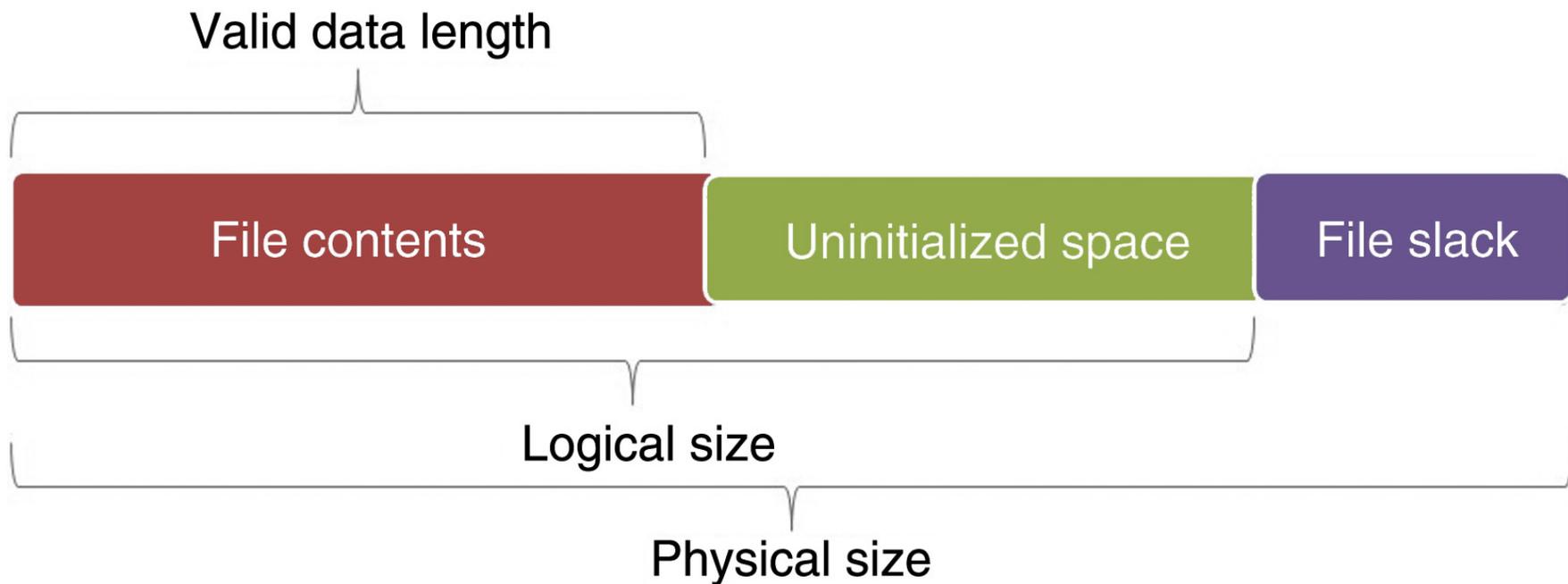
NTFS – \$MFT

- example of one record in \$MFT
- the record consists of attributes, the record is the size of the 1kB
- if the file is small, it is stored in the record
- when the flag is deleted, then the record is reused

```
Pointed to by file:  
E:\review.pgd  
File Type:  
data  
MD5 of content:  
19d3508b078a10b3852b75f46ef9be5a  
SHA-1 of content:  
3229c020dcbd2c38ba44c462c1970cbc13db473b  
Details:  
MFT Entry Header Values:  
Entry: 29 Sequence: 1  
$LogFile Sequence Number: 16842551  
Allocated File  
Links: 1  
  
$STANDARD_INFORMATION Attribute Values:  
Flags: Archive  
Owner ID: 0 Security ID: 260  
Created: Tue Mar 6 21:24:51 2007  
File Modified: Wed Mar 7 19:16:13 2007  
MFT Modified: Wed Mar 7 19:16:13 2007  
Accessed: Wed Mar 7 19:16:13 2007  
  
$FILE_NAME Attribute Values:  
Flags: Archive  
Name: review.pgd  
Parent MFT Entry: 5 Sequence: 5  
Allocated Size: 0 Actual Size: 0  
Created: Tue Mar 6 21:24:51 2007  
File Modified: Tue Mar 6 21:24:51 2007  
MFT Modified: Tue Mar 6 21:24:51 2007
```

NTFS - search for data

- there is a physical file size (cluster), logical size (directory entry) and the end of the file (EOF)



NTFS – MFT record

- MFT record and the difference between sizes

0C07F5000	46 49 4C 45 30 00 03 00	31 43 0C 8F 00 00 00 00	FILE0	1C	!					
0C07F5010	03 00 02 00 38 00 01 00	E0 01 00 00 00 04 00 00	8	à						
0C07F5020	00 00 00 00 00 00 00 00	05 00 00 00 D4 1F 00 00		Ô						
0C07F5030	02 00 00 00 00 00 00 00	10 00 00 00 60 00 00 00		,						
0C07F5040	00 00 00 00 00 00 00 00	48 00 00 00 18 00 00 00		H						
0C07F5050	48 08 C6 77 A8 C5 CA 01	48 08 C6 77 A8 C5 CA 01	H	Æw'ÂÊ	H	Æw'ÂÊ				
0C07F5060	48 08 C6 77 A8 C5 CA 01	28 D3 9A 7A A8 C5 CA 01	H	Æw'ÂÊ	(Ó	!z'ÂÊ				
0C07F5070	20 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00								
0C07F5080	00 00 00 00 69 01 00 00	00 00 00 00 00 00 00 00		i						
0C07F5090	00 00 00 00 00 00 00 00	30 00 00 00 70 00 00 00		0	p					
0C07F50A0	00 00 00 00 00 00 03 00	52 00 00 00 18 00 01 00		R						
0C07F50B0	E6 24 00 00 00 00 01 00	48 08 C6 77 A8 C5 CA 01	æ\$	H	Æw'ÂÊ					
0C07F50C0	48 08 C6 77 A8 C5 CA 01	48 08 C6 77 A8 C5 CA 01	H	Æw'ÂÊ	H	Æw'ÂÊ				
0C07F50D0	48 08 C6 77 A8 C5 CA 01	00 00 00 00 00 00 00 00	H	Æw'ÂÊ						
0C07F50E0	00 00 00 00 00 00 00 00	20 00 00 00 00 00 00 00								
0C07F50F0	08 02 43 00 4D 00 44 00	4C 00 41 00 42 00 7E 00		C	M	D	L	A	B	~
0C07F5100	32 00 73 00 65 00 74 00	30 00 00 00 88 00 00 00	2	s	e	t	0	!		
0C07F5110	00 00 00 00 00 00 02 00	6A 00 00 00 18 00 01 00		j						
0C07F5120	E6 24 00 00 00 00 01 00	48 08 C6 77 A8 C5 CA 01	æ\$	H	Æw'ÂÊ					
0C07F5130	48 08 C6 77 A8 C5 CA 01	48 08 C6 77 A8 C5 CA 01	H	Æw'ÂÊ	H	Æw'ÂÊ				
0C07F5140	48 08 C6 77 A8 C5 CA 01	00 00 00 00 00 00 00 00	H	Æw'ÂÊ						
0C07F5150	00 00 00 00 00 00 00 00	20 00 00 00 00 00 00 00								
0C07F5160	14 01 63 00 6D 00 64 00	4C 00 61 00 62 00 73 00		c	m	d	L	a	b	s
0C07F5170	2D 00 73 00 65 00 74 00	76 00 61 00 6C 00 69 00	-	s	e	t	v	a	l	i
0C07F5180	64 00 64 00 61 00 74 00	61 00 00 00 00 00 00 00	d	d	a	t	a			
0C07F5190	80 00 00 00 48 00 00 00	01 00 00 00 00 00 04 00	!	H						
0C07F51A0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00								
0C07F51B0	40 00 00 00 00 00 00 00	00 10 00 00 00 00 00 00	@							
0C07F51C0	00 04 Logical Size	00 00 00 E8 03 Valid Data Length		è						
0C07F51D0	31 01 CE AB 03 00 01 00	FF FF FF FF 82 79 47 11	1	!	<<	ÿÿÿÿ	!	y	G	

NTFS - search for data

- In one directory we can have multiple files with the same name

File system NTFS

- *Challenge: Which Clusters Compose Your File?*
- *Challenge: Find a busy but unused part of your file (on which clusters) and what's in it.*
- *Challenge: What happens if we make 1000 files, then we delete 1000 and work on it?*

Time coding for files

- FAT: 1.1.1980 + LLLLLLLM MMMDDDDD hhhhhmmm mmmsssss

Volume	File	Preview	Details	Gallery	Calendar	Legend	Search	Sync									
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00002600	53	41	4C	45	53	20	20	20	20	20	20	28	00	00	00	00	SALES (
00002610	00	00	00	00	00	00	9A	7C	8D	2E	00	00	00	00	00	00	.
00002620	42	69	00	78	00	2E	00	64	00	6F	00	0F	00	F1	63	00	Bi x . d o ñc
00002630	00	00	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy yyyy
00002640	01	73	00	6B	00	69	00	77	00	61	00	0F	00	F1	79	00	s k i w a ñy
00002650	73	00	2D	00	67	00	65	00	74	00	00	00	61	00	66	00	s - g e t a f
00002660	53	4B	49	57	41	59	7E	31	44	4F	43	20	00	0A	00	64	SKIWAY~1DOC d
00002670	AD	2E	AD	2E	00	00	45	5F	AD	2E	B8	00	00	54	00	00	-.-. E_-. T
00002680	41	74	00	6F	00	64	00	6F	00	2E	00	0F	00	B3	74	00	At o d o . ?t
00002690	78	00	74	00	00	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	x t yyyy yyyy
000026A0	54	4F	44	4F	20	20	20	20	54	58							TODO TXT >>d
000026B0	AD	2E	AD	2E	00	00	18	65	AD	2E							-.-. e-â z
000026C0	42	74	00	00	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	Bt yyyyyy yy
000026D0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy yyyy
000026E0	01	6E	00	65	00	77	00	61	00	64	00	0F	00	8C	64	00	n e w a d d
000026F0	72	00	65	00	73	00	73	00	2E	00	00	00	74	00	78	00	r e s s . t x
00002700	4E	45	57	41	44	44	7E	31	54	58	54	20	00	85	48	65	NEWADD~1TXT He

Data Interpreter ✖

DOS Date: 05/13/2003
11:58:10

Time coding for files

- FILETIME
- 64 bit record
 - value = 1.1.1600 + number * 100ns



NTFS - tracks files

- various operations have a different impact on the recorded times in the directory (creation - CR, last access - LA, last change - LC, record changed (NTFS) - RC):
 - moving the file into a directory: it does not affect anything
 - moving the file to another directory: CR, LA, RC
 - copy file (target file): CR, LA, RC
 - copy/paste: LA(*)
 - *drag&drop*: LA(*)
 - delete: LA, RC
- special features:
 - file on a stick, can be via scp/...: CR > LC
 - when deleting a directory, file information does not change

NTFS - tracks files ...

- the content of office files contains metadata from the directory
 - *Save as: if an existing file is picked, the data in the file is overwritten and no new file is created in the directory*
- printing first copies the file to a special directory and then prints it
 - *C:\Windows\Spool\Printers, C:\WinNT\System32\Spool\Printers*
 - even when we print online content, etc.

NTFS - tracks files ...

- *Challenge: Find a file that has a creation time greater than the time of the last change.*
- Challenge: What can you say, is there such a file on the system that has the last access time same at the time of the creation?
- Challenge: What is the EMF printing method ? What is stored in the print file (spooler)?

Data recovery

- recover deleted files
 - various tools that can run on WinOS

- SleuthKit combined with Autopsy Browser can even browse through the browser (<http://www.sleuthkit.org/autopsy/>)

The screenshot shows the Autopsy browser interface. The main window displays a file analysis report for the C:\ directory. The report is organized into a table with the following columns: DEL, Type, NAME, WRITTEN, ACCESSED, CREATED, SIZE, UID, GID, and META. The table lists several deleted files, including directories like APE_A=1.DIR, COM_SW/, and MSSTQF.T/, and files like AUTOEXEC.BAT and AUTOEXEC.SYD. The file AUTOEXEC.BAT is selected, and its contents are displayed in a text area below the table. The contents of the file are:

```
C:\DEV\TEAC\MSCDEX.EXE /D:TEAC-CDI /M:15
C:\DEV\MOUSE\BALL.COM
```

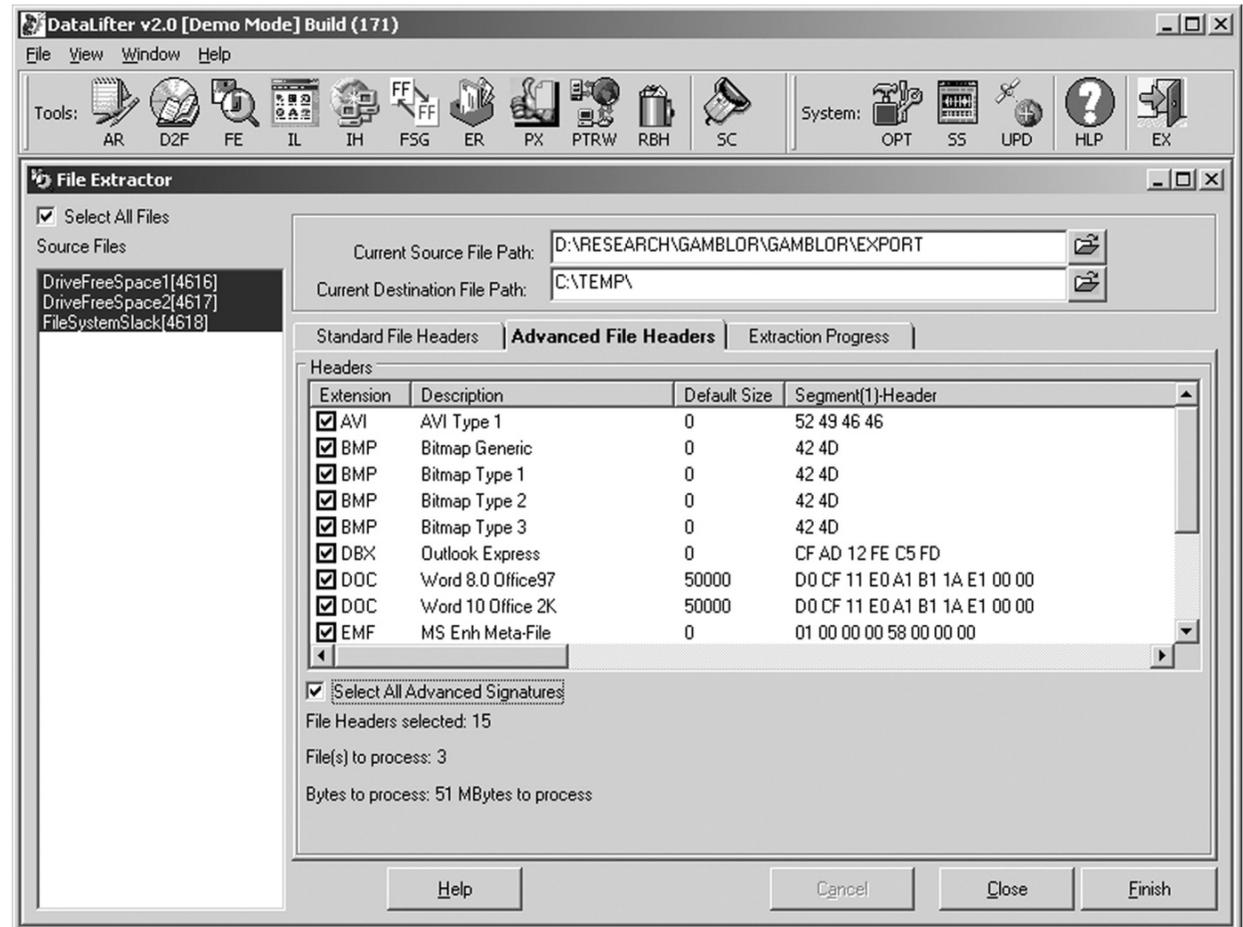
The interface also includes a sidebar with a file tree, a top menu bar, and a status bar at the bottom showing the document name and processing time.

Data recovery ...

- *Challenge: install sleuthkit and Autopsy Browser and find the lost files.*

Data recovery ...

- searching for lost files from a large unformatted mound
 - same as curving files
- tool DataLifter:
looks for a lost file from two empty spaces and one of the rest of the file system



Log files

- the operating system (depending on the settings) records
 - access to resources
 - appearance and deletion of resources,
 - errors, etc.
- saved on *%systemroot%\system32\config (c:\winnt\...)*
 - different notes in different files: *Appevent.evt, Secevent.evt, Sysevent.evt*

Log files

- Challenge: check the format of the evt file and check what is in them and when did you logged in to the system.

Register

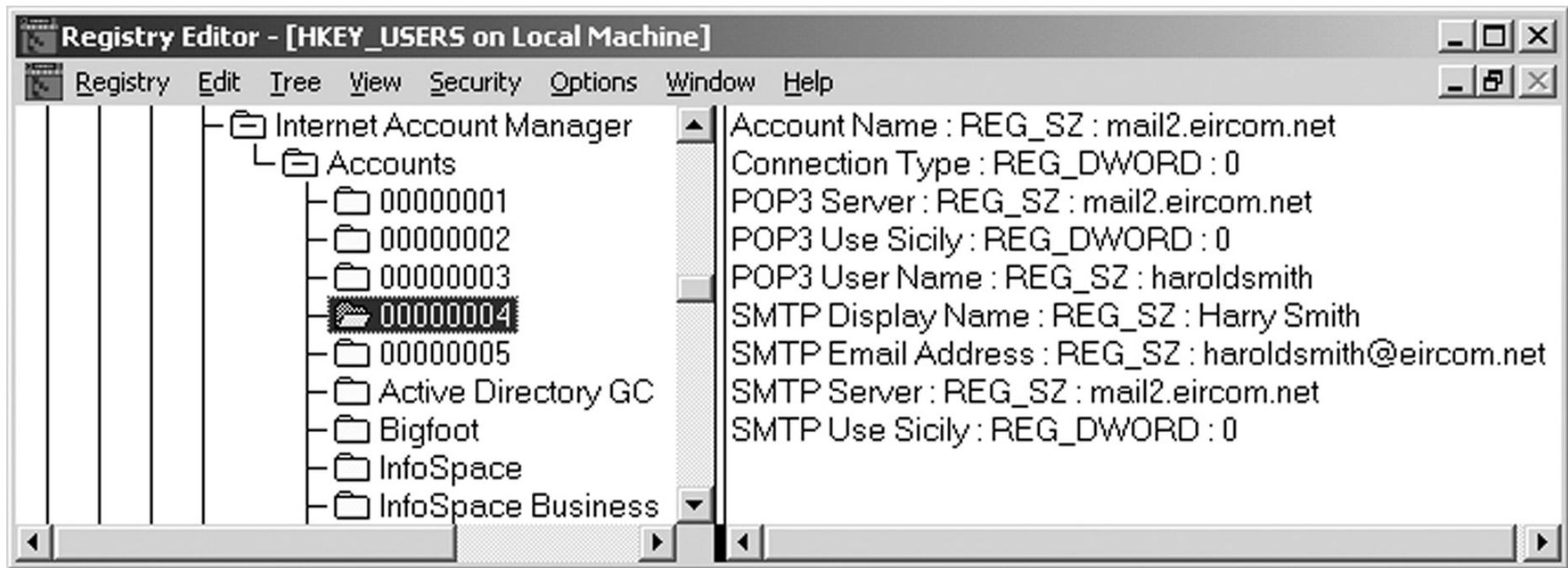
- In Windows OS, the process environment variables are defined in the registers
- actually, the data is stored in the files (hives) in the system directory *%systemroot%\system32\config*
 - *ntuser.dat* for each user account
- files can be viewed with the Windows tool regedt32 (EnCase, FTK, ...)

Register

- *Challenge: examine the forensic value of the data in the registry.*

Network tracking

- sometimes from the system environment
 - when connecting, ...
- mostly comes directly from application
 - browsers, mail agents, ...

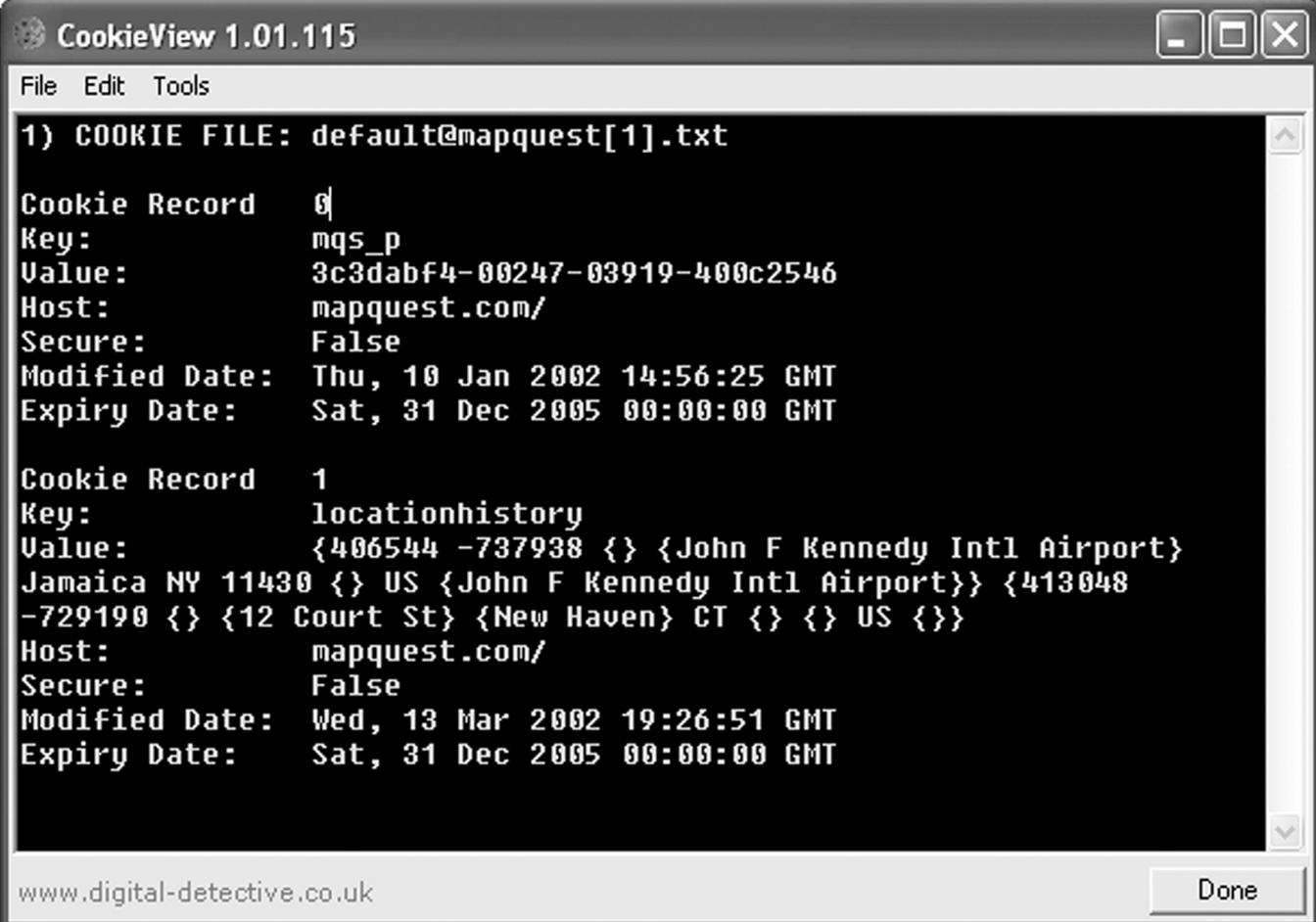


Network Tracking - Browsers

- history:
 - firefox-3 is storing history in the sqlite databases *Places.sqlite*
 - Internet Explorer stores history in the file *index.dat*
 - tools that are available to search through these databases: *Odessa* (www.odessa.sourceforge.net)
- local cache
- cookies

Browsers - Cookies

- example of cookies inspection in CookieView (www.digitaldetective.co.uk)



```
CookieView 1.01.115
File Edit Tools
1) COOKIE FILE: default@mapquest[1].txt
Cookie Record 0
Key: mqs_p
Value: 3c3dabf4-00247-03919-400c2546
Host: mapquest.com/
Secure: False
Modified Date: Thu, 10 Jan 2002 14:56:25 GMT
Expiry Date: Sat, 31 Dec 2005 00:00:00 GMT

Cookie Record 1
Key: locationhistory
Value: {406544 -737938 {} {John F Kennedy Intl Airport}
Jamaica NY 11430 {} US {John F Kennedy Intl Airport}} {413048
-729190 {} {12 Court St} {New Haven} CT {} {} US {}
Host: mapquest.com/
Secure: False
Modified Date: Wed, 13 Mar 2002 19:26:51 GMT
Expiry Date: Sat, 31 Dec 2005 00:00:00 GMT

www.digital-detective.co.uk Done
```

Browsers

- *Challenge: Find out what leftovers you do have in your cache and check with your browsing history.*
- *Challenge: Get a file from your friend's browser history and disassemble it.*
- *Challenge: Check out what kind of traces are left behind by the IE browser, what kind by the Mozilla and what kind by the Opera.*

E-mail

- Traces depend on the mail agent we use
 - sent and received mails
 - summary of IMAP mailbox
- content that is interesting
 - text mails only
 - attachments (!) – MIME format

Other programs

- different programs leave different traces
- network software
 - access to other systems
 - allow other systems to access in our system
- system programs leave traces in the registry

Network access tracking

- telnet access to acf2.nyu.edu

