

# File Signature Analysis

Dr Syed Naqvi

[syed.naqvi@bcu.ac.uk](mailto:syed.naqvi@bcu.ac.uk)



# Outline

- Introduction
- Files, common file types and file signatures
- File signature analysis using EnCase

# Specific type of search

- File signature analysis is a specific type of search used to check files are what they report to be by the file system.
- Files indicate their *type* and consequently their contents through the filename **extension** on MS Windows operating systems. Extensions are only a **convention**.
- In both FAT and NTFS file systems, other OSs don't rely on extensions as much.
- Extension is a set of characters at the end of the file name separated by the final dot, . , from the filename.
- Usual to find a three character extension 'code' to indicate what type of file it is.
- A legacy from early FAT file systems on MS DOS where a filename is a maximum of 11 characters in total, 8 for the filename and 3 characters for the extension.

# Filenames and extensions

- FAT12/16 file systems only permits 11 character filenames.
- Directory structure recording the filename and extension, timestamps and importantly the first cluster allocated to the file.
- Directory structure is 32 bytes in length hence space for filenames is limited.
- FAT32 works in the same way except each entry in the FAT is 32 bits instead of 16 bits.
- Also, can have long filenames up to 256 characters where multiple directory entries are used when filename is in excess of 11 characters and are in UNICODE.
- Does this by using multiple 32 byte directory entries.

# FAT16 Directory Structure



File as shown by Windows Explorer

```

46 49 4C 45 4E 41 4D 45 45 58 54 20 18 AC 5B 6F | FILENAMEEXT  -[o
51 3F 51 3F 00 00 23 71 51 3F 00 00 00 00 00 | Q?Q?  #qQ?

```

File as stored by file system. First 11 bytes are the filename and extension. Other 21 bytes are timestamps, length (0) bytes and first cluster number allocated to the file.

Important to note that the . is not stored in filename as filename is a fixed structure where last three characters are interpreted as the extension

# Long file names

Name ▲	Ext.	Size	Created	Modified
(Root directory)		4.0 KB		
This is a long file name for a.jpg	jpg	8.0 KB	07/04/2011 10:49:20	07/04/2011 10:49:22
Boot sector		17.0 KB		
FAT 1		8.0 MB		
FAT 2		8.0 MB		
Free space		8.0 GB		

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0016756736	46	41	54	33	32	20	20	20	20	20	20	08	00	00	00	00	FAT32
0016756752	00	00	00	00	00	00	20	A6	4C	43	00	00	00	00	00	00	IC
0016756768	43	6F	00	72	00	20	00	61	00	2E	00	0F	00	B0	6A	00	Co r a . ^j
0016756784	70	00	67	00	00	00	FF	FF	FF	FF	00	00	FF	FF	FF	FF	p g yyy yyy
0016756800	02	67	00	20	00	66	00	69	00	6C	00	0F	00	B0	65	00	g f i l ^e
0016756816	20	00	6E	00	61	00	6D	00	65	00	00	00	20	00	66	00	n a m e f
0016756832	01	54	00	68	00	69	00	73	00	20	00	0F	00	B0	69	00	T h i s ^i
0016756848	73	00	20	00	61	00	20	00	6C	00	00	00	6F	00	6E	00	s a l o n
0016756864	54	48	49	53	49	53	7E	31	4A	50	47	20	00	3C	2A	56	THISIS~1JPG < *V

Dot

- FAT32 introduced long filenames of up to 255 characters.
- Makes use of multiple 32 byte directory entries in order from end to start of filename
- Includes the “.” character.

# NTFS

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
00029696	46	49	4C	45	30	00	03	00	39	1A	00	01	00	00	00	00	FILE0 9
00029712	01	00	02	00	38	00	01	00	00	02	00	00	00	04	00	00	8
00029728	00	00	00	00	00	00	00	00	05	00	00	00	1D	00	00	00	G
00029744	02	00	47	11	00	00	00	00	10	00	00	00	60	00	00	00	H
00029760	00	00	00	00	00	00	00	00	48	00	00	00	18	00	00	00	öaz  öE öaz  öE
00029776	F6	E4	7A	96	10	F5	CB	01	F6	E4	7A	96	10	F5	CB	01	öaz  öE öaz  öE
00029792	F6	E4	7A	96	10	F5	CB	01	F6	E4	7A	96	10	F5	CB	01	öaz  öE öaz  öE
00029808	20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00029824	00	00	00	00	04	01	00	00	00	00	00	00	00	00	00	00	
00029840	00	00	00	00	00	00	00	00	30	00	00	00	78	00	00	00	0 x
00029856	00	00	00	00	00	00	03	00	5A	00	00	00	18	00	01	00	Z
00029872	05	00	00	00	00	00	05	00	F6	E4	7A	96	10	F5	CB	01	öaz  öE
00029888	F6	E4	7A	96	10	F5	CB	01	F6	E4	7A	96	10	F5	CB	01	öaz  öE öaz  öE
00029904	F6	E4	7A	96	10	F5	CB	01	00	00	00	00	00	00	00	00	öaz  öE
00029920	00	00	00	00	00	00	00	00	20	00	00	00	00	00	00	00	
00029936	0C	02	54	00	48	00	49	00	53	00	49	00	53	00	7E	00	T H I S I S ~
00029952	31	00	2E	00	4A	00	50	00	47	00	6E	00	67	00	20	00	1 . J P G n g
00029968	30	00	00	00	A0	00	00	00	00	00	00	00	00	00	02	00	0
00029984	84	00	00	00	18	00	01	00	05	00	00	00	00	00	05	00	!
00030000	F6	E4	7A	96	10	F5	CB	01	F6	E4	7A	96	10	F5	CB	01	öaz  öE öaz  öE
00030016	F6	E4	7A	96	10	F5	CB	01	F6	E4	7A	96	10	F5	CB	01	öaz  öE öaz  öE
00030032	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00030048	20	00	00	00	00	00	00	00	21	01	54	00	68	00	69	00	! T h i
00030064	73	00	20	00	69	00	73	00	20	00	61	00	20	00	6C	00	s i s a l
00030080	6F	00	6E	00	67	00	20	00	66	00	69	00	6C	00	65	00	o n g f i l e
00030096	6E	00	61	00	6D	00	65	00	20	00	66	00	6F	00	72	00	n a m e f o r
00030112	20	00	61	00	2E	00	6A	00	70	00	67	00	00	00	00	00	a . j p g

- Long and short filenames stored at \$FILENAME attribute.
- Includes '/' in both cases.

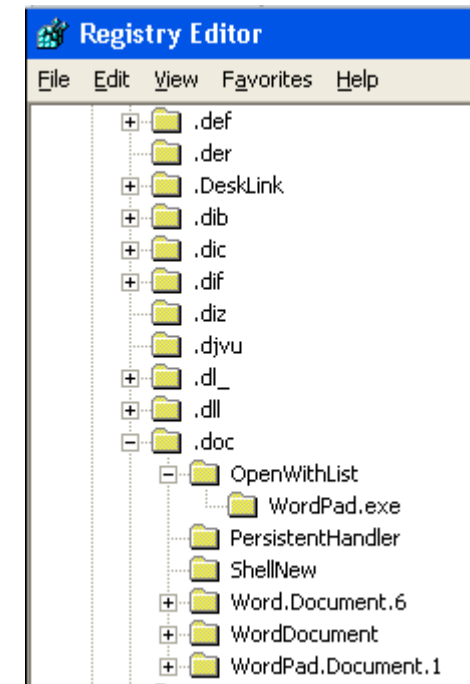
# File extensions

- Over time, file extensions have been created to help users identify files on a storage device.
- File1 is a poor name for a file.
- File1.jpg is still not good but have an idea about the contents.
- Extensions are arbitrary where ones around today have become de facto.
- Extensions are associated with a type of file and also a program that can view and/or edit the data in the file.
- MS Windows OS has a list of associations between extension and application program so that double clicking on the file in Explorer starts the program associated with the extension and loads the file ready to be viewed/edited.



# Windows Extension Associations

- Stored in Registry under HK\_CLASSES\_ROOT.
- Many of the keys have the extension as the name, e.g. .doc, .dll, .exe, .case and so on.
- Have an application associated with the extension that is used to open the file in programs such as File Explorer.
- Extension is bound to the application or applications.



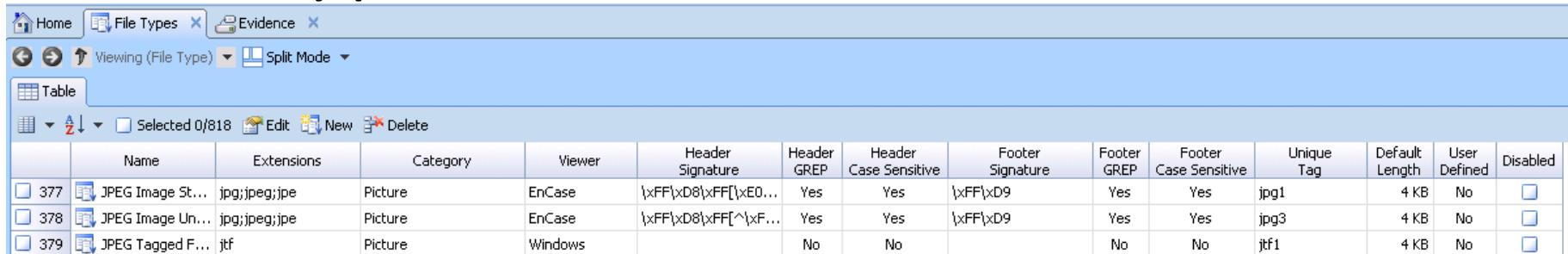
# Common extensions (MS Windows)

- Common: jpg, png, gif, bmp, doc, xls, ppt, mp3, mp4, aac, wav, pdf, zip.
- Common but not so obvious: exe, com, dll, html, css, txt, js, cab, chm, cur, fon, hlp, inf, lnk, pnf, sys, tlb.
- Key distinction is that common extensions are found on user created content/data.
- Common but not so obvious are found on system files.

# Quick exercise

- Find out what types of files the following extensions refer:
  - JPG
  - EXE
  - CAB
  - DOCX
  - LNK
  - INF
- Find out if the file associated with the extension is viewable/editable by the user and what program would be used.

# File Types in EnCase v7



The screenshot shows the 'File Types' window in EnCase v7. The window title is 'File Types' and it contains a table with the following data:

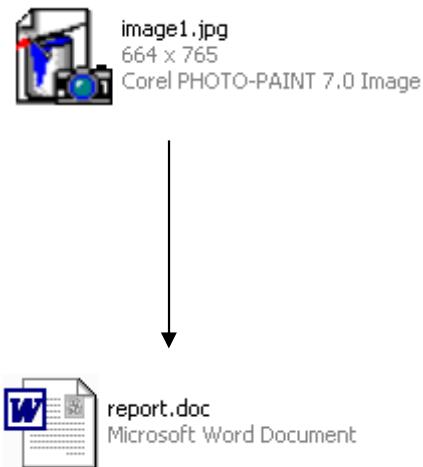
	Name	Extensions	Category	Viewer	Header Signature	Header GREP	Header Case Sensitive	Footer Signature	Footer GREP	Footer Case Sensitive	Unique Tag	Default Length	User Defined	Disabled	
<input type="checkbox"/>	377	JPEG Image St...	.jpg;.jpeg;.jpe	Picture	EnCase	{\xFF\xD8}{\xFF[\xE0...	Yes	Yes	{\xFF\xD9	Yes	Yes	jpg1	4 KB	No	<input type="checkbox"/>
<input type="checkbox"/>	378	JPEG Image Un...	.jpg;.jpeg;.jpe	Picture	EnCase	{\xFF\xD8}{\xFF[^{\xF...	Yes	Yes	{\xFF\xD9	Yes	Yes	jpg3	4 KB	No	<input type="checkbox"/>
<input type="checkbox"/>	379	JPEG Tagged F...	.jtf	Picture	Windows		No	No		No	No	jtf1	4 KB	No	<input type="checkbox"/>

- File Type defined in terms
  - Extension or extensions for the type.
  - Category such as Picture or Document.
  - Header signature.
  - Optional footer signature.
- Note that if a File Type is missing a Header Signature then all it defines is extension, EnCase not able to determine if signature is correct for extension.

# Problems file extensions pose for investigator

- Extensions are a convention only and not enforced by the file system or OS (Windows).
- Association between extension and file content is 'loosely' defined in MS Windows.
- Easy for user to change extension using Windows File Explorer if they know what they are doing.
- Rudimentary form of data obfuscation, change the extension to give a false impression of what the file is.
- Cannot rely on extension as indicating the true content of the file.

# Example



Simple task of clicking on filename and typing in a new filename and extension or right click file and select Rename

# Identification of file contents

- Fortunately, changing a file's extension is a weak form of data obfuscation.
- As it does not effect the file contents, that is an JPEG image file will still contain the image data after renaming.
- Usefully file formats have a structure we can make use of.
- Typically file **header** contains metadata about the file to help a software application validate the file content is correct, load the data and view or process the data.
- An important element is a **magic number** typically at the start of the file and present in files of that type.
- Magic number allows a program to initially determine if the file is valid by first checking presence of number.
- However magic numbers are not rigorously defined by any organisation such as Microsoft.

# File signatures

- File signature is a sequence of bytes that is used by application programs to confirm file data before loading and processing the rest of the file.
- Often quoted example in computer forensics is JPEG byte sequence 0xFF 0xD8 0xFF 0xE0, appear as the characters `ÿØÿà` when JPEG file viewed in a hex/ASCII editor.
- Reason for these values is that the JPEG file format is based on storing image data plus commands to tell the image viewer program what to do, 0xFF means command where the next number is the command where 0xD8 means 'Start of Image'.
- No standards exist where the selection of bytes for signature is arbitrary or application specific such as JPEG.
- Good source for file signatures is [http://www.garykessler.net/library/file\\_sigs.html](http://www.garykessler.net/library/file_sigs.html).



# Examples of file signatures

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	25	50	44	46	2D	31	2E	32	0D	25	E2	E3	CF	D3	0D	0A

PDF-1.2 %äÿÏÓ

PDF file

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	52	49	46	46	32	00	00	00	57	41	56	45	66	6D	74	20

RIFF2 WAVEfmt

WAV audio file

Above signatures are based on the acronyms of the files, PDF and Resource Interchangeable File Format

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00

MZÿÿ

Executable program file. Why MZ?  
Because file format was developed by Mark Zbikowski

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	FF	D8	FF	E0	00	10	4A	46	49	46	00	01	01	01	00	60

ÿÿà JFIF

JPEG image file

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	D0	CF	11	E0	A1	B1	1A	E1	00	00	00	00	00	00	00	00

Ëÿ à± á

MS Word document