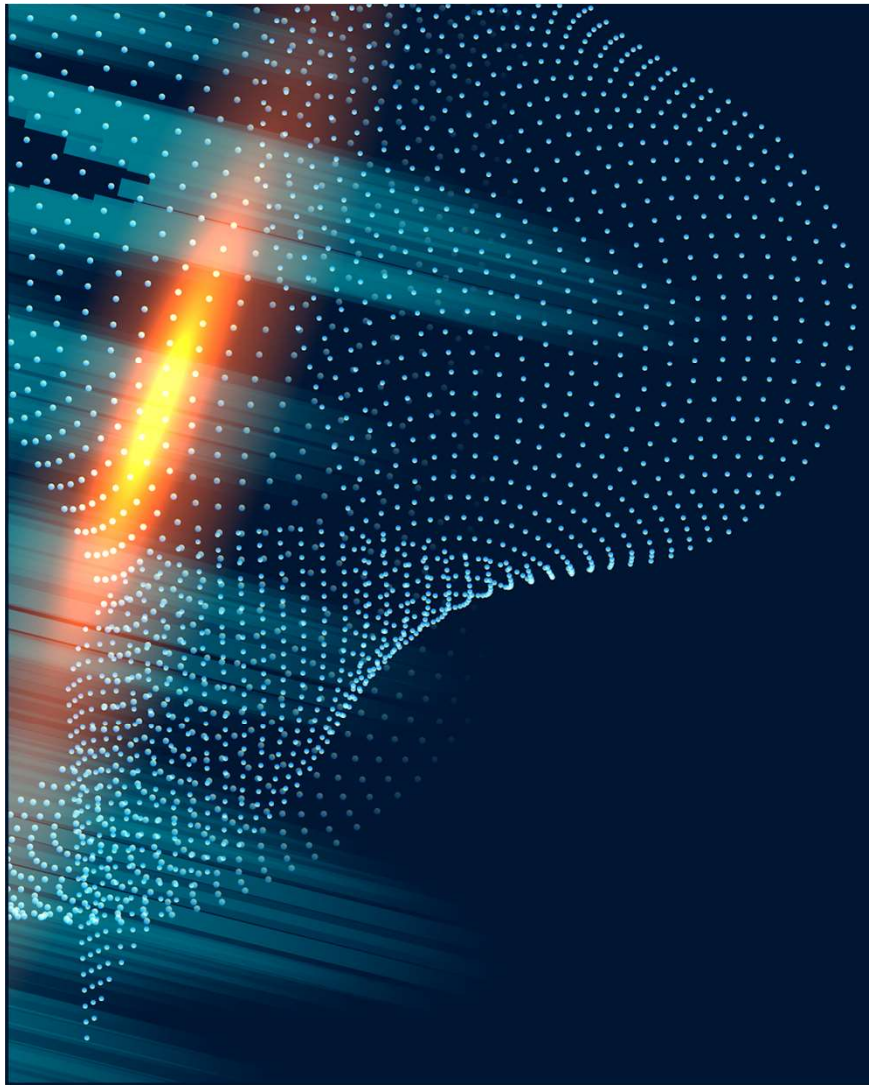
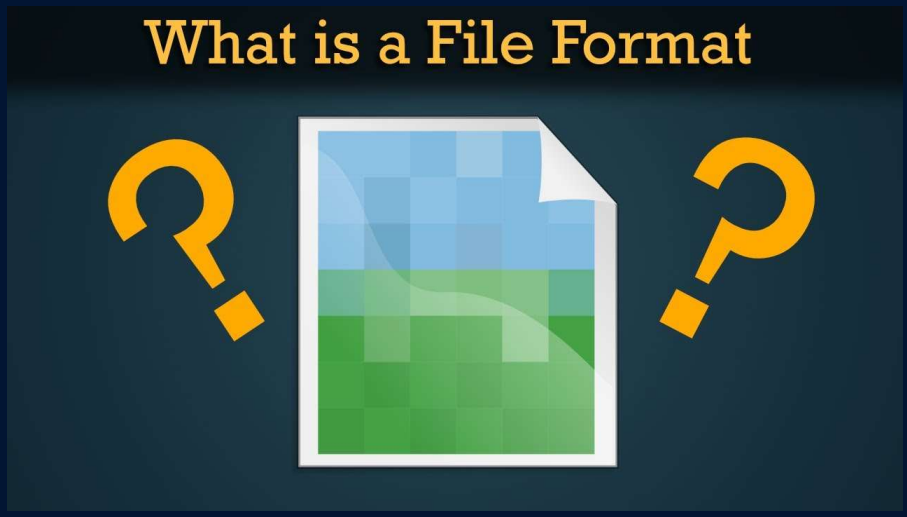

HCMUTE
ISKI
INFORMATION SECURITY CLUB
Digital Forensic
By Raviel


01
File Format
02
Steganography
03
Network Traffic
04
Memory dump

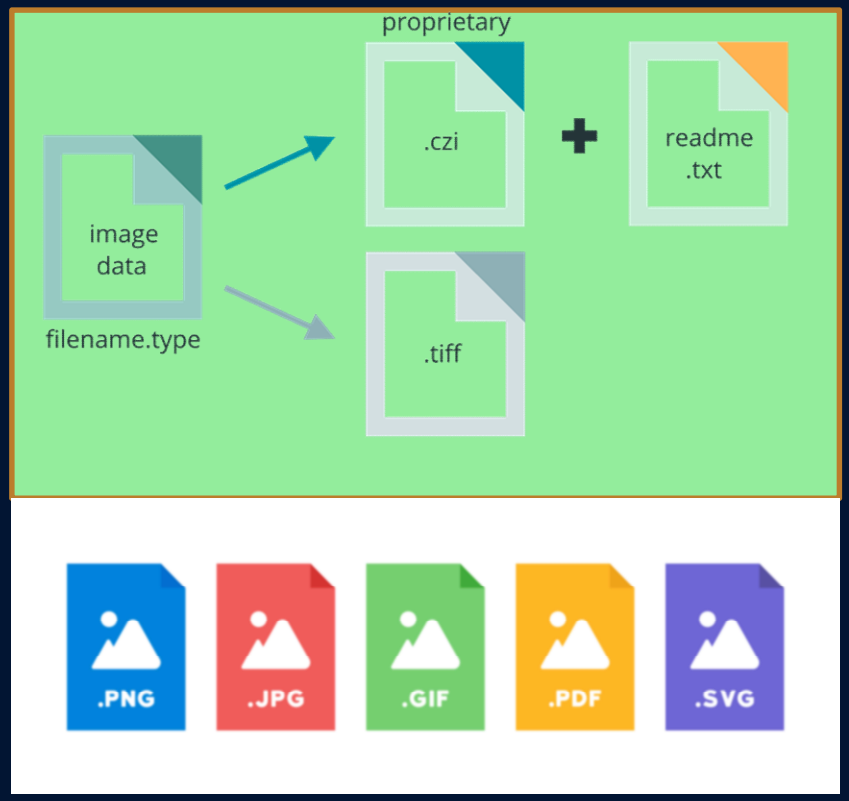


File Format

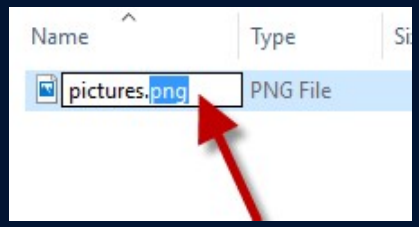


How does a computer know what is the type of the file?

File extension?



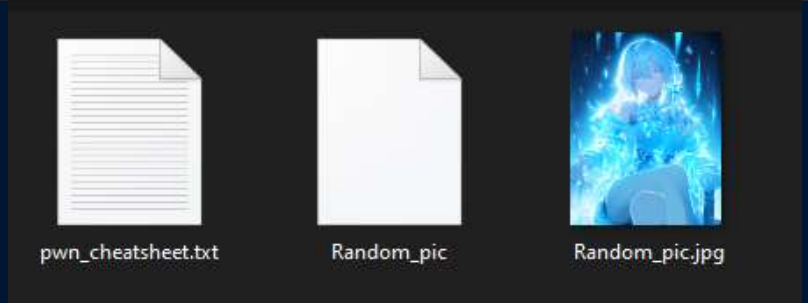
File Extensions are not the sole way to identify the type of a file, files have certain leading bytes called **file signatures** which allow programs to parse the data in a consistent manner. Files can also contain additional **"hidden"** data called **metadata** which can be useful in finding out information about the context of a file's data.



File signatures which allow programs to parse the data in a consistent manner.
Files can also contain additional **"hidden"** data called **metadata**

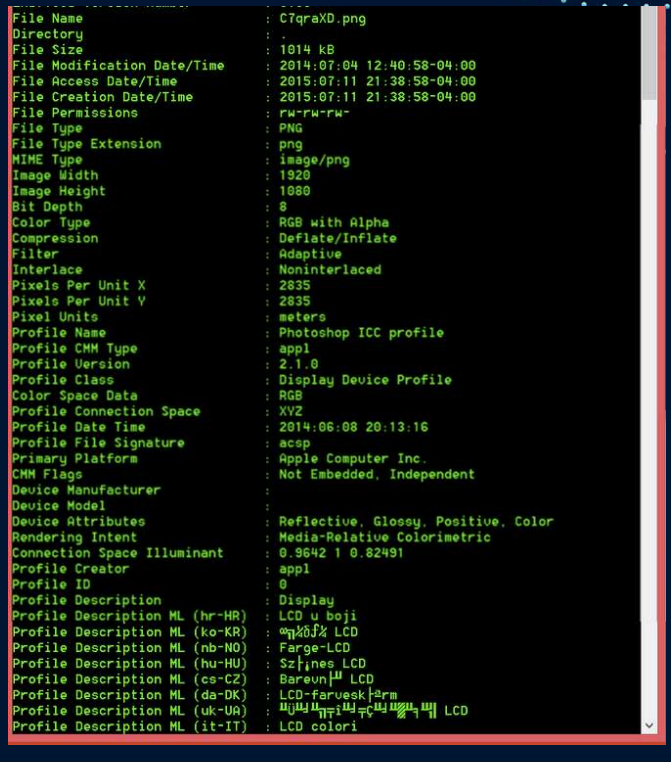
“file” command in Linux

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ file Random_pic.jpg
Random_pic.jpg: JPEG image data, JFIF standard 1.01, aspect ratio, density 72×7
3×2048, components 3

┌──(raviel㉿kali)-[~/Desktop]
└─$ file Random_pic
Random_pic: JPEG image data, JFIF standard 1.01, aspect ratio, density 72×72, s
48, components 3

┌──(raviel㉿kali)-[~/Desktop]
└─$ file pwn_cheatsheet.txt
pwn_cheatsheet.txt: Python script, ASCII text executable

┌──(raviel㉿kali)-[~/Desktop]
└─$ file -h
Usage: file [-bcCdEhikLlNnprsSvzZ0] [--apple] [--extension] [--mime-encoding]
            [--mime-type] [-e <testname>] [-F <separator>]  [-f <namefile>]
            [-m <magicfiles>] [-P <parameter=value>] [--exclude-quiet]
            <file> ...
       file -C [-m <magicfiles>]
       file [--help]
```

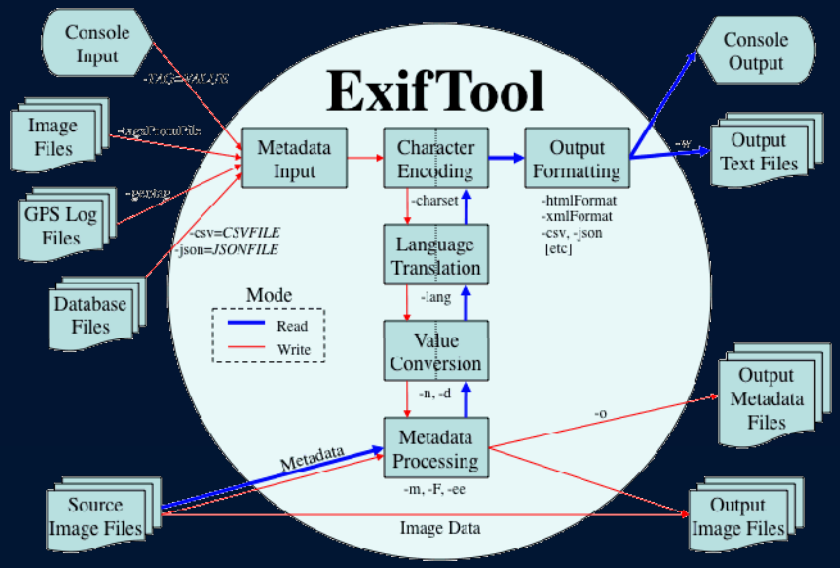


Metadata

Metadata is data about data. Different types of files have different metadata. The metadata on a photo could include dates, camera information, GPS location, comments, etc.



How exiftool work?



[Click here]

Exiftool –a <add/file>

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ exiftool -a 115194173_p0_master1200.jpg
ExifTool Version Number         : 12.76
File Name                       : 115194173_p0_master1200.jpg
Directory                       : .
File Size                       : 1334 kB
File Modification Date/Time     : 2024:06:04 04:28:27-04:00
File Access Date/Time           : 2024:06:04 04:28:39-04:00
File Inode Change Date/Time     : 2024:06:04 04:28:38-04:00
File Permissions                : -rw-------
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : None
X Resolution                    : 1
Y Resolution                    : 1
Profile CMM Type                : Linotronic
Profile Version                 : 2.1.0
Profile Class                   : Display Device Profile
Color Space Data                : RGB
Profile Connection Space        : XYZ
Profile Date Time               : 1998:02:09 06:49:00
Profile File Signature          : acsp
Primary Platform                : Microsoft Corporation
CMM Flags                       : Not Embedded, Independent
Device Manufacturer             : Hewlett-Packard
Device Model                    : sRGB
Device Attributes               : Reflective, Glossy, Positive, Color
Rendering Intent                : Perceptual
Connection Space Illuminant     : 0.9642 1 0.82491
Profile Creator                 : Hewlett-Packard
Profile ID                      : 0
Profile Copyright               : Copyright (c) 1998 Hewlett-Packard Company
Profile Description             : sRGB IEC61966-2.1
Media White Point               : 0.95045 1 1.08905
Media Black Point               : 0 0 0
```

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ exiftool -a pwn_cheatsheet.txt
ExifTool Version Number         : 12.76
File Name                       : pwn_cheatsheet.txt
Directory                       : .
File Size                       : 177 bytes
File Modification Date/Time     : 2024:01:16 09:29:56-05:00
File Access Date/Time           : 2024:06:04 04:44:11-04:00
File Inode Change Date/Time     : 2024:06:04 04:44:08-04:00
File Permissions                : -rwxrw-rw-
File Type                       : TXT
File Type Extension             : txt
MIME Type                       : text/plain
MIME Encoding                   : us-ascii
Newlines                        : Unix LF
Line Count                      : 14
Word Count                      : 19
```

File signatures

(magic numbers or Magic Bytes)

file signatures are datas used to identify or verify the content of a file

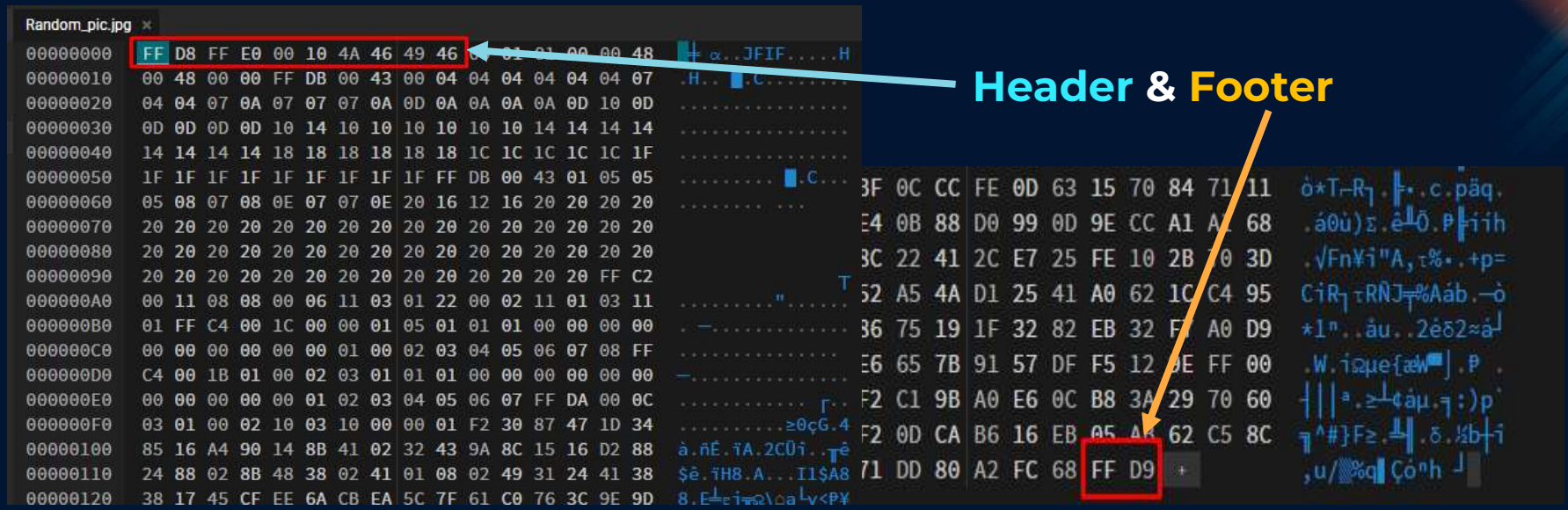


Image Files

File type	Typical extension	Hex digits xx = variable	Ascii digits . = not an ascii char
Bitmap format	.bmp	42 4d	BM
FITS format	.fits	53 49 4d 50 4c 45	SIMPLE
GIF format	.gif	47 49 46 38	GIF8
Graphics Kernel System	.gks	47 4b 53 4d	GKSM
IRIS rgb format	.rgb	01 da	..
ITC (CMU WM) format	.itc	f1 00 40 bb	....
JPEG File Interchange Format	.jpg	ff d8 ff e0	....
NIFF (Navy TIFF)	.nif	49 49 4e 31	IIN1
PM format	.pm	56 49 45 57	VIEW
PNG format	.png	89 50 4e 47	.PNG
Postscript format	.[e]ps	25 21	%!
Sun Rasterfile	.ras	59 a6 6a 95	Y.j.
Targa format	.tga	xx xx xx	...
TIFF format (Motorola - big endian)	.tif	4d 4d 00 2a	MM.*
TIFF format (Intel - little endian)	.tif	49 49 2a 00	II*.
X11 Bitmap format	.xbm	xx xx	
XCF Gimp file structure	.xcf	67 69 6d 70 20 78 63 66 20 76	gimp xcf
Xfig format	.fig	23 46 49 47	#FIG
XPM format	.xpm	2f 2a 20 58 50 4d 20 2a 2f	/* XPM */

Compressed files

File type	Typical extension	Hex digits xx = variable	Ascii digits . = not an ascii char
Bzip	.bz	42 5a	BZ
Compress	.Z	1f 9d	..
gzip format	.gz	1f 8b	..
pkzip format	.zip	50 4b 03 04	PK..

Archive files

File type	Typical extension	Hex digits xx = variable	Ascii digits . = not an ascii char
TAR (pre-POSIX)	.tar	xx xx	(a filename)
TAR (POSIX)	.tar	75 73 74 61 72	ustar (offset by 257 bytes)

Excecutable files

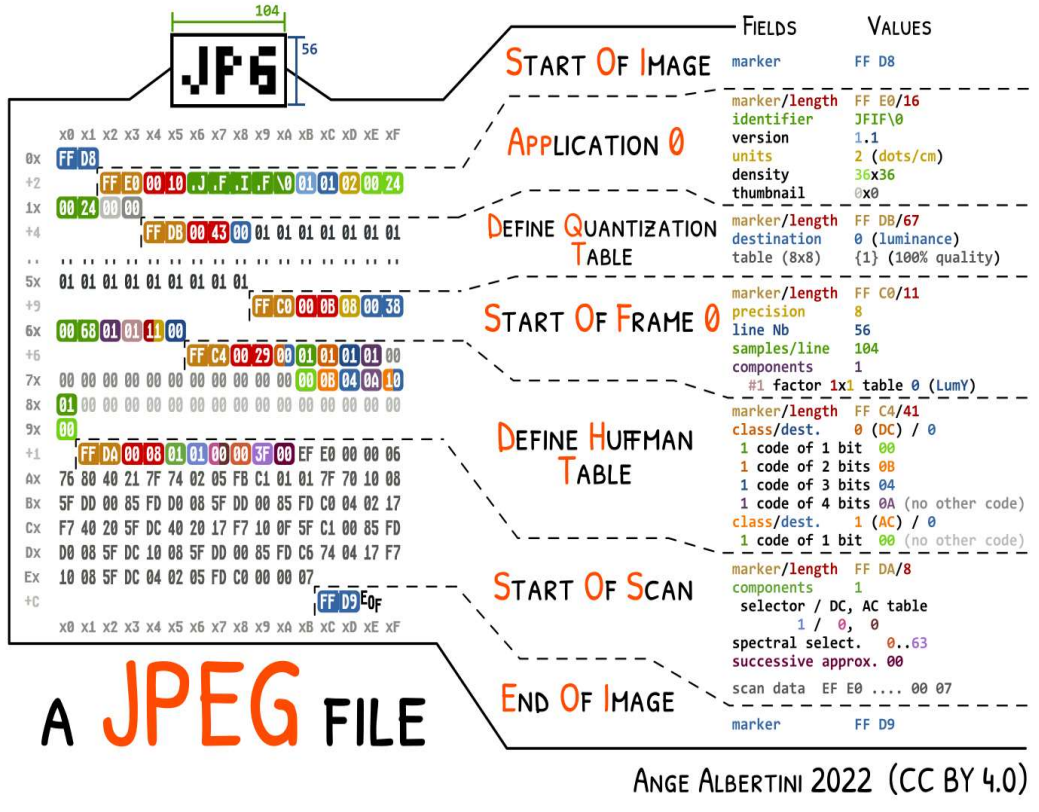
File type	Typical extension	Hex digits xx = variable	Ascii digits . = not an ascii char
MS-DOS, OS/2 or MS Windows		4d 5a	MZ
Unix elf		7f 45 4c 46	.ELF

Corrupted File?

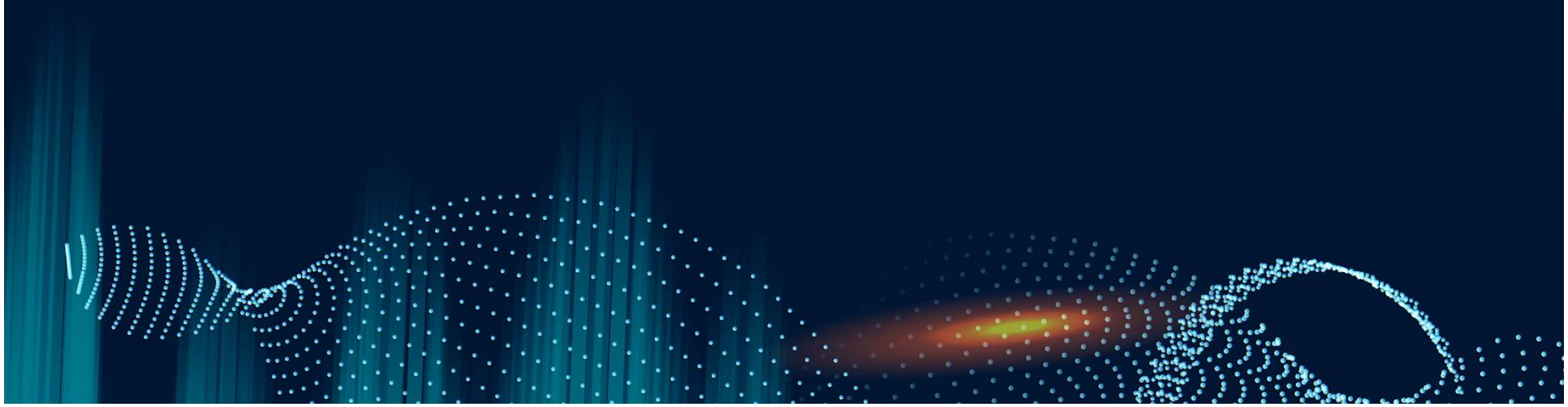
corrupted.jpg

```
00000000  DC 00 FE E1 20 10 4A 47 50 16 00 01 01 00 00 48
00000010  00 48 00 00 FF DB 00 43 00 04 04 04 04 04 04 07
00000020  04 04 07 0A 07 07 07 0A 0D 0A 0A 0A 0A 0D 10 0D
00000030  0D 0D 0D 0D 10 14 10 10 10 10 10 10 14 14 14 14
00000040  14 14 14 14 18 18 18 18 18 18 1C 1C 1C 1C 1C 1F
00000050  1F 1F 1F 1F 1F 1F 1F 1F FF DB 00 43 01 05 05 05
00000060  05 08 07 08 0E 07 07 0E 20 16 12 16 20 20 20 20
00000070  20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
00000080  20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
```

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ file corrupted.jpg
corrupted.jpg: data
```



steganography



steganography

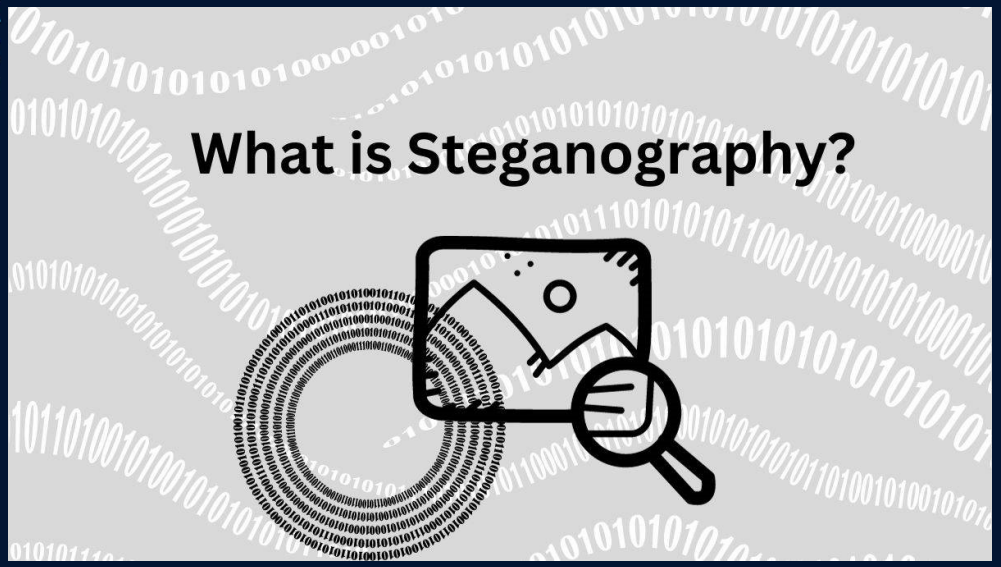
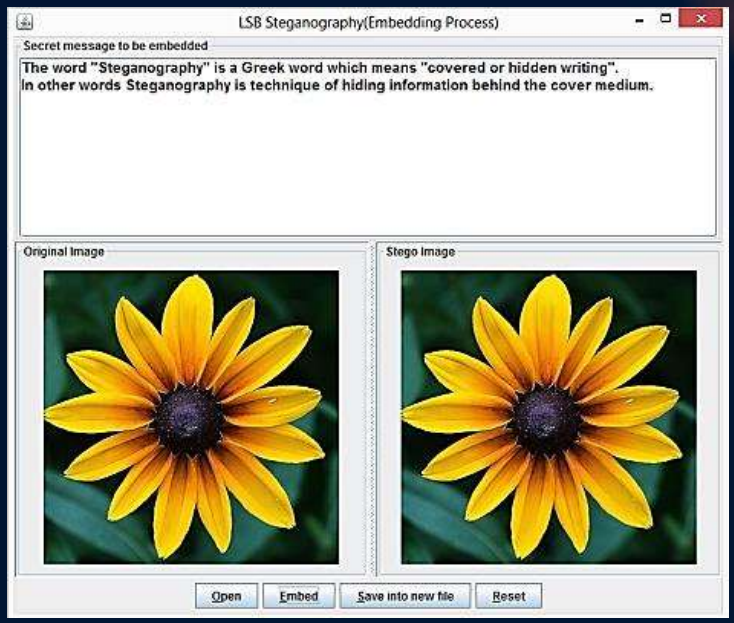


Image Steganography

Based on what we have learnt so far, can you give me an idea how to hide a “data” inside an image



Strings Command

We can use Strings to crawl all the printable ascii-letters in a file's data

Manual

Strings <add/name>

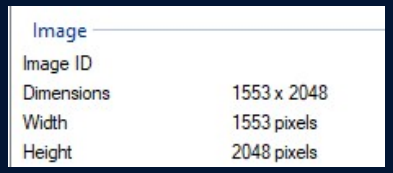
```
┌──(raviel㉿kali)-[~/Desktop]
└─$ strings chall
/lib64/ld-linux-x86-64.so.2
__libc_start_main
__gmon_start__
_ITM_deregisterTMCloneTable
_ITM_registerTMCloneTable
__cxa_finalize
_edata
__bss_start
_end
printf
__isoc99_scanf
libc.so.6
GLIBC_2.2.5
GLIBC_2.7
GLIBC_2.34
libc++.so.1
libc++abi.so.1
libunwind.so.1
libstdc++.so.6
libm.so.6
libgcc_s.so.1
```

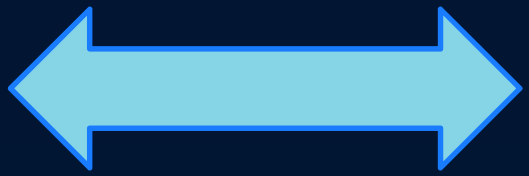
```
┌──(raviel㉿kali)-[~/Desktop]
└─$ echo "this is some printable ascii-letters" > flag

┌──(raviel㉿kali)-[~/Desktop]
└─$ strings flag
this is some printable ascii-letters

┌──(raviel㉿kali)-[~/Desktop]
└─$ strings flag > output.txt

┌──(raviel㉿kali)-[~/Desktop]
└─$ cat output.txt
this is some printable ascii-letters
```

You can also use ">" <name>.<ext> to print the output to a file

Dimensional Hidden text

How can you determine the image's size? Open properties?





Raviel{n0w_u_Kn0W_d0m41n_3xp4ns10n}

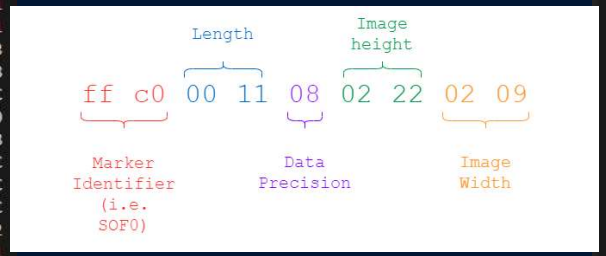
JPG image height

spot the difference yet?

Raviel{n0w_u_Kn0W_d0m41n_3xp4ns10n}

tesst.jpg × Random_pic.jpg ×

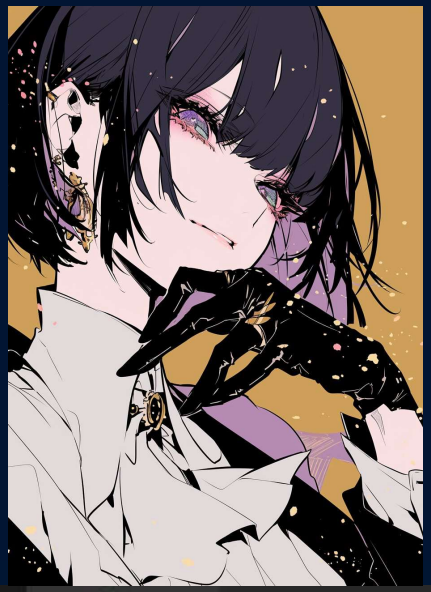
```
FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 00 60
00 60 00 00 FF E1 00 22 45 78 69 66 00 00 4D 4D
00 2A 00 00 00 08 00 01 01 12 00 03 00 00 00 01
00 01 00 00 00 00 00 00 FF DB 00 43 00 02 01 01
02 01 01 02 02 02 02 02 02 02 02 03 05 03 03 03
03 03 06 04 04 03 05 07 06 07 07 07 06 07 07 08
09 0B 09 08 08 0A 08 07 07 0A 0D 0A 0A 0B 0C 0C
0C 0C 07 09 0E 0F 0D 0C 0E 0B 0C 0C 0C FF DB 00
43 01 02 02 02 03 03 03 06 03 03 06 0C 08 07 08
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C FF C0 00 11 08 07 C0 06 10 03 01 22 00 02
11 01 03 11 01 FF C4 00 1F 00 00 01 05 01 01 01
01 01 01 00 00 00 00 00 00 00 00 01 02 03 04 05
06 07 08 09 0A 0B FF C4 00 B5 10 00 02 01 03 03
```



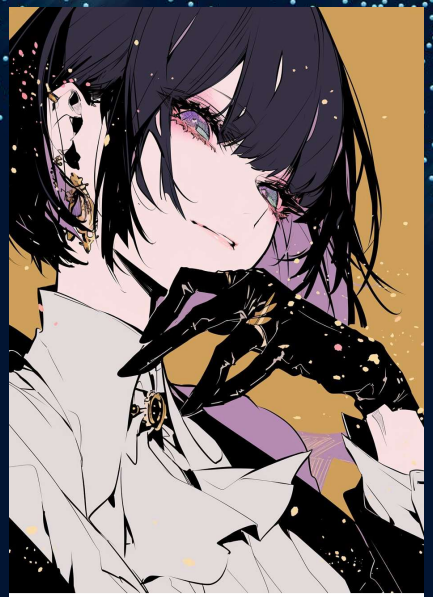
tesst.jpg × Random_pic.jpg ×

```
FF D8 FF E0 00 10 4A 46 49 46 00 01 01 01 00 60
00 60 00 00 FF E1 00 22 45 78 69 66 00 00 4D 4D
00 2A 00 00 00 08 00 01 01 12 00 03 00 00 00 01
00 01 00 00 00 00 00 00 FF DB 00 43 00 02 01 01
02 01 01 02 02 02 02 02 02 02 02 03 05 03 03 03
03 03 06 04 04 03 05 07 06 07 07 07 06 07 07 08
09 0B 09 08 08 0A 08 07 07 0A 0D 0A 0A 0B 0C 0C
0C 0C 07 09 0E 0F 0D 0C 0E 0B 0C 0C 0C FF DB 00
43 01 02 02 02 03 03 03 06 03 03 06 0C 08 07 08
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C 0C
0C 0C FF C0 00 11 08 08 C0 06 10 03 01 22 00 02
11 01 03 11 01 FF C4 00 1F 00 00 01 05 01 01 01
01 01 01 00 00 00 00 00 00 00 00 01 02 03 04 05
```

PNG image height



spot the difference yet?



Raviel{4n0th3r_Fl4g}

changed.png × png_pic.png ×

```
89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52
00 00 03 A0 00 00 05 10 08 06 00 00 00 09 A7 88
2D 00 00 00 01 73 52 47 42 00 AE CE 1C E9 00 00
00 04 67 41 4D 41 00 00 B1 8F 0B FC 61 05 00 00
00 09 70 48 59 73 00 00 0E F2 00 00 0E F2 01 CE
14 7B DE 00 00 00 11 74 45 58 74 54 69 74 6C 65
```

changed.png × png_pic.png ×

```
89 50 4E 47 0D 0A 1A 0A 00 00 00 0D
00 00 03 A0 00 00 05 92 08 06 00 00
2D 00 00 00 01 73 52 47 42 00 AE CE
00 04 67 41 4D 41 00 00 B1 8F 0B FC
00 09 70 48 59 73 00 00 0E F2 00 00
14 7B DE 00 00 00 11 74 45 58 74 54
```

Extract hidden file from image

Manual

steghide extract -sf <add\name> (jpg,bmp only)

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ steghide extract -sf Random_pic.jpg
Enter passphrase:
wrote extracted data to "flag".
```

they do require a password but you can find it by using other medthods or just press Enter

binwalk --dd='.*' <add/name>

```
┌──(raviel㉿kali)-[~/Desktop]
└─$ binwalk --dd='.*' Random_pic.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
```

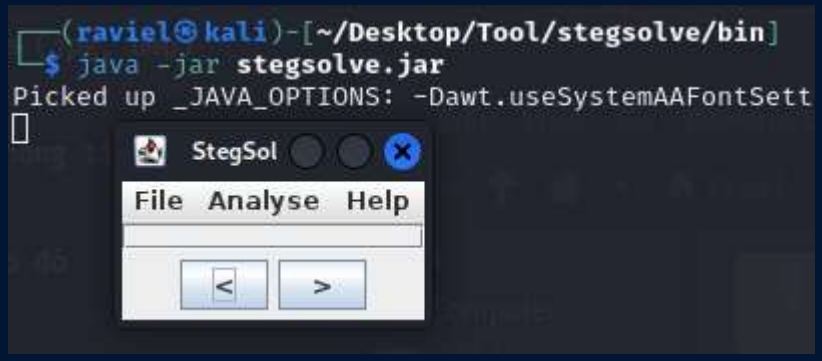
Extract hidden text in plain colour

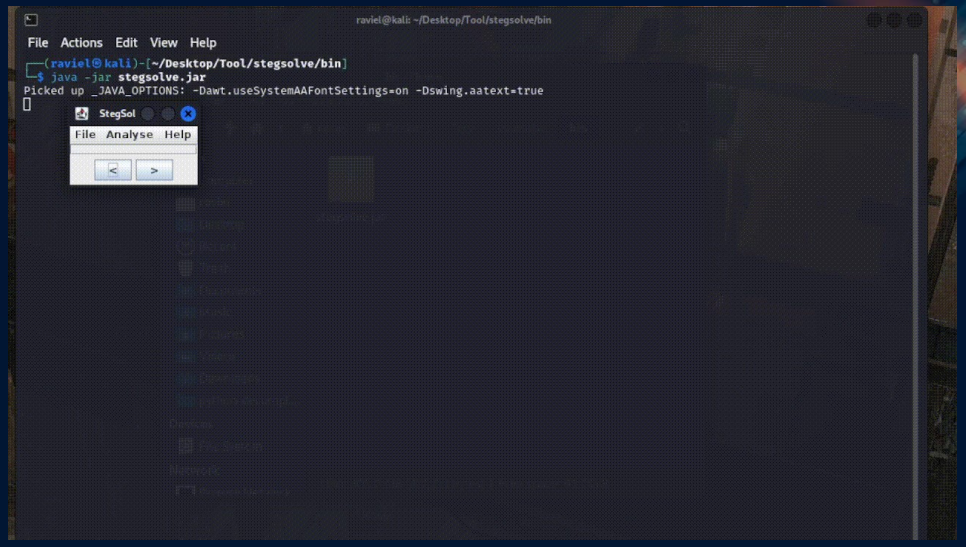
Install stegsolve.jar

wget http://www.caesum.com/handbook/Stegsolve.jar -O stegsolve.jar
chmod +x stegsolve.jar

Run stegsolve.jar

java –jar stegsolve.jar





(Gif Example)

LSB/MSB (sigbits)

Install:

pip install Pillow
git clone https://github.com/Pulho/sigBits

Manual:

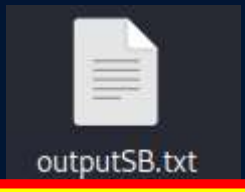
*python sigBits.py --type=LSB + <add/name> (if the picture look completely normal)

*python sigBits.py --type=MSB + <add/name> (if the picture's colour is corrupted)

This is an example of MSB

```
┌──(raviel㉿kali)-[~/Desktop/Tool/sigBits]
└─$ python sigBits.py --type=MSB MSB.png
Done, check the output file!
```

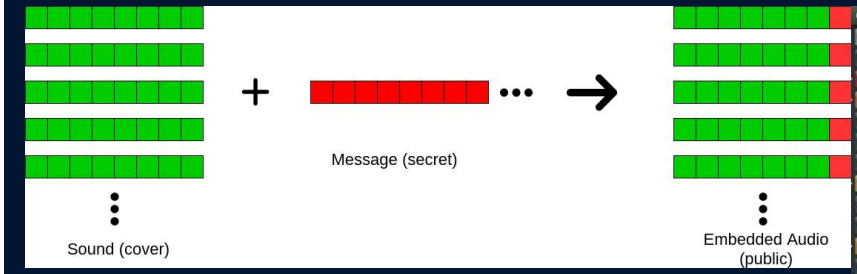
After using the command there will be an output file named outputSB.txt

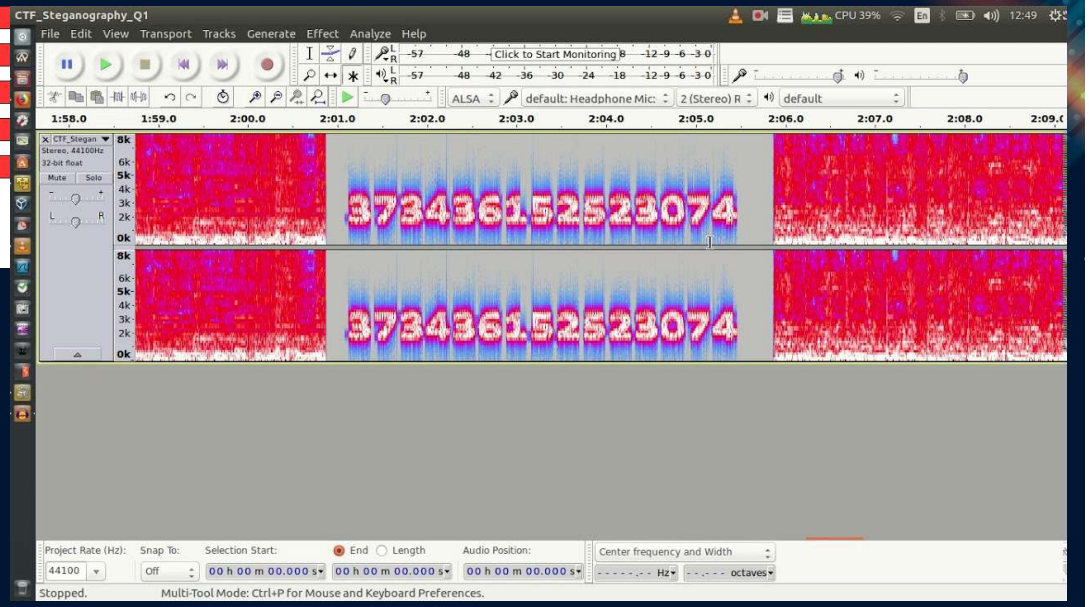


Since the file contain enormous amount of letters its may crash or froze your VM, it should be opened on your real machine!!!

Audio Steganography

Audio Steganography is a technique used to transmit hidden information by modifying an audio signal in an imperceptible manner.



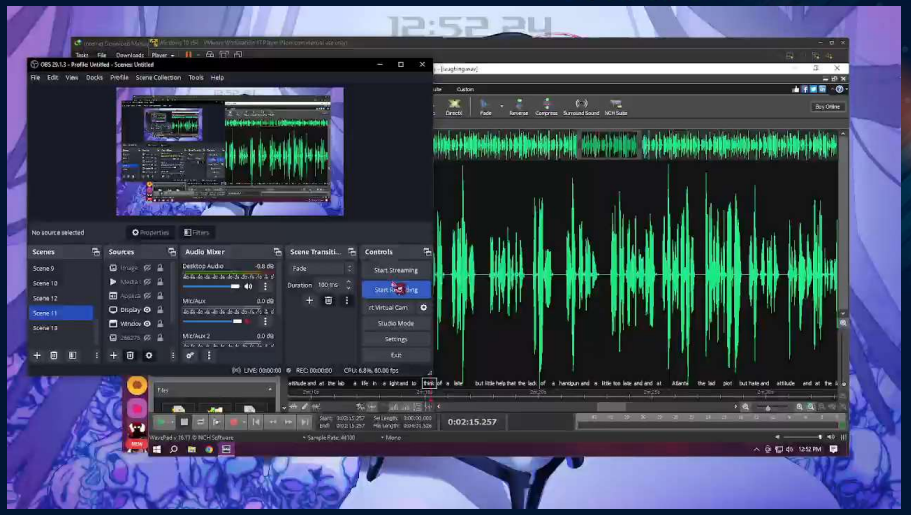


Slow down/speed up/revert Audio

One of the most common task in audio steganography

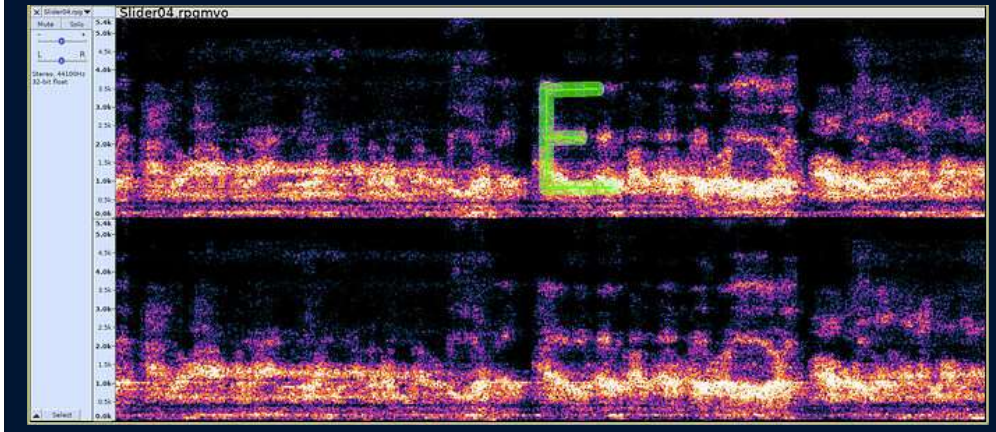
Original Audio

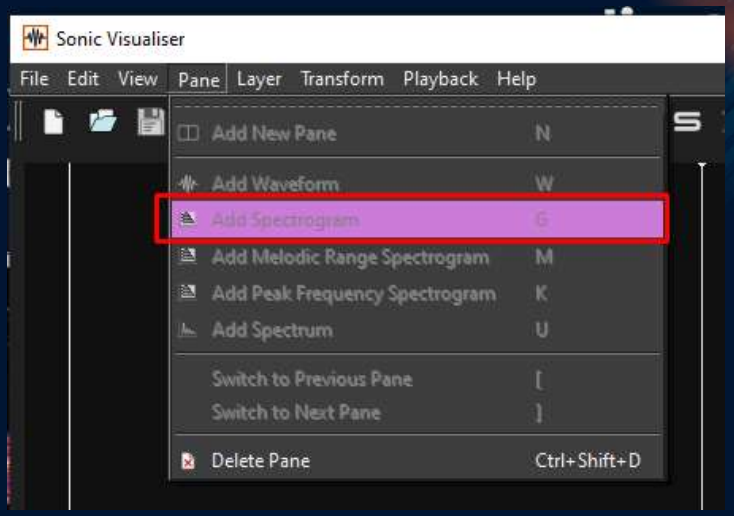
After repairing
The audio



Spectogram

Sometime time they may hid word in spectogram, you can use some application such as audacity, sonic visualizer, then show spectogram to observe





Encoded Audio

Sometime audio steganography can be really guessy

Do you know what type of encoding this is?

HINT: How did pictures from the moon landing get sent back to Earth?

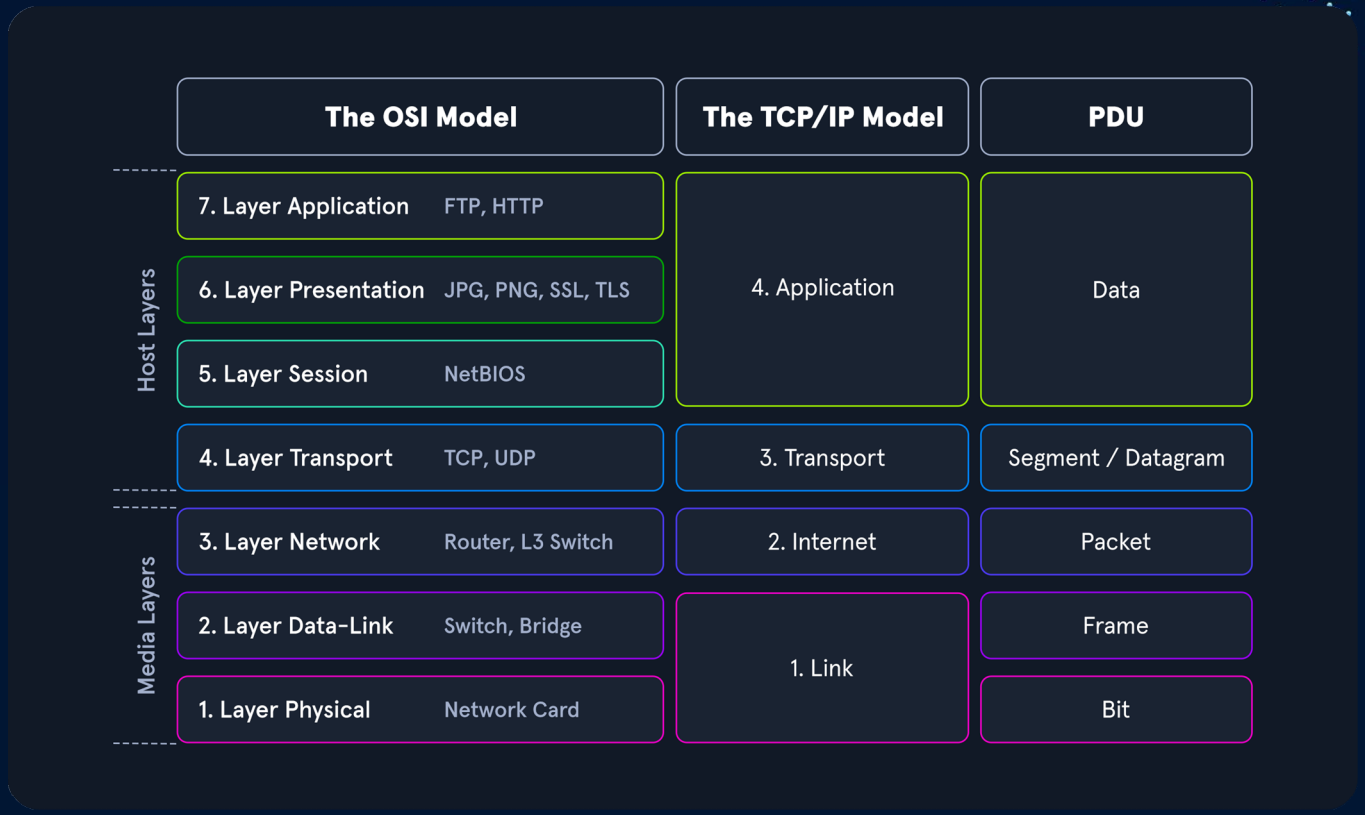
ANS

SOLUTION


Network Traffic

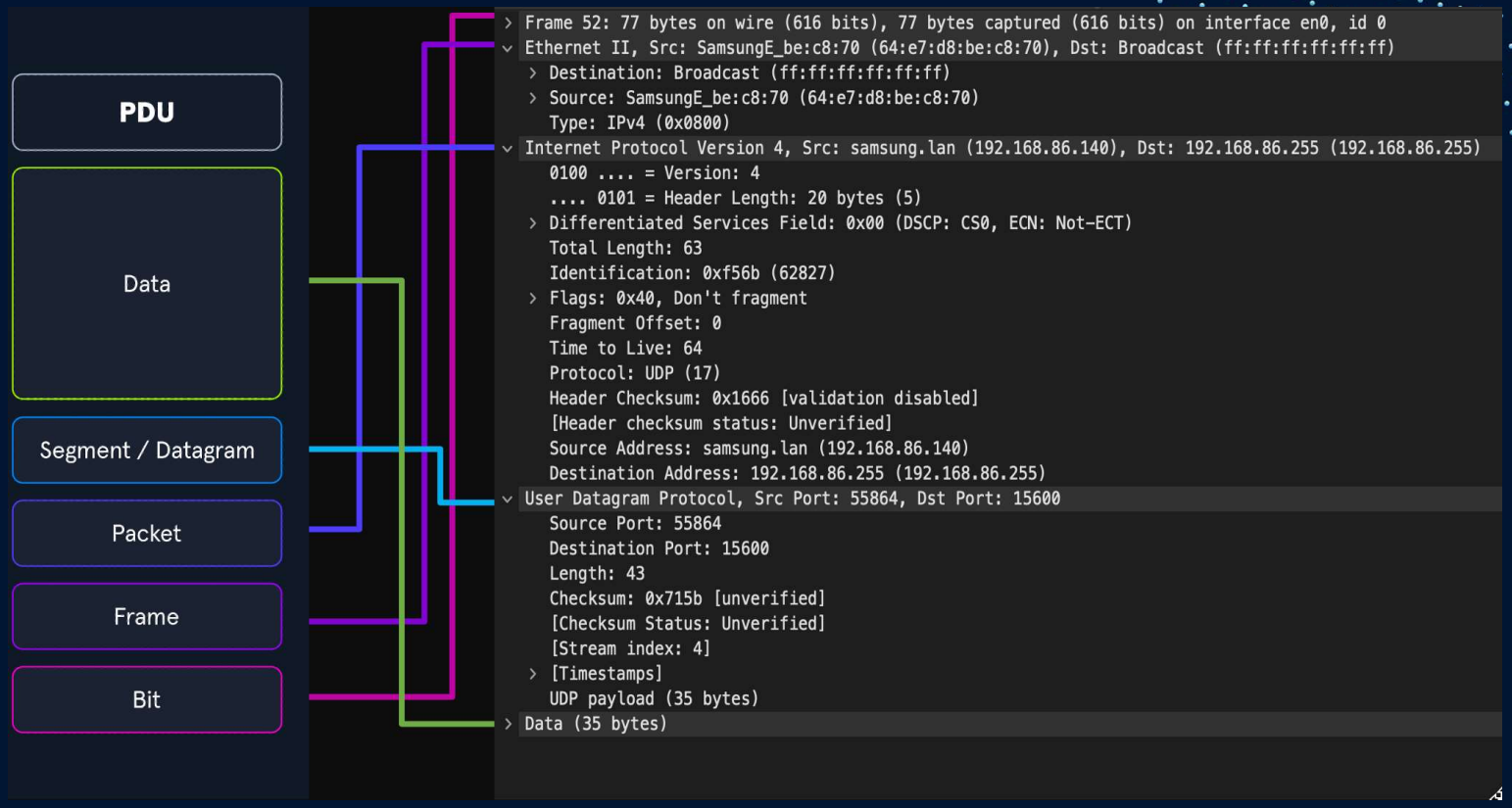
Review

Before going into Network traffic, we need some revision



OSI model?
UDP, TCP?
IP, Port?

PDU Packet Breakdown



IPv4/IPv6/Mac

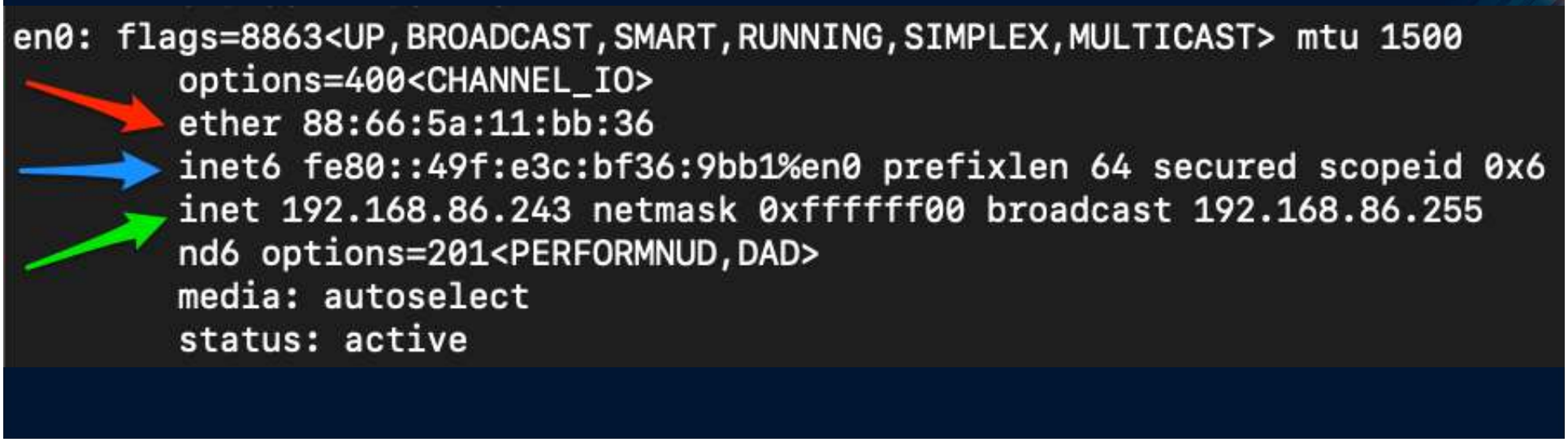
Which is IPv4, IPv6 and Mac address?

IPv4:

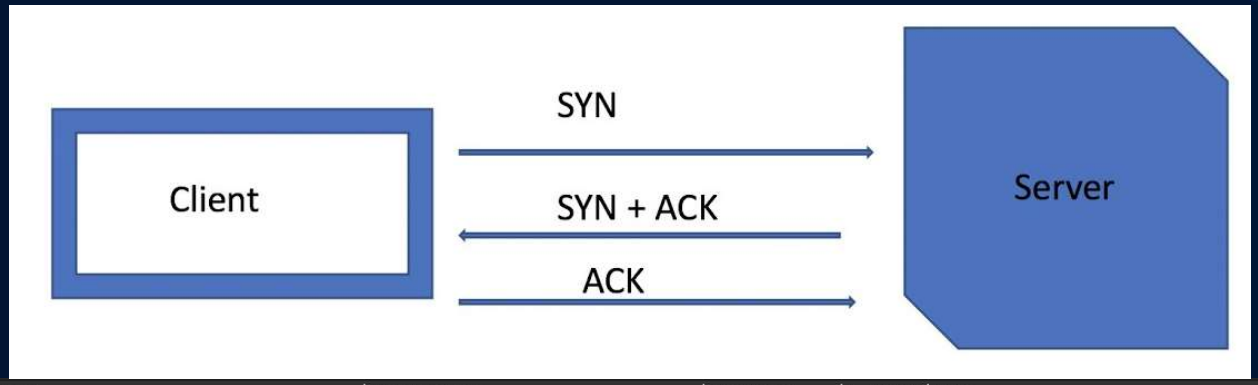
IPv6:

MAC:

ANS

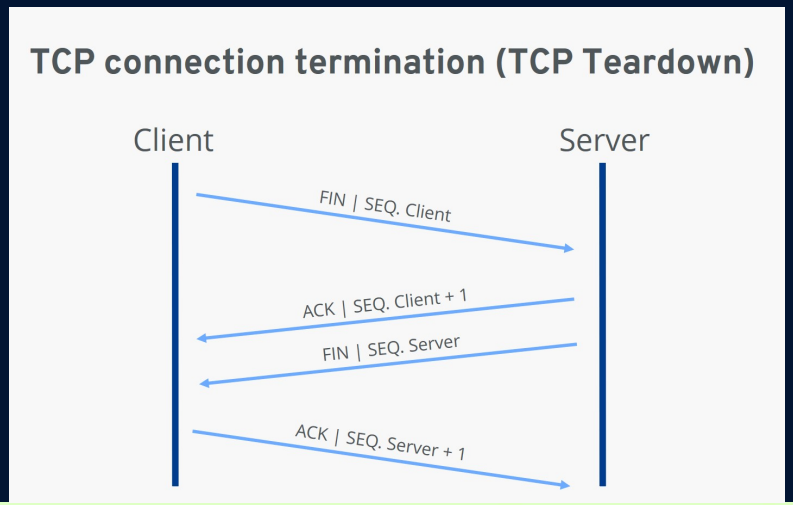


TCP Three-way Handshake



Source	Destination	Protocol	Length	Info
192.168.1.140	174.143.213.184	TCP	74	57678 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=
174.143.213.184	192.168.1.140	TCP	74	80 → 57678 [SYN, ACK] Seq=0 Ack=1 Win=5792
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=
192.168.1.140	174.143.213.184	HTTP	200	GET /images/layout/logo.png HTTP/1.0
174.143.213.184	192.168.1.140	TCP	66	80 → 57678 [ACK] Seq=1 Ack=135 Win=6912 Le
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=1 Ack=135 Win=6912 Le
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=1449 Win=8832
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=1449 Ack=135 Win=6912
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=2897 Win=1164
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=2897 Ack=135 Win=6912
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=4345 Win=1459
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=4345 Ack=135 Win=6912
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=5793 Win=1753
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=5793 Ack=135 Win=6912

TCP Session Teardown



174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=15929 Ack=135
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=17377
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=17377 Ack=135
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=18825
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=18825 Ack=135
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=20273
174.143.213.184	192.168.1.140	TCP	1514	80 → 57678 [ACK] Seq=20273 Ack=135
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=21721
174.143.213.184	192.168.1.140	HTTP	391	HTTP/1.1 200 OK (PNG)
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=135 Ack=22046
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [FIN, ACK] Seq=135 Ack=
174.143.213.184	192.168.1.140	TCP	66	80 → 57678 [FIN, ACK] Seq=22046 Ac
192.168.1.140	174.143.213.184	TCP	66	57678 → 80 [ACK] Seq=136 Ack=22047

Wireshark

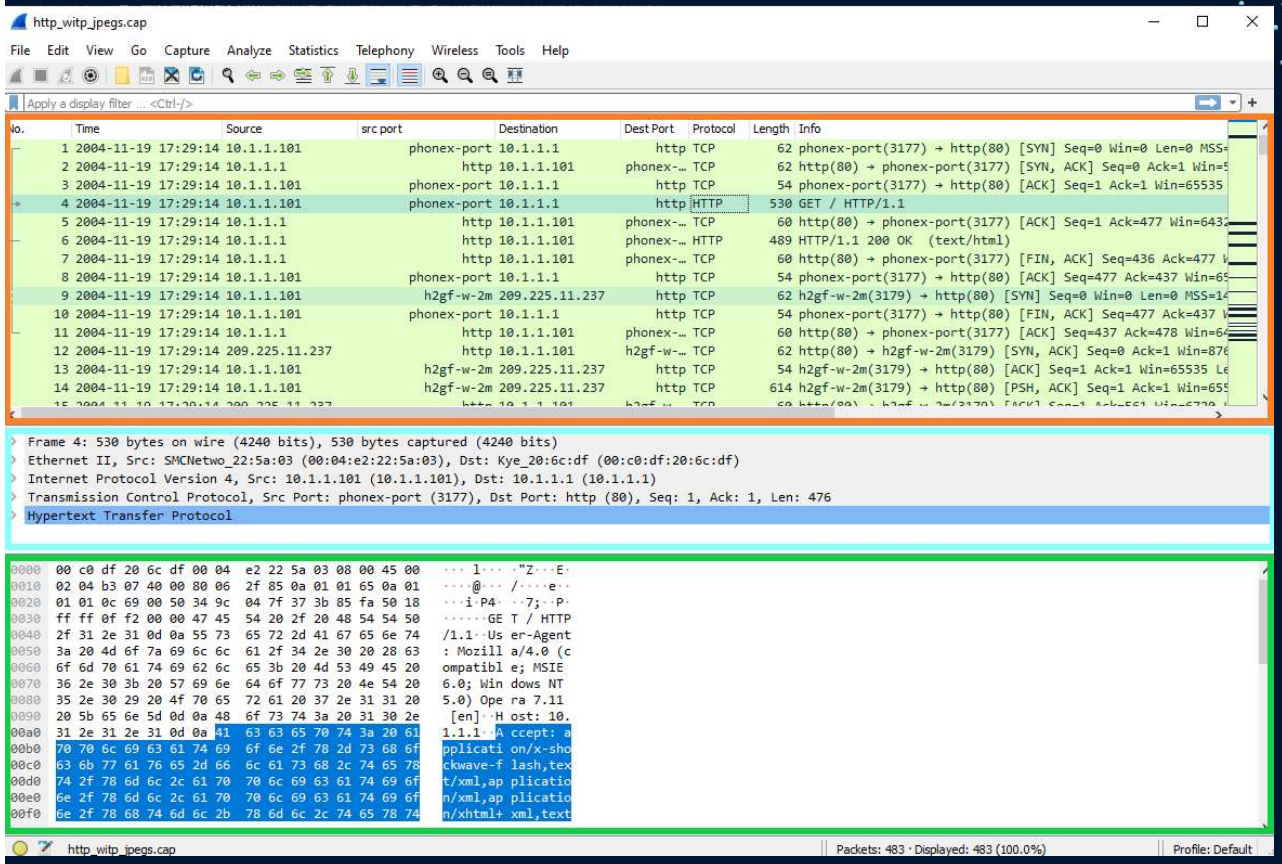
Install (linux)

$ sudo apt install wireshark

Install (windows)

https://www.wireshark.org/download.html

Wireshark GUI

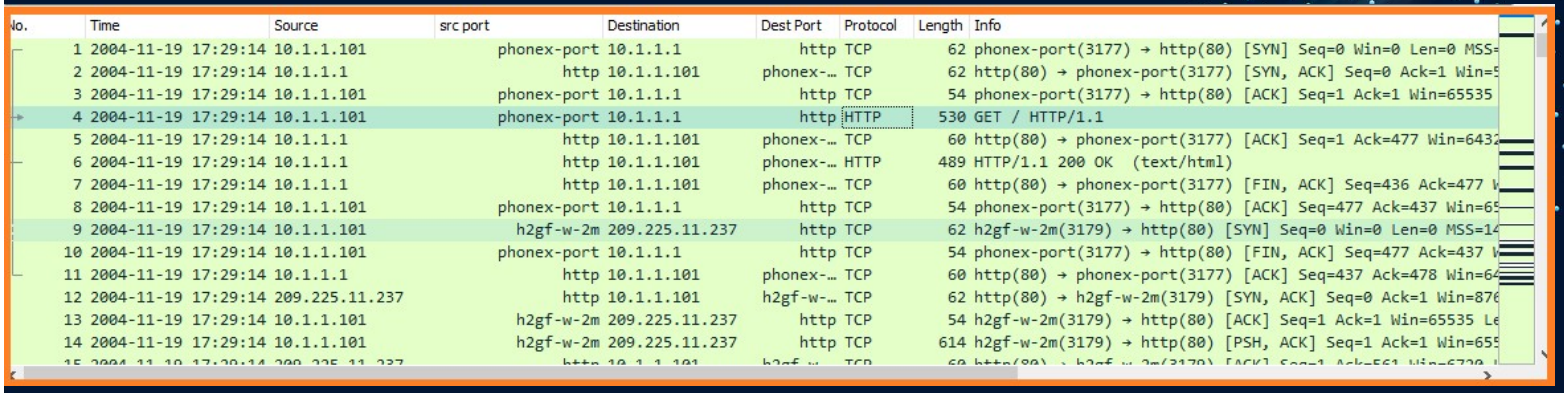


Packet List

Packet Details

Packet Bytes

Packet List

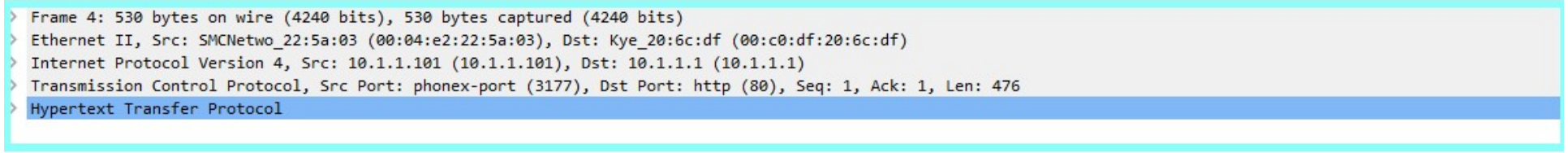


No.	Time	Source	src port	Destination	Dest Port	Protocol	Length	Info
1	2004-11-19 17:29:14	10.1.1.101	phonex-port	10.1.1.1	http	TCP	62	phonex-port(3177) → http(80) [SYN] Seq=0 Win=0 Len=0 MSS=
2	2004-11-19 17:29:14	10.1.1.1	http	10.1.1.101	phonex-…	TCP	62	http(80) → phonex-port(3177) [SYN, ACK] Seq=0 Ack=1 Win=5
3	2004-11-19 17:29:14	10.1.1.101	phonex-port	10.1.1.1	http	TCP	54	phonex-port(3177) → http(80) [ACK] Seq=1 Ack=1 Win=65535
4	2004-11-19 17:29:14	10.1.1.101	phonex-port	10.1.1.1	http	HTTP	530	GET / HTTP/1.1
5	2004-11-19 17:29:14	10.1.1.1	http	10.1.1.101	phonex-…	TCP	60	http(80) → phonex-port(3177) [ACK] Seq=1 Ack=477 Win=643
6	2004-11-19 17:29:14	10.1.1.1	http	10.1.1.101	phonex-…	HTTP	489	HTTP/1.1 200 OK (text/html)
7	2004-11-19 17:29:14	10.1.1.1	http	10.1.1.101	phonex-…	TCP	60	http(80) → phonex-port(3177) [FIN, ACK] Seq=436 Ack=477 W
8	2004-11-19 17:29:14	10.1.1.101	phonex-port	10.1.1.1	http	TCP	54	phonex-port(3177) → http(80) [ACK] Seq=477 Ack=437 Win=65
9	2004-11-19 17:29:14	10.1.1.101	h2gf-w-2m	209.225.11.237	http	TCP	62	h2gf-w-2m(3179) → http(80) [SYN] Seq=0 Win=0 Len=0 MSS=14
10	2004-11-19 17:29:14	10.1.1.101	phonex-port	10.1.1.1	http	TCP	54	phonex-port(3177) → http(80) [FIN, ACK] Seq=477 Ack=437 W
11	2004-11-19 17:29:14	10.1.1.1	http	10.1.1.101	phonex-…	TCP	60	http(80) → phonex-port(3177) [ACK] Seq=437 Ack=478 Win=64
12	2004-11-19 17:29:14	209.225.11.237	http	10.1.1.101	h2gf-w-…	TCP	62	http(80) → h2gf-w-2m(3179) [SYN, ACK] Seq=0 Ack=1 Win=876
13	2004-11-19 17:29:14	10.1.1.101	h2gf-w-2m	209.225.11.237	http	TCP	54	h2gf-w-2m(3179) → http(80) [ACK] Seq=1 Ack=1 Win=65535 Le
14	2004-11-19 17:29:14	10.1.1.101	h2gf-w-2m	209.225.11.237	http	TCP	614	h2gf-w-2m(3179) → http(80) [PSH, ACK] Seq=1 Ack=1 Win=655

In this window, we see a summary line of each packet that includes the fields listed below by default. We can add or remove columns to change what information is presented.

- Number- Order the packet arrived in Wireshark
- Time- Unix time format
- Source- Source IP
- Destination- Destination IP
- Protocol- The protocol used (TCP, UDP, DNS, ETC.)
- Information- Information about the packet. This field can vary based on the type of protocol used within. It will show, for example, what type of query It is for a DNS packet.

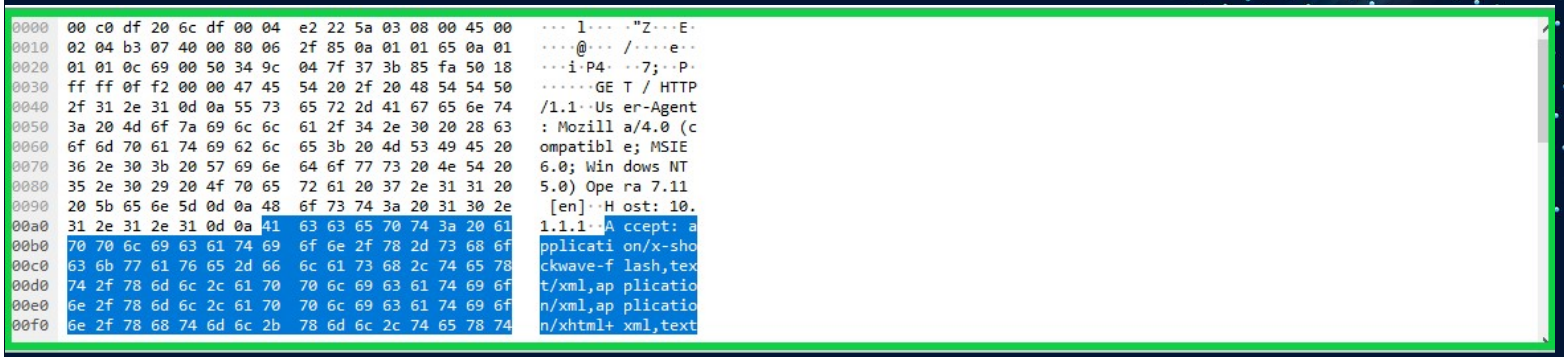
Packet Details



The Packet Details window allows us to drill down into the packet to inspect the protocols with greater detail. It will break it down into chunks that we would expect following the typical OSI Model reference. The packet is dissected into different encapsulation layers for inspection.

Keep in mind, Wireshark will show this encapsulation in reverse order with lower layer encapsulation at the top of the window and higher levels at the bottom.

Packet Bytes



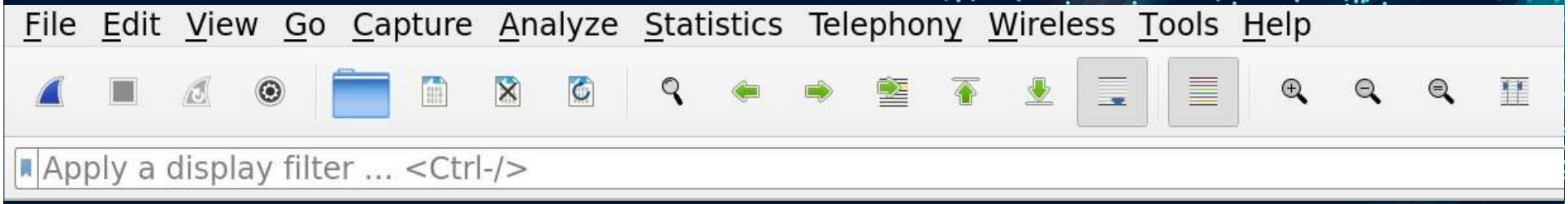
```
0000  00 c0 df 20 6c df 00 04  e2 22 5a 03 08 00 45 00   ··· l··· ·"Z···E·
0010  02 04 b3 07 40 00 80 06  2f 85 0a 01 01 65 0a 01   ····@··· /····e··
0020  01 01 0c 69 00 50 34 9c  04 7f 37 3b 85 fa 50 18   ···i·P4· ··7;··P·
0030  ff ff 0f f2 00 00 47 45  54 20 2f 20 48 54 54 50   ·····GE T / HTTP
0040  2f 31 2e 31 0d 0a 55 73  65 72 2d 41 67 65 6e 74   /1.1··Us er-Agent
0050  3a 20 4d 6f 7a 69 6c 6c  61 2f 34 2e 30 20 28 63   : Mozill a/4.0 (c
0060  6f 6d 70 61 74 69 62 6c  65 3b 20 4d 53 49 45 20   ompatibl e; MSIE
0070  36 2e 30 3b 20 57 69 6e  64 6f 77 73 20 4e 54 20   6.0; Win dows NT
0080  35 2e 30 29 20 4f 70 65  72 61 20 37 2e 31 31 20   5.0) Ope ra 7.11
0090  20 5b 65 6e 5d 0d 0a 48  6f 73 74 3a 20 31 30 2e    [en]··H ost: 10.
00a0  31 2e 31 2e 31 0d 0a 41  63 63 65 70 74 3a 20 61   1.1.1··A ccept: a
00b0  70 70 6c 69 63 61 74 69  6f 6e 2f 78 2d 73 68 6f   pplicati on/x-sho
00c0  63 6b 77 61 76 65 2d 66  6c 61 73 68 2c 74 65 78   ckwave-f lash,tex
00d0  74 2f 78 6d 6c 2c 61 70  70 6c 69 63 61 74 69 6f   t/xml,ap plicatio
00e0  6e 2f 78 6d 6c 2c 61 70  70 6c 69 63 61 74 69 6f   n/xml,ap plicatio
00f0  6e 2f 78 68 74 6d 6c 2b  78 6d 6c 2c 74 65 78 74   n/xhtml+ xml,text
```

The Packet Bytes window allows us to look at the packet contents in ASCII or hex output. As we select a field from the windows above, it will be highlighted in the Packet Bytes window and show us where that bit or byte falls within the overall packet.

This is a great way to validate that what we see in the Details pane is accurate and the interpretation Wireshark made matches the packet output.

Each line in the output contains the data offset, sixteen hexadecimal bytes, and sixteen ASCII bytes. Non-printable bytes are replaced with a period in the ASCII format.

Wireshark Menu

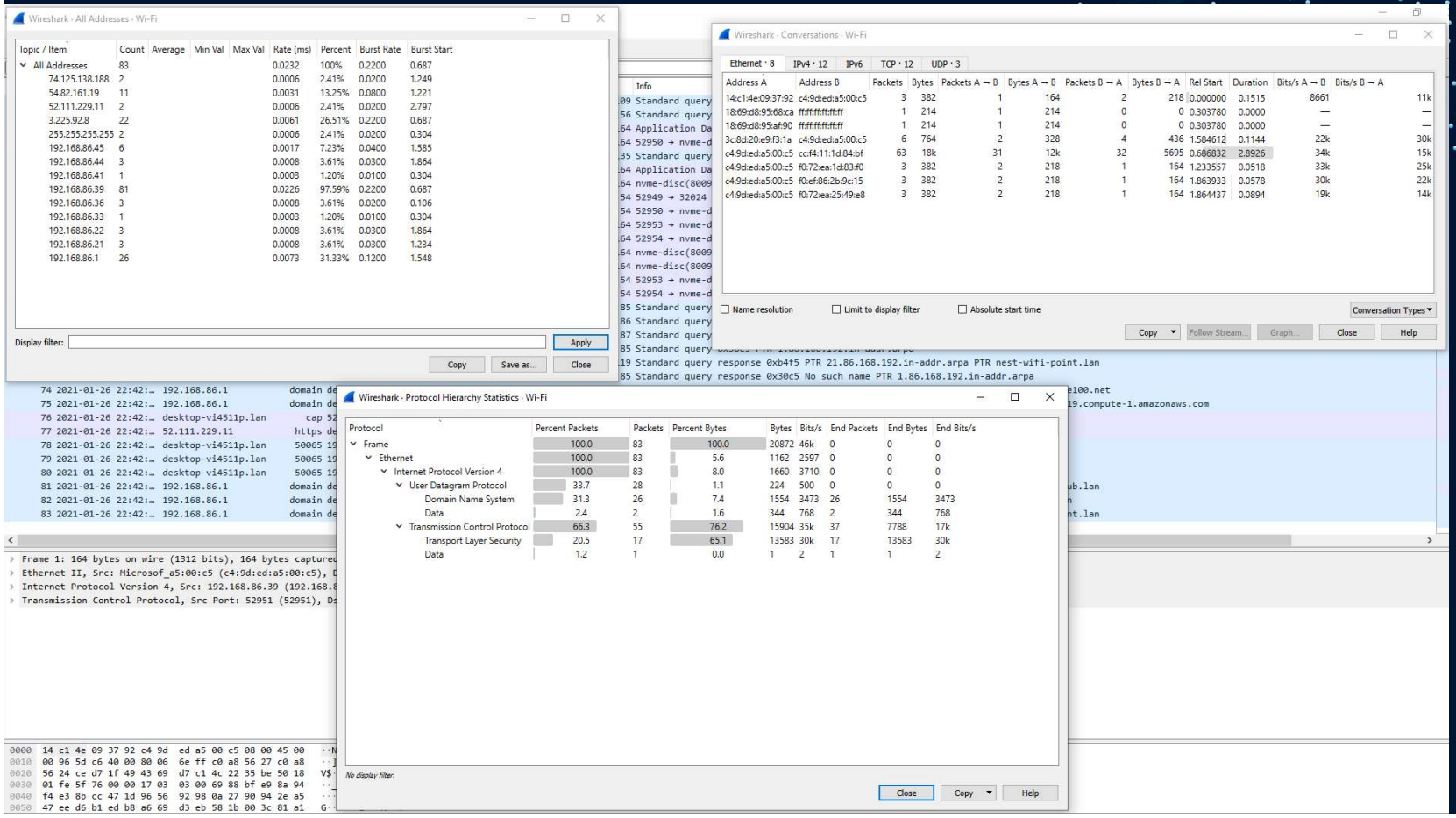


Display Filter Command Table

Display Filters	Result
ip.addr == x.x.x.x	Capture only traffic pertaining to a certain host. This is an OR statement.
ip.addr == x.x.x.x/24	Capture traffic pertaining to a specific network. This is an OR statement.
ip.src/dst == x.x.x.x	Capture traffic to or from a specific host
dns / tcp / ftp / arp / ip	filter traffic by a specific protocol. There are many more options.
tcp.port == x	filter by a specific tcp port.
tcp.port / udp.port != x	will capture everything except the port specified
and / or / not	AND will concatenate, OR will find either of two options, NOT will exclude your input option.

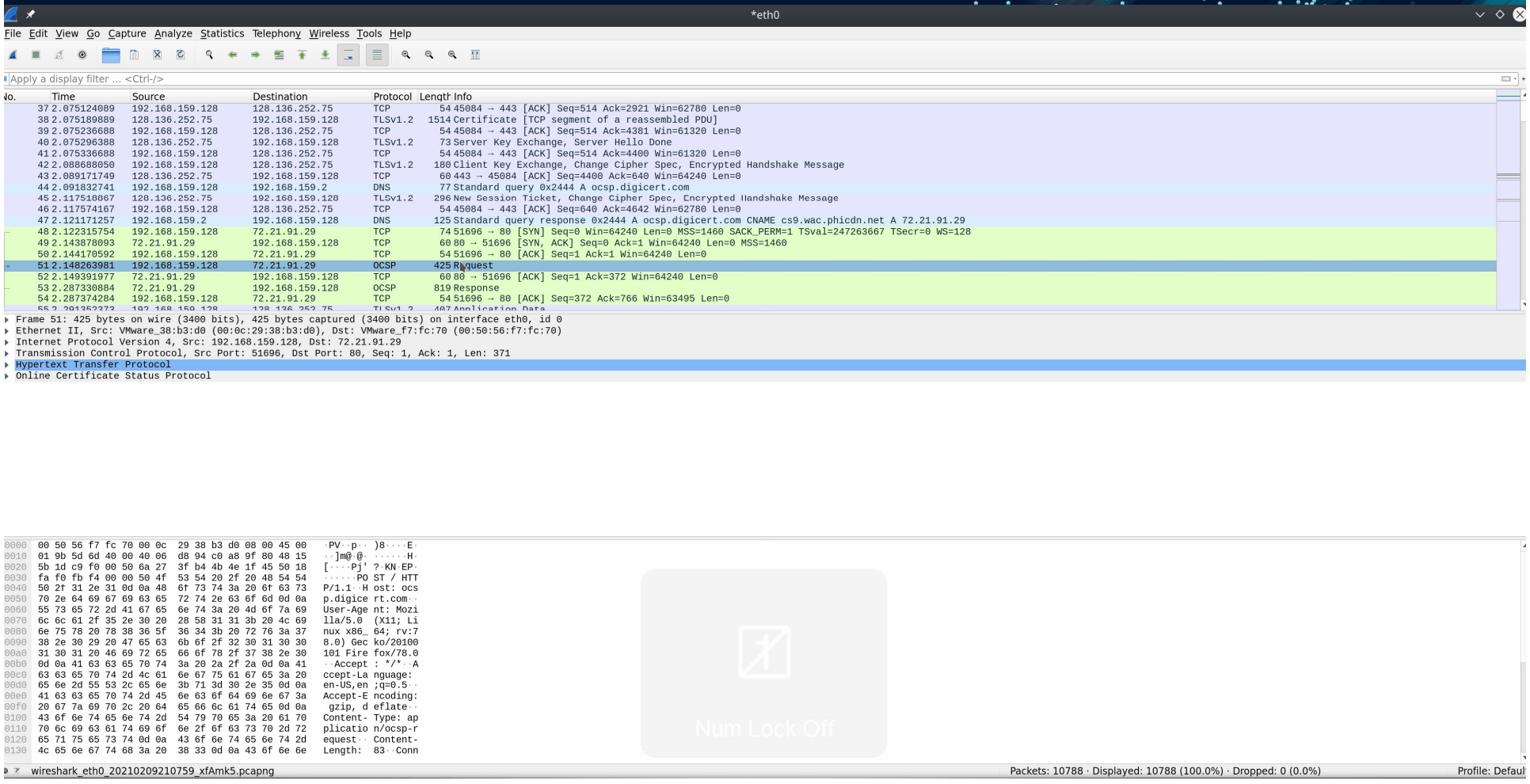
Wireshark Menu

The Statistics Tab

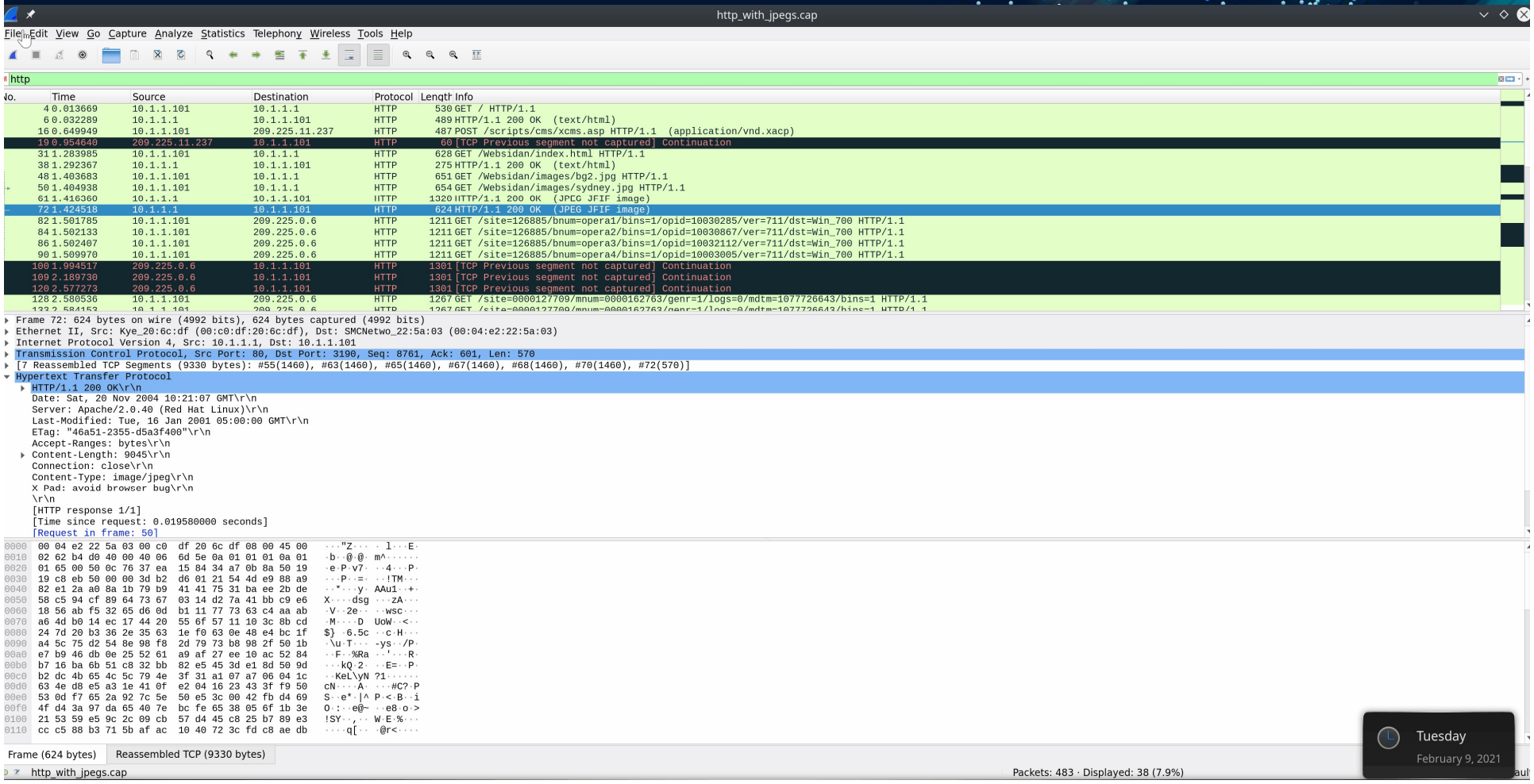


The Statistics Tab give us detailed reports about the network traffic being utilized. It can show us everything from the top talkers in our environment to specific conversations and even breakdown by IP and protocol.

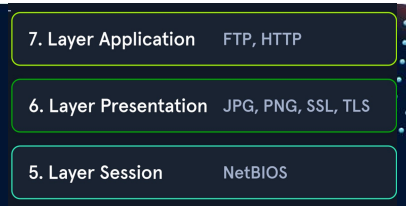
Following a packet (tcp.stream eq X)



Extract Files From The GUI



Application layer(5-7)



HTTP Methods(port 80 / 8000)

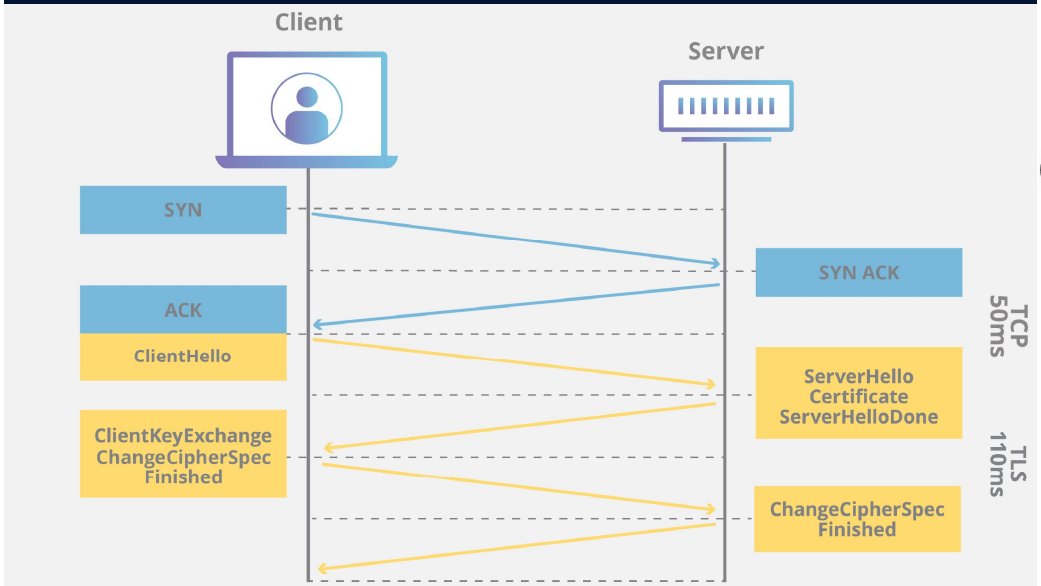
To perform operations such as fetching webpages, requesting items for download, or posting your most recent tweet all require the use of specific methods. These methods define the actions taken when requesting a URI

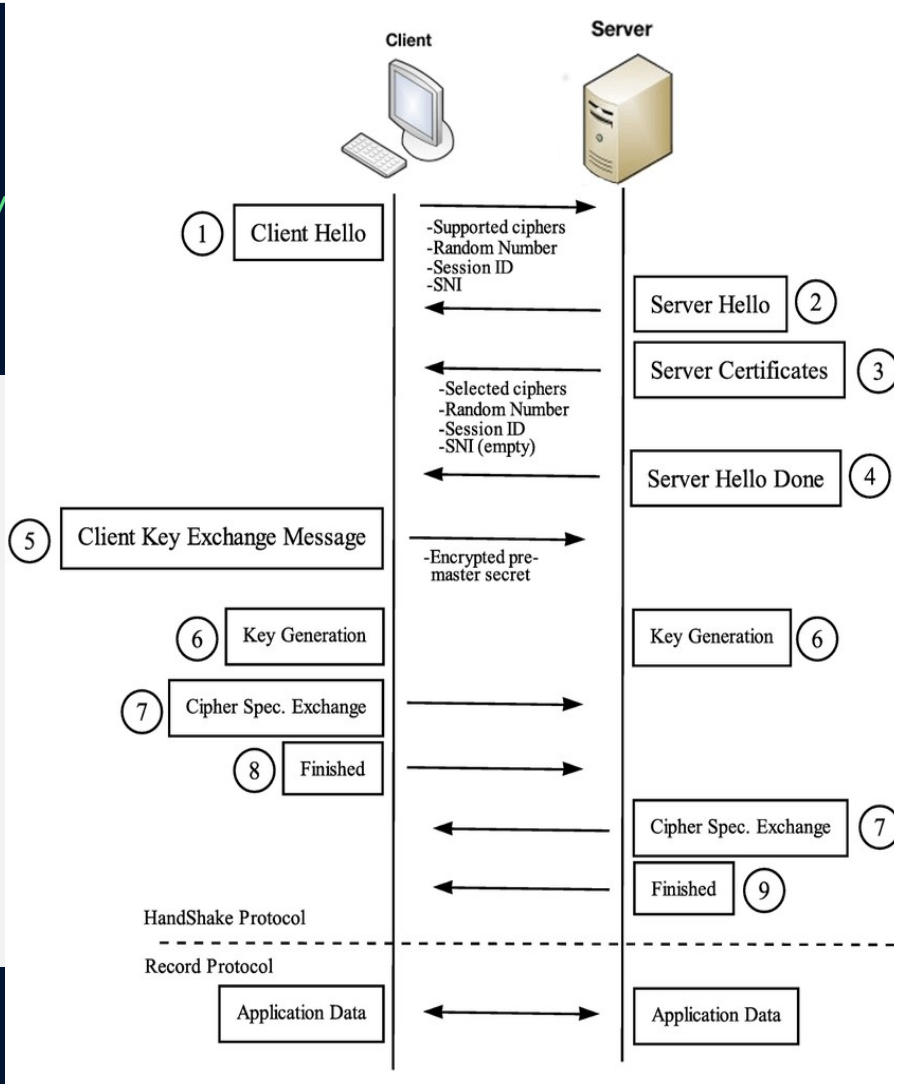
Method	Description
HEAD	required **is a safe method that requests a response from the server similar to a Get request except that the message body is not included. It is a great way to acquire more information about the server and its operational status.**
GET	required Get is the most common method used. It requests information and content from the server. For example, GET http://10.1.1.1/Webserver/index.html requests the index.html page from the server based on our supplied URI.
POST	optional Post is a way to submit information to a server based on the fields in the request. For example, submitting a message to a Facebook post or website forum is a POST action. The actual action taken can vary based on the server, and we should pay attention to the response codes sent back to validate the action.
PUT	optional Put will take the data appended to the message and place it under the requested URI. If an item does not exist there already, it will create one with the supplied data. If an object already exists, the new PUT will be considered the most up-to-date, and the object will be modified to match. The easiest way to visualize the differences between PUT and POST is to think of it like this; PUT will create or update an object at the URI supplied, while POST will create child entities at the provided URI. The action taken can be compared with the difference between creating a new file vs. writing comments about that file on the same page.
DELETE	optional **Delete does as the name implies. It will remove the object at the given URI.**
TRACE	optional Allows for remote server diagnosis. The remote server will echo the same request that was sent in its response if the TRACE method is enabled.
OPTIONS	optional **The Options method can gather information on the supported HTTP methods the server recognizes. This way, we can determine the requirements for interacting with a specific resource or server without actually requesting data or objects from it.**
CONNECT	optional Connect is reserved for use with Proxies or other security devices like firewalls. Connect allows for tunneling over HTTP. (SSL tunnels)

HTTPS(port 433 / 8433)

HTTP Secure (HTTPS) is a modification of the HTTP protocol designed to utilize Transport Layer Security (TLS) or Secure Sockets Layer (SSL)

TLS Handshake Via HTTPS





TLS Handshake Via HTTPS

Source	Destination	Protocol	Length	Info
192.168.86.243	104.20.55.68	TCP	78	60201 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64
104.20.55.68	192.168.86.243	TCP	66	443 → 60201 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=
192.168.86.243	104.20.55.68	TCP	54	60201 → 443 [ACK] Seq=1 Ack=1 Win=262144 Len=0
192.168.86.243	104.20.55.68	TLSv1.3	607	Client Hello
104.20.55.68	192.168.86.243	TCP	54	443 → 60201 [ACK] Seq=1 Ack=554 Win=67584 Len=0
104.20.55.68	192.168.86.243	TLSv1.3	266	Server Hello, Change Cipher Spec, Application Data
192.168.86.243	104.20.55.68	TCP	54	60201 → 443 [ACK] Seq=554 Ack=213 Win=261888 Len=0
192.168.86.243	104.20.55.68	TLSv1.3	118	Change Cipher Spec, Application Data
192.168.86.243	104.20.55.68	TLSv1.3	146	Application Data
192.168.86.243	104.20.55.68	TLSv1.3	1270	Application Data
192.168.86.243	104.20.55.68	TLSv1.3	107	Application Data
104.20.55.68	192.168.86.243	TCP	60	443 → 60201 [ACK] Seq=213 Ack=618 Win=67584 Len=0
104.20.55.68	192.168.86.243	TCP	60	443 → 60201 [ACK] Seq=213 Ack=710 Win=67584 Len=0
104.20.55.68	192.168.86.243	TLSv1.3	575	Application Data, Application Data
192.168.86.243	104.20.55.68	TCP	54	60201 → 443 [ACK] Seq=1979 Ack=734 Win=261568 Len=0
192.168.86.243	104.20.55.68	TLSv1.3	85	Application Data
104.20.55.68	192.168.86.243	TCP	60	443 → 60201 [ACK] Seq=734 Ack=1926 Win=69632 Len=0

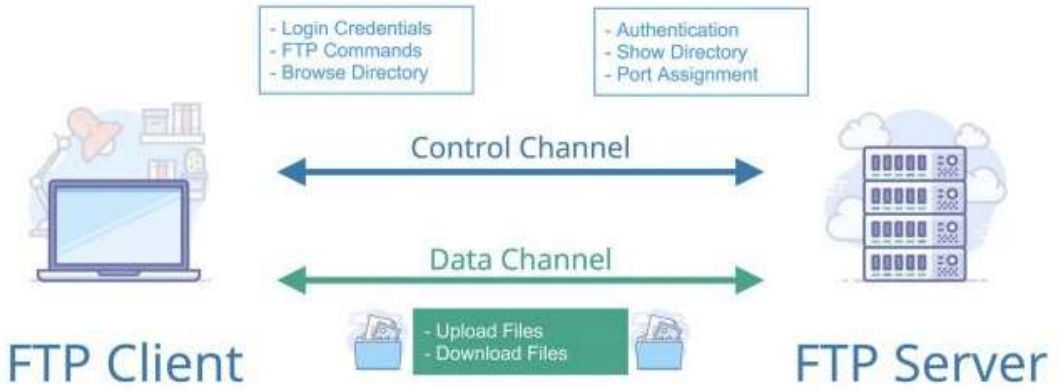
In the first few packets, we can see that the client establishes a session to the server using port 443 boxed in blue. This signals the server that it wishes to use HTTPS as the application communication protocol.

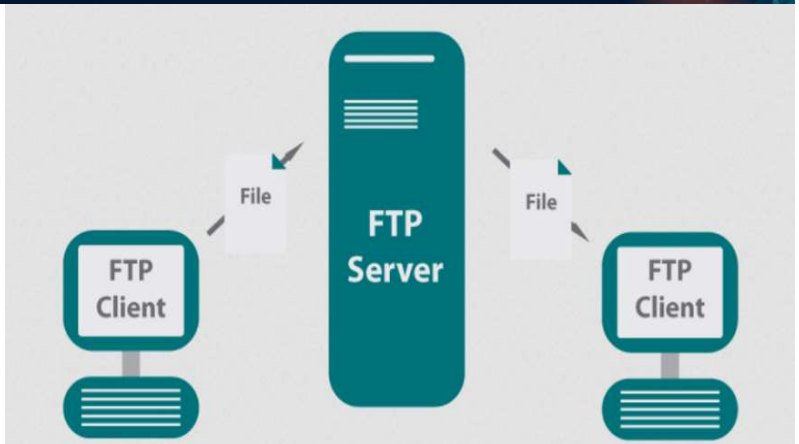
Once the session is established, all data and methods will be sent through the TLS connection and appear as TLS Application Data as seen in the red box. TLS is still using TCP as its transport protocol, so we will still see acknowledgment packets from the stream coming over port 443.

FTP(port 20 / 21)

Port 20 is used for data transfer, while port 21 is utilized for issuing commands controlling the FTP session

File Transfer Protocol (FTP) is an Application Layer protocol that enables quick data transfer between computing devices.

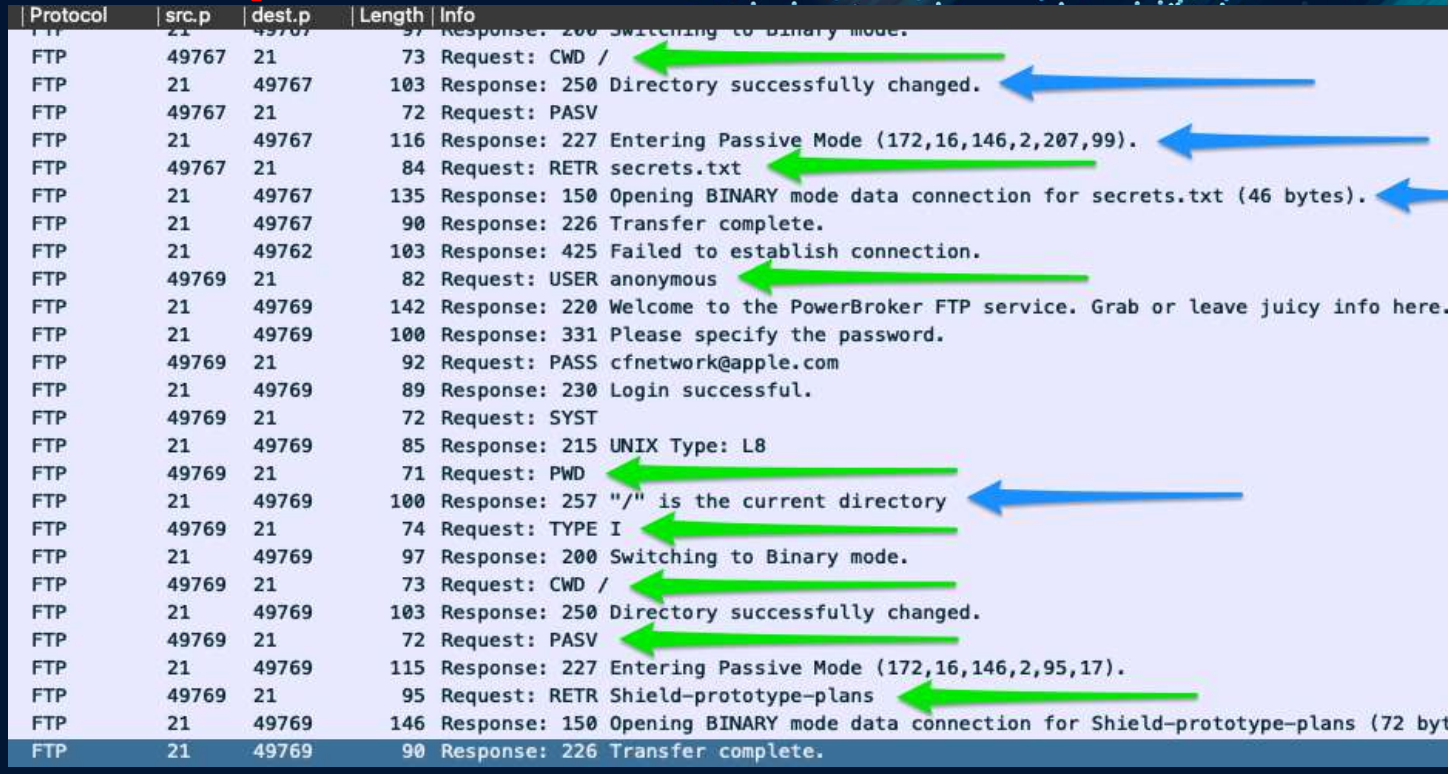




FTP Command & Response

Active mode is the default, with the server listening for a **PORT command** from the client to determine the data transfer port.

Passive mode, used for accessing servers behind firewalls or NAT, involves the client sending a PASV command to receive the IP and port details from the server for data transfer.

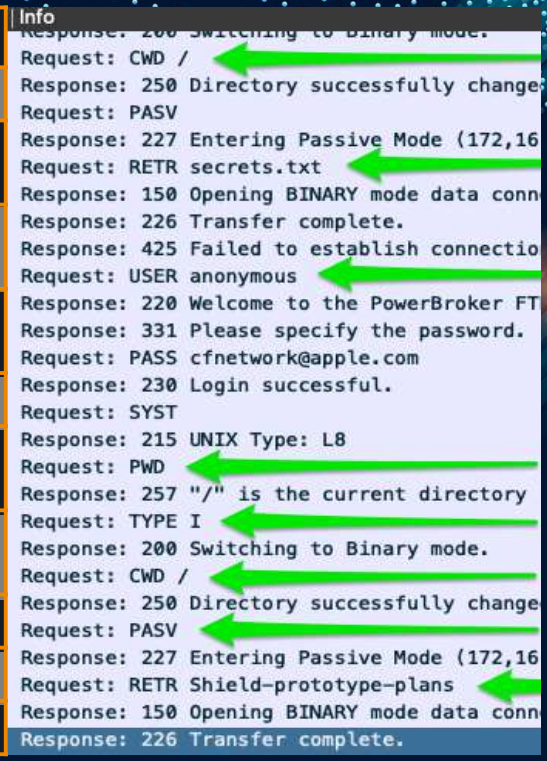


Client Request

FTP server respone

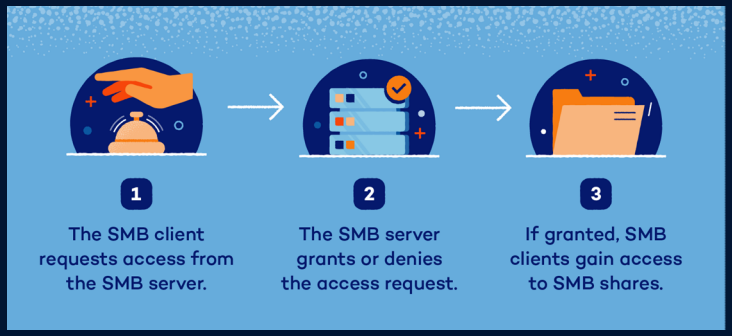
FTP Commands Table

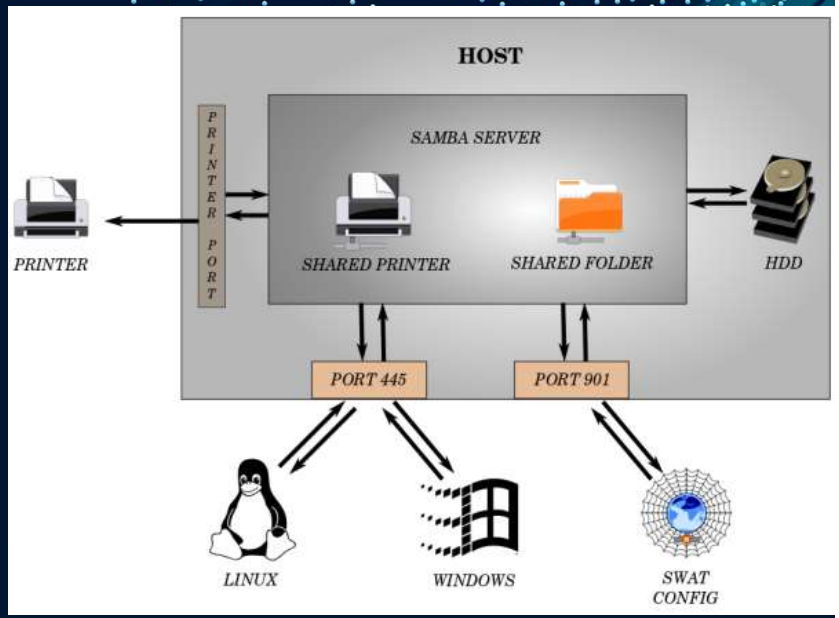
Command	Description
USER	specifies the user to log in as.
PASS	sends the password for the user attempting to log in.
PORT	when in active mode, this will change the data port used.
PASV	switches the connection to the server from active mode to passive.
LIST	displays a list of the files in the current directory.
CWD	will change the current working directory to one specified.
PWD	prints out the directory you are currently working in.
SIZE	will return the size of a file specified.
RETR	retrieves the file from the FTP server.
QUIT	ends the session.



SMB(port 137/138 [old] | port 445 (TCP) | port 139 (NetBIos))

Server Message Block (SMB) is a protocol most widely seen in Windows enterprise environments that enables sharing resources between hosts over common networking architectures.





As a user, SMB provides us easy and convenient access to resources like printers, shared drives, authentication servers, and more. For this reason, SMB is very attractive to potential attackers as well.

Like any other application that uses TCP as its transport mechanism, it will perform standard functions like the three-way handshake and acknowledging received packets.

SMB on the wire

Source	Destination	Protocol	src.p	dest.p	Length	Info
192.168.199.132	192.168.199.255	NBNS	137	137	92	Name query NB WPAD<00>
192.168.199.132	SCV	DNS	537…	53	86	Standard query 0x142e A officeclient.microsoft.com
192.168.199.133	SCV	DNS	643…	53	92	Standard query 0xfa7d A geo-prod.do.dsp.mp.microsoft.com
VMware_61:f5:5f	SCV	ARP			42	Who has 192.168.199.1? Tell 192.168.199.133
SCV	VMware_61:f5:5f	ARP			42	192.168.199.1 is at 00:50:56:c0:00:01
192.168.199.132	192.168.199.133	TCP	496…	445	66	49670 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.133	192.168.199.132	TCP	445	496…	66	445 → 49670 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.132	192.168.199.133	TCP	496…	445	54	49670 → 445 [ACK] Seq=1 Ack=1 Win=65536 Len=0
192.168.199.132	192.168.199.133	SMB	496…	445	213	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	506	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	232	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	566	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	220	Session Setup Request, NTLMSSP_NEGOTIATE
192.168.199.133	192.168.199.132	SMB2	445	496…	401	Session Setup Response, Error: STATUS_MORE_PROCESSING_REQUIRED, NTLMSSP_CHALLENGE
192.168.199.132	192.168.199.133	SMB2	496…	445	711	Session Setup Request, NTLMSSP_AUTH, User: DESKTOP-2AEFM7G\user
192.168.199.133	192.168.199.132	SMB2	445	496…	131	Session Setup Response, Error: STATUS_LOGON_FAILURE
192.168.199.132	192.168.199.133	TCP	496…	445	54	49670 → 445 [RST, ACK] Seq=1161 Ack=1389 Win=0 Len=0
192.168.199.132	192.168.199.133	TCP	496…	445	66	49671 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.133	192.168.199.132	TCP	445	496…	66	445 → 49671 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.132	192.168.199.133	TCP	496…	445	54	49671 → 445 [ACK] Seq=1 Ack=1 Win=65536 Len=0
192.168.199.132	192.168.199.133	SMB2	496…	445	232	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	566	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	220	Session Setup Request, NTLMSSP_NEGOTIATE
192.168.199.133	192.168.199.132	SMB2	445	496…	401	Session Setup Response, Error: STATUS_MORE_PROCESSING_REQUIRED, NTLMSSP_CHALLENGE
192.168.199.132	192.168.199.133	SMB2	496…	445	711	Session Setup Request, NTLMSSP_AUTH, User: DESKTOP-2AEFM7G\user
192.168.199.133	192.168.199.132	SMB2	445	496…	131	Session Setup Response, Error: STATUS_LOGON_FAILURE
192.168.199.132	192.168.199.133	TCP	496…	445	54	49671 → 445 [RST, ACK] Seq=1002 Ack=937 Win=0 Len=0
192.168.199.132	192.168.199.133	TCP	496…	445	66	49672 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.133	192.168.199.132	TCP	445	496…	66	445 → 49672 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.199.132	192.168.199.133	TCP	496…	445	54	49672 → 445 [ACK] Seq=1 Ack=1 Win=65536 Len=0
192.168.199.132	192.168.199.133	SMB2	496…	445	232	Negotiate Protocol Request
192.168.199.133	192.168.199.132	SMB2	445	496…	566	Negotiate Protocol Response
192.168.199.132	192.168.199.133	SMB2	496…	445	220	Session Setup Request, NTLMSSP_NEGOTIATE

Orange box: TCP-handshake

Blue box: ports

Green box: request from user and response from SMB

Did you saw something suspicious???

Common ports

Port Number	Protocol	Description
20	FTP-Data	Data channel for passing FTP files.
21	FTP-Command	Control channel for issuing commands to an FTP server.
22	SSH	Secure Shell Service port. Provides secure remote communications
23	Telnet	Telnet service provides cleartext communications between hosts.
25	SMTP	Simple Mail Transfer protocol. Utilized for email transmissions between servers.
53	DNS	Domain Name Services. Provides name resolution with multiple protocols
69	TFTP	Trivial File Transfer Protocol. A lightweight, minimal-function transfer protocol.
80	HTTP	HyperText Transfer Protocol. Provides dynamic web services
88	Kerberos	Providing cryptographic network authentication
110	POP3	Mail service utilized by clients to retrieve email from a server.
111	RPC	Remote Procedure Call. Remote service for managing network file systems.

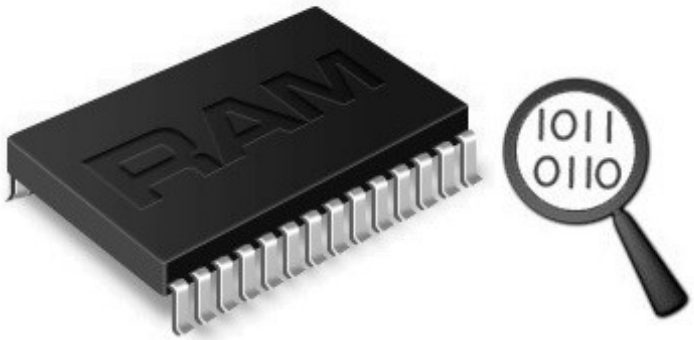
Port Number	Protocol	Description
115	SFTP	SSH File Transfer Protocol. An extension of SSH providing secure and reliable FTP services.
123	NTP	Network Time Protocol. Provides timing and sync services for network devices.
137	Netbios-NS	Local network name resolution.
139	Netbios-SSN	Provides session services for data transfer. Services like SMB can utilize it.
179	BGP	Border Gateway Protocol. BGP is a protocol for exchanging routing info with autonomous systems worldwide.
389	LDAP	Lightweight Directory Access Protocol. System agnostic authentication and authorization services.
443	HTTPS	HyperText Transfer Protocol Secure. An extension of HTTP utilizing SSL/TLS for encrypting the communications.
445	SMB	Server Message Block. SMB allows for the sharing of services, files, networking ports, and printers between hosts.


Memory dump

Memory Forensic

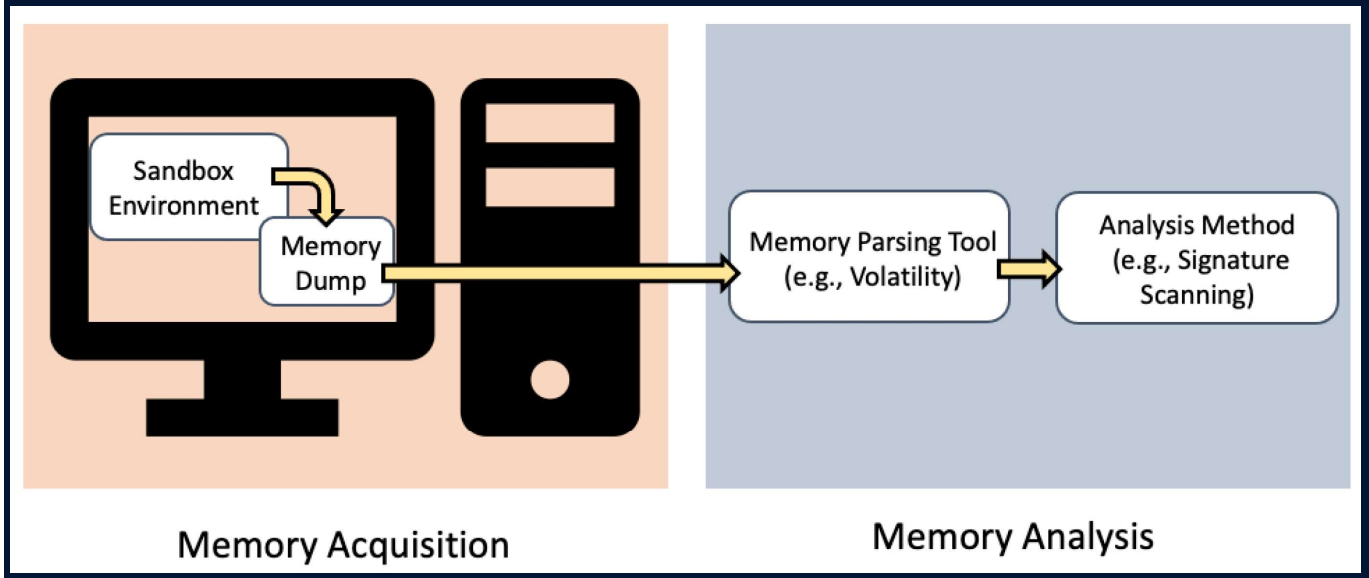
Memory forensics, also known as volatile memory analysis, is a specialized branch of digital forensics that focuses on the examination and analysis of the volatile memory (RAM) of a computer or digital device. Unlike traditional digital forensics, which involves analyzing data stored on non-volatile storage media like hard drives or solid-state drives, memory forensics deals with the live state of a system at a particular moment in time.

```
~/work/LZ77Huffman » vol -f Win8-d7fd4556.vmem prefetch
Volatility 3 Framework 2.0.1
Progress:  100.00               PDB scanning finished
ExecutableName  FileSize        PrefetchHash    LastExecution   ExecutionCounter

CONHOST.EXE     10648   0x3218e401      2022-07-18 12:22:04.000000      1
CONHOST.EXE     10648   0x3218e401      2022-07-18 12:22:04.000000      1
DLLHOST.EXE     18130   0x71214090      2022-07-18 12:22:14.000000      2
USERINIT.EXE    11664   0xf39ab672      2022-07-18 12:21:36.000000      1
SPPEXTCOMOBJ.EXE        17738   0xf8c1c601      2022-07-18 12:21:56.000000      1
NISSRV.EXE      30312   0x899f0efa      2022-07-18 12:21:33.000000      1
SEARCHINDEXER.EXE       100080  0x77d27bac      2022-07-18 12:21:35.000000      1
WMIPRVSE.EXE    20412   0x43972d0f      2022-07-18 12:21:48.000000      1
SEARCHFILTERHOST.EXE    15876   0xaa7a1fdd      2022-07-18 12:21:47.000000      1
SLUI.EXE        14426   0xa65918c4      2022-07-18 12:22:04.000000      1
EXPLORER.EXE    44598   0x7a3328da      2022-07-18 12:21:36.000000      1
```



Datas that can be found in RAM

- Network connections
- File handles and open Files
- Open registry keys
- Running processes on the system
- Loaded modules
- Loaded device drivers
- Command history and console sessions
- Kernel data structures
- User and credential information
- Malware artifacts
- System configuration
- Process memory regions



The Volatility Framework (NON-UI)

The preferred tool for conducting memory forensics is Volatility. Volatility is a leading open-source memory forensics framework.

```
BEATRIX volatility # python vol.py sockets -f /home/michael/stuxnet.vmem
Volatility Foundation Volatility Framework 2.3.1
Offset(V)       PID   Port  Proto Protocol        Address         Create Time
---------- -------- ------ ------ --------------- --------------- -----------
0x81dc2008      680    500     17 UDP             0.0.0.0         2010-10-29 17:09:05 UTC+0000
0x82061c08        4    445      6 TCP             0.0.0.0         2010-10-29 17:08:53 UTC+0000
0x82294aa8      940    135      6 TCP             0.0.0.0         2010-10-29 17:08:55 UTC+0000
0x821a5008      188   1025      6 TCP             127.0.0.1       2010-10-29 17:09:09 UTC+0000
0x81cb3d70     1080   1141     17 UDP             0.0.0.0         2010-10-31 16:36:16 UTC+0000
0x81da4d18      680      0    255 Reserved        0.0.0.0         2010-10-29 17:09:05 UTC+0000
0x81fdbe98     1032    123     17 UDP             127.0.0.1       2011-06-03 04:25:47 UTC+0000
0x81c79778     1080   1142     17 UDP             0.0.0.0         2010-10-31 16:36:16 UTC+0000
0x81c20898     1200   1900     17 UDP             127.0.0.1       2011-06-03 04:25:47 UTC+0000
0x82060008      680   4500     17 UDP             0.0.0.0         2010-10-29 17:09:05 UTC+0000
0x81cb9e98     1580   5152      6 TCP             127.0.0.1       2010-10-29 17:09:05 UTC+0000
0x81da54b0        4    445     17 UDP             0.0.0.0         2010-10-29 17:08:53 UTC+0000
```

Install (Volatility 3)

$ git clone https://github.com/volatilityfoundation/volatility3.git

```
┌──(raviel㉿kali)-[~/…/Tool/Volatility/Vol3/volatility3]
└─$ python3 vol.py -h
Volatility 3 Framework 2.5.2
usage: volatility [-h] [-c CONFIG] [--parallelism [{processes,threads,off}]] [-e EXTEND] [-p PLUGIN_
                  [-q] [-r RENDERER] [-f FILE] [--write-config] [--save-config SAVE_CONFIG] [--clear
                  [--single-location SINGLE_LOCATION] [--stackers [STACKERS ...]] [--single-swap-loc
                  plugin ...

An open-source memory forensics framework

options:
  -h, --help            Show this help message and exit, for specific plugin options use 'volatility
  -c CONFIG, --config CONFIG
                        Load the configuration from a json file
  --parallelism [{processes,threads,off}]
                        Enables parallelism (defaults to off if no argument given)
  -e EXTEND, --extend EXTEND
                        Extend the configuration with a new (or changed) setting
  -p PLUGIN_DIRS, --plugin-dirs PLUGIN_DIRS
                        Semi-colon separated list of paths to find plugins
  -s SYMBOL_DIRS, --symbol-dirs SYMBOL_DIRS
                        Semi-colon separated list of paths to find symbols
  -v, --verbosity       Increase output verbosity
  -l LOG, --log LOG     Log output to a file as well as the console
  -o OUTPUT_DIR, --output-dir OUTPUT_DIR
```

Install (Volatility 2)

$ wget https://github.com/volatilityfoundation/volatility/releases/download/2.6.1/volatility_2.6_lin64_standalone.zip

```
┌──(raviel㉿kali)-[~/Desktop/dump/vol2.6]
└─$ wget https://github.com/volatilityfoundation/volatility/releases/download/2.6.1/volatility_2.6_lin64_standalone.zip
--2024-06-11 06:33:06--  https://github.com/volatilityfoundation/volatility/releases/download/2.6.1/volatility_2.6_lin64_standalone
.zip
Resolving github.com (github.com)... 20.205.243.166
Connecting to github.com (github.com)|20.205.243.166|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/19114225/3488f159-fa81-4ffa-b1c8-502ac0f4cba
e?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240611%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20
240611T103305Z&X-Amz-Expires=300&X-Amz-Signature=0db7f58889fa1421e39d8a489b09f4f48c11b72a5bbc0d149464607eaaf935e1&X-Amz-SignedHeade
rs=host&actor_id=0&key_id=0&repo_id=19114225&response-content-disposition=attachment%3B%20filename%3Dvolatility_2.6_lin64_standalon
e.zip&response-content-type=application%2Foctet-stream [following]
--2024-06-11 06:33:07--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/19114225/3488f159-fa81-4ffa-b
1c8-502ac0f4cbae?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240611%2Fus-east-1%2Fs3%2Faws4_reques
t&X-Amz-Date=20240611T103305Z&X-Amz-Expires=300&X-Amz-Signature=0db7f58889fa1421e39d8a489b09f4f48c11b72a5bbc0d149464607eaaf935e1&X-
Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=19114225&response-content-disposition=attachment%3B%20filename%3Dvolatility_2.6_
lin64_standalone.zip&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.110.133, 185.199.109.133, 185.199.111.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 14737820 (14M) [application/octet-stream]
Saving to: 'volatility_2.6_lin64_standalone.zip'

volatility_2.6_lin64_standalone. 100%[═══════════════════════════════════════════════════════════════════>]  14.05M  2.45MB/s    in 5.7s

2024-06-11 06:33:13 (2.47 MB/s) - 'volatility_2.6_lin64_standalone.zip' saved [14737820/14737820]


┌──(raviel㉿kali)-[~/Desktop/dump/vol2.6]
└─$ unzip volatility_2.6_lin64_standalone.zip
Archive:  volatility_2.6_lin64_standalone.zip
   creating: volatility_2.6_lin64_standalone/
  inflating: volatility_2.6_lin64_standalone/AUTHORS.txt
  inflating: volatility_2.6_lin64_standalone/CREDITS.txt
  inflating: volatility_2.6_lin64_standalone/LEGAL.txt
  inflating: volatility_2.6_lin64_standalone/LICENSE.txt
  inflating: volatility_2.6_lin64_standalone/README.txt
  inflating: volatility_2.6_lin64_standalone/volatility_2.6_lin64_standalone

┌──(raviel㉿kali)-[~/Desktop/dump/vol2.6]
└─$ cd volatility_2.6_lin64_standalone

┌──(raviel㉿kali)-[~/…/Trash/files/vol2.6/volatility_2.6_lin64_standalone]
└─$ chmod +x volatility_2.6_lin64_standalone

┌──(raviel㉿kali)-[~/…/Trash/files/vol2.6/volatility_2.6_lin64_standalone]
└─$ mv volatility_2.6_lin64_standalone vol2.6

┌──(raviel㉿kali)-[~/…/Trash/files/vol2.6/volatility_2.6_lin64_standalone]
└─$ ./vol2.6 -h
Volatility Foundation Volatility Framework 2.6
Usage: Volatility - A memory forensics analysis platform.
```

IDENTIFY OS PROFILE

Volatility 2

```
vol.py -f “/path/to/file” imageinfo
vol.py -f “/path/to/file” kdbgscan
```

Volatility 3

```
vol.py -f “/path/to/file” windows.info
```

PROCESS INFORMATION

PSLIST (lists the current processes running in memory)

Volatility 2

```
vol.py -f “/path/to/file” --profile <profile> pslist
vol.py -f “/path/to/file” --profile <profile> psscan
vol.py -f “/path/to/file” --profile <profile> pstree
vol.py -f “/path/to/file” --profile <profile> psxview
```

Volatility 3

```
vol.py -f “/path/to/file” windows.pslist
vol.py -f “/path/to/file” windows.psscan
vol.py -f “/path/to/file” windows.pstree
```

PROCESS INFORMATION

PROCDUMP

Volatility 2

vol.py -f “/path/to/file” --profile <profile> procdump -p <PID> --dump-dir=“/path/to/dir”

Volatility 3

vol.py -f “/path/to/file” -o “/path/to/dir” windows.dumpfiles --pid <PID>

PROCESS INFORMATION

MEMDUMP

Volatility 2

vol.py -f “/path/to/file” --profile <profile> memdump -p <PID> --dump-dir=“/path/to/dir”

Volatility 3

vol.py -f “/path/to/file” -o “/path/to/dir” windows.memmap --dump --pid <PID>

PROCESS INFORMATION

HANDLES

Volatility 2

vol.py -f “/path/to/file” --profile <profile> handles -p <PID>

Volatility 3

vol.py -f “/path/to/file” windows.handles --pid <PID>

PROCESS INFORMATION

DLLS

Volatility 2

vol.py -f “/path/to/file” --profile <profile> dlllist -p <PID>

Volatility 3

vol.py -f “/path/to/file” windows.dlllist --pid <PID>

PROCESS INFORMATION

CMDLINES

Volatility 2

vol.py -f “/path/to/file” --profile <profile> cmdline
vol.py -f “/path/to/file” --profile <profile> cmdscan
vol.py -f “/path/to/file” --profile <profile> consoles

Volatility 3

vol.py -f “/path/to/file” windows.cmdline

NETWORK INFORMATION

NETSCAN

Volatility 2

vol.py -f “/path/to/file” --profile <profile> netscan
vol.py -f “/path/to/file” --profile <profile> netstat

Volatility 3

vol.py -f “/path/to/file” windows.netscan
vol.py -f “/path/to/file” windows.netstat

REGISTRY

HIVELIST

Volatility 2

vol.py -f “/path/to/file” --profile <profile> hivescan
vol.py -f “/path/to/file” --profile <profile> hivelist

Volatility 3

vol.py -f “/path/to/file” windows.registry.hivescan
vol.py -f “/path/to/file” windows.registry.hivelist

REGISTRY

PRINTKEY

Volatility 2

vol.py -f “/path/to/file” --profile <profile> printkey
vol.py -f “/path/to/file” --profile <profile> printkey –K “Software\Microsoft\Windows\CurrentVersion”

Volatility 3

vol.py -f “/path/to/file” windows.registry.printkey
vol.py -f “/path/to/file” windows.registry.printkey --key “Software\Microsoft\Windows\CurrentVersion”

REGISTRY

HIVEDUMP

Volatility 2

vol.py -f “/path/to/file” --profile hivedump -o <offset>

Volatility 3

vol.py -f “/path/to/file” windows.filescan

REGISTRY

HIVELIST

Volatility 2

vol.py -f “/path/to/file” --profile <profile> hivescan
vol.py -f “/path/to/file” --profile <profile> hivelist

Volatility 3

vol.py -f “/path/to/file” windows.registry.hivescan
vol.py -f “/path/to/file” windows.registry.hivelist

FILES

FILESCAN

Volatility 2

vol.py -f “/path/to/file” --profile <profile> filescan

Volatility 3

vol.py -f “/path/to/file” windows.filescan

FILES

FILEDUMP

Volatility 2

vol.py -f “/path/to/file” --profile <profile> dumpfiles --dump-dir=“/path/to/dir”
vol.py -f “/path/to/file” --profile <profile> dumpfiles --dump-dir=“/path/to/dir” -Q <offset>
vol.py -f “/path/to/file” --profile <profile> dumpfiles --dump-dir=“/path/to/dir” -p <PID>

Volatility 3

vol.py -f “/path/to/file” -o “/path/to/dir” windows.dumpfiles
vol.py -f “/path/to/file” -o “/path/to/dir” windows.dumpfiles --virtaddr <offset>
vol.py -f “/path/to/file” -o “/path/to/dir” windows.dumpfiles --physaddr <offset>

MISCELLANEOUS

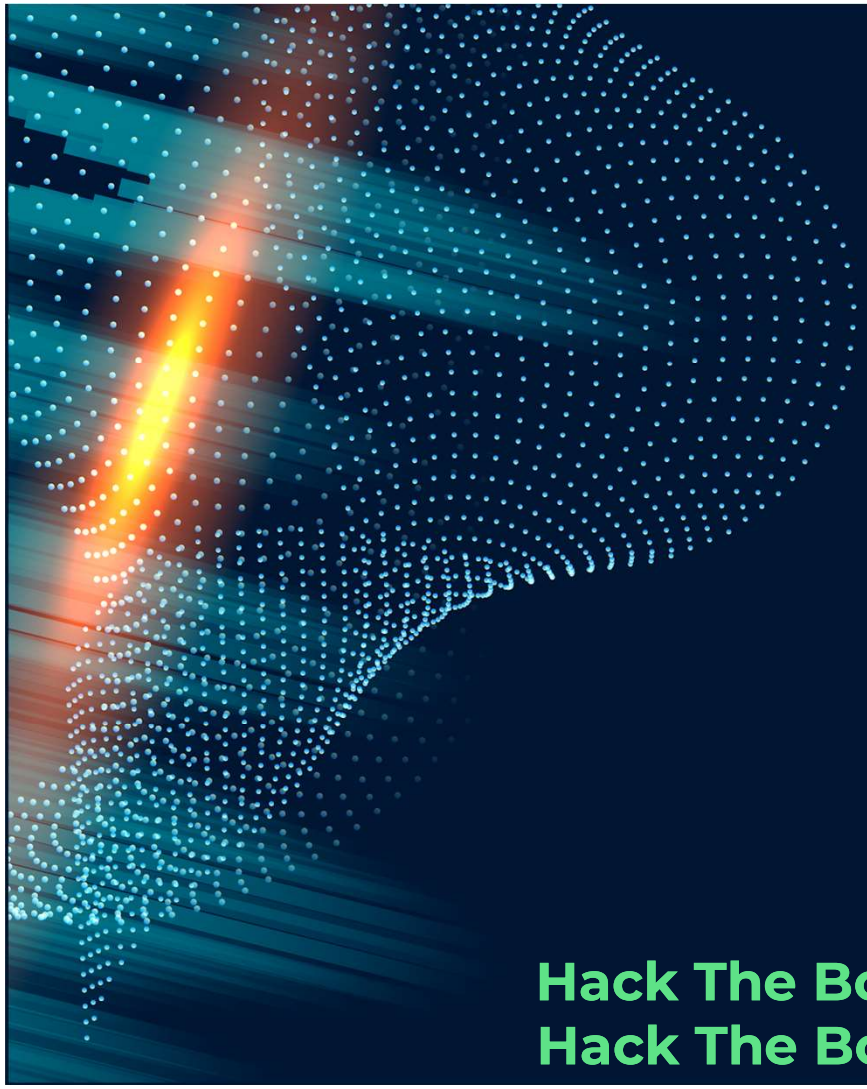
MALFIND

Volatility 2

vol.py -f “/path/to/file” --profile <profile> malfind

Volatility 3

vol.py -f “/path/to/file” windows.malfind



THE END

Contacts:

Facebook:
https://www.facebook.com/Kann.Raviel
Discord:
raviyelna

Reference:

CTF101
Hack The Box Academy – Intro to Network traffic
Hack The Box Academy – Intro to Digital forensic