Intro to Steganography

Hiding things in plain sight

Hiding text in an image

Many ways to find hidden texts in images:

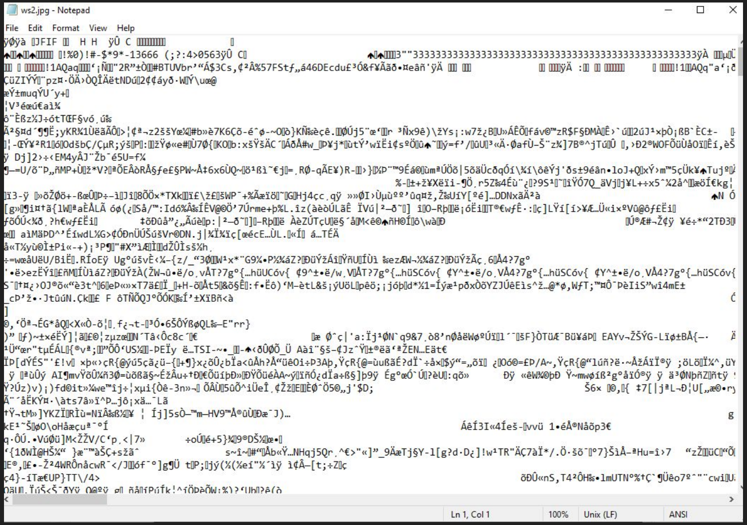
- Brightness / Contrast
- Levels
- Squinting really hard at it until you see it
- Stegsolve

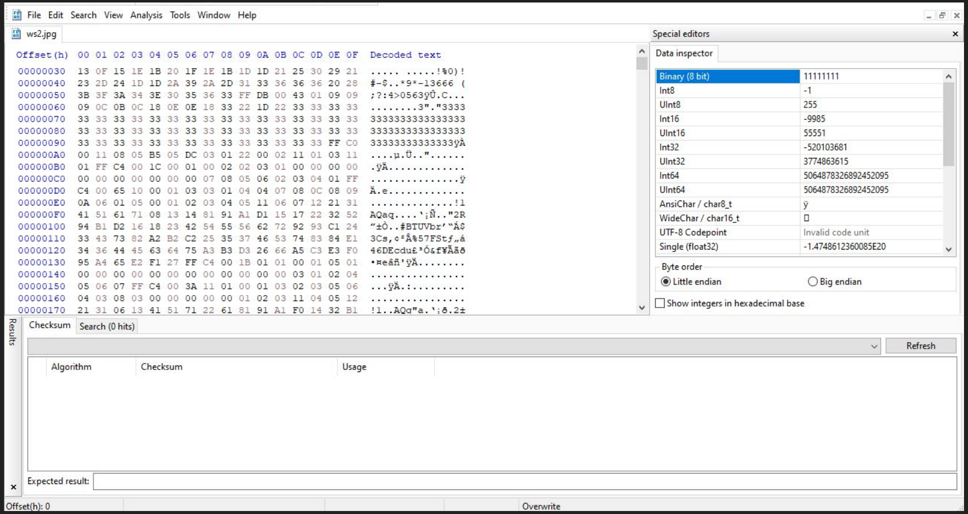
Demo: latin.png

Text / Hex Editor

Text sees in ascii (only shows printable characters), hex shows hex of each byte (and also ascii)

Text vs hex





```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000030  13 0F 15 1E 1B 20 1F 1E 1B 1D 1D 21 25 30 29 21  ..... .....!%0)!
00000040  23 2D 24 1D 1D 2A 39 2A 2D 31 33 36 36 36 20 28  #-$..*9*-13666 (
00000050  3B 3F 3A 34 3E 30 35 36 33 FF DB 00 43 01 09 09  ;?:4>0563ÿÛ.C...
00000060  09 0C 0B 0C 18 0E 0E 18 33 22 1D 22 33 33 33 33  ........3"."3333
00000070  33 33 33 33 33 33 33 33 33 33 33 33 33 33 33 33  3333333333333333
00000080  33 33 33 33 33 33 33 33 33 33 33 33 33 33 33 33  3333333333333333
00000090  33 33 33 33 33 33 33 33 33 33 33 33 33 33 FF C0  33333333333333ÿÀ
000000A0  00 11 08 05 B5 05 DC 03 01 22 00 02 11 01 03 11  ....µ.Ü.."......
000000B0  01 FF C4 00 1C 00 01 00 02 02 03 01 00 00 00 00  .ÿÄ.............
000000C0  00 00 00 00 00 00 00 07 08 05 06 02 03 04 01 FF  ...............ÿ
000000D0  C4 00 65 10 00 01 03 03 01 04 04 07 08 0C 08 09  Ä.e.............
000000E0  0A 06 01 05 00 01 02 03 04 05 11 06 07 12 21 31  ..............!1
000000F0  41 51 61 71 08 13 14 81 91 A1 D1 15 17 22 32 52  AQaq....'¡Ñ.."2R
00000100  94 B1 D2 16 18 23 42 54 55 56 62 72 92 93 C1 24  ”±Ò..#BTUVbr'“Á$
00000110  33 43 73 82 A2 B2 C2 25 35 37 46 53 74 83 84 E1  3Cs,¢²Â%57FStƒ„á
00000120  34 36 44 45 63 64 75 A3 B3 D3 26 66 A5 C3 E3 F0  46DEcdu£³Ó&f¥Ããð
00000130  95 A4 65 E2 F1 27 FF C4 00 1B 01 01 00 01 05 01  •¤eâñ'ÿÄ........
00000140  00 00 00 00 00 00 00 00 00 00 00 00 03 01 02 04  ................
00000150  05 06 07 FF C4 00 3A 11 01 00 01 03 02 03 05 06  ...ÿÄ.:.........
00000160  04 03 08 03 00 00 00 00 00 01 02 03 11 04 05 12  ................
00000170  21 31 06 13 41 51 71 22 61 81 91 A1 F0 14 32 B1  !1..AQq"a.'¡ð.2±
```

Binary (8 bit)	11111111
Int8	-1
UInt8	255
Int16	-9985
UInt16	55551
Int32	-520103681
UInt32	3774863615
Int64	5064878326892452095
UInt64	5064878326892452095
AnsiChar / char8_t	ÿ
WideChar / char16_t	▯
UTF-8 Codepoint	Invalid code unit
Single (float32)	-1.4748612360085E20

Byte order

● Little endian ○ Big endian

□ Show integers in hexadecimal base

Algorithm | Checksum | Usage

Expected result:

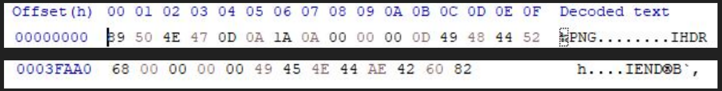
Offset(h): 0

File extensions

- .pdf, .mp3, .docx, .jpg

Every file has “magic numbers” (file headers / signatures) that can be seen when you open them in a hex editor.

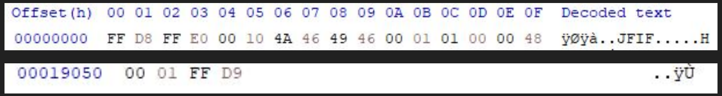
PNG File



```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000   89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52  ‰PNG........IHDR

0003FAA0   68 00 00 00 00 49 45 4E 44 AE 42 60 82              h....IEND®B`‚
```

JPG File



```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text
00000000   FF D8 FF E0 00 10 4A 46 49 46 00 01 01 00 00 48  ÿØÿà..JFIF.....H

00019050   00 01 FF D9                                                  ..ÿÙ
```

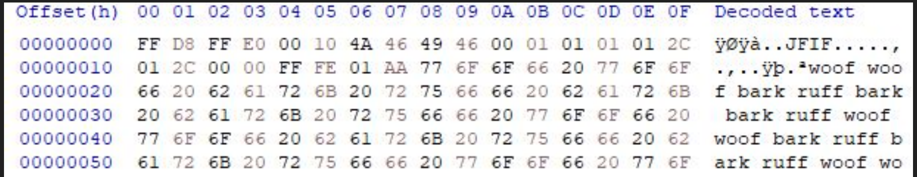
List of files and their file signatures: https://www.garykessler.net/library/file_sigs.html

The `file` command on linux (as well as binwalk) tells you what file you are dealing with

Hiding something *in* the file

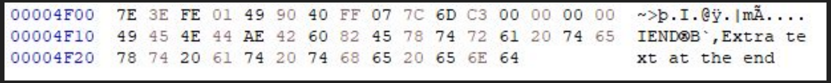
Some extra information usually found in the end, but can be in the middle or even towards the beginning

(From TAMUCTF 2020)



```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00000000  FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 01 2C  ÿØÿà..JFIF.....,
00000010  01 2C 00 00 FF FE 01 AA 77 6F 6F 66 20 77 6F 6F  .,..ÿþ.ªwoof woo
00000020  66 20 62 61 72 6B 20 72 75 66 66 20 62 61 72 6B  f bark ruff bark
00000030  20 62 61 72 6B 20 72 75 66 66 20 77 6F 6F 66 20   bark ruff woof 
00000040  77 6F 6F 66 20 62 61 72 6B 20 72 75 66 66 20 62  woof bark ruff b
00000050  61 72 6B 20 72 75 66 66 20 77 6F 6F 66 20 77 6F  ark ruff woof wo
```



```
00004F00  7E 3E FE 01 49 90 40 FF 07 7C 6D C3 00 00 00 00  ~>þ.I.@ÿ.|mÃ....
00004F10  49 45 4E 44 AE 42 60 82 45 78 74 72 61 20 74 65  IEND®B`,Extra te
00004F20  78 74 20 61 74 20 74 68 65 20 65 6E 64           xt at the end
```

`strings` can assist with hidden text in file, and `binwalk` can help with revealing hidden files within a file

Practical uses of this?

Demo: My_Wishlist.png

Common appearances in CTFs

Password to StegHide, a ZIP / anything that’s locked

Demo: common_examples.jpg

Spectrograms

One of the many ways to visualise audio

Weird alien noises generally mean spectrogram

Tools: Audacity / Sonic Visualiser

Some online tools exist, though in my experience, it’s not as smooth as the tools

Demo: Spectro.wav

Practice Challenges

This workshop is in no way a comprehensive guide to solve every stego challenge

Sometimes, some problem solving and a lot of googling is required

https://workshop-ctf.umisc.info/ has some challenges for you to solve, some skills have been covered in the workshop, others require some exploration, and looking it up on the web

Feel free to ask questions about any of the challenges!