IntroToCTF - Forensics

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Introduction

- Oscar I, Cyber Soc Welfare
- Interned at CrowdStrike last summer doing incident response – investigating real world digital crimes using forensics.
- Half an hour talk on the basics and concepts of Forensics challenges and common tools used.
- Hour practical session for solving selected challenges (easy and medium level)
- No disk forensics today (too long for todays session)

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Forensics, what is

Concept

Forensics in its simplest form is the inspection of various pieces of digital evidence to find something interesting (the flag).

“Interesting” could be:

- *IP addresses*
- *A file*
- *Evidence that something happened on a system*

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

How do we do forensics challenges?

In a CTF scenario

Forensic CTF challenges are a bit like using a Swiss army knife.

You have a problem, and you must find the tool in your arsenal that best suits that problem.

Some tools you will use more than others, and sometimes you will encounter a problem that needs a new tool!

Reminder

Hopefully, ***most*** of you have been able to download the tool list, we put out on the discord. If you need help with this , please ask **AFTER** the slides.

But if not please start getting these downloaded, they will all be used today:

Volatility - https://github.com/volatilityfoundation/volatility.git

^ Clone this repository with “git clone”

Wireshark – *“sudo apt install wireshark”*

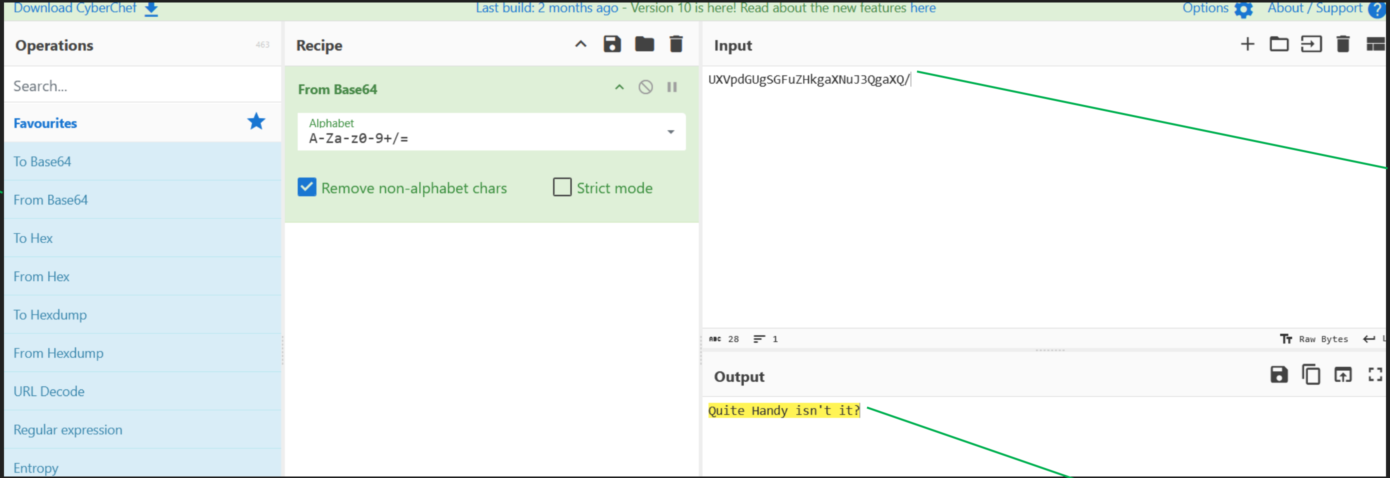
A Linux VM is needed, if you have kali Linux Wireshark is already installed!

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

CyberChef

- Tool made by GCHQ to translate and modify data to different encoding formats; people like to hide things by encoding data known as Obfuscation!

All the ways we can manipulate data



From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars ☐ Strict mode

```
UXVpdGUgSGFuZHkgaXNuJ3QgaXQ/
```

28 1

```
Quite Handy isn't it?
```

Base-64 encoded string

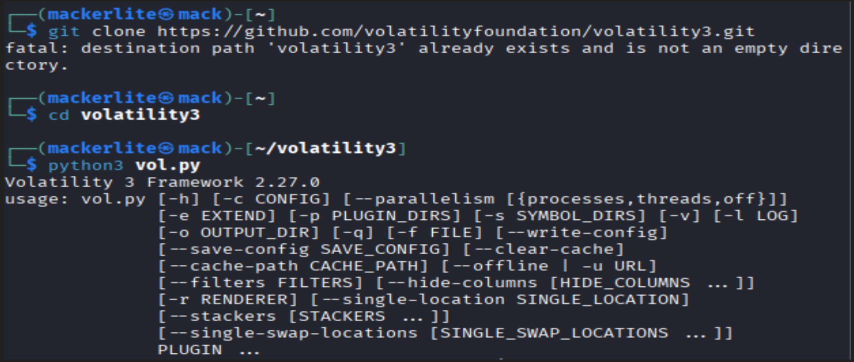
Translated output

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Memory Forensics – The Basics

Memory Forensics utilizes a command line utility called volatility (v3), which enables us to parse and analyze samples of memory (RAM) from machines.

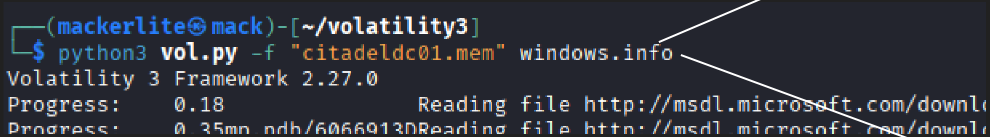
This can is very useful for analyzing a device suspected of doing something malicious as it tells us what was actively running in memory when the capture was taken.



```
┌──(mackerlite㉿mack)-[~]
└─$ git clone https://github.com/volatilityfoundation/volatility3.git
fatal: destination path 'volatility3' already exists and is not an empty dire
ctory.

┌──(mackerlite㉿mack)-[~]
└─$ cd volatility3

┌──(mackerlite㉿mack)-[~/volatility3]
└─$ python3 vol.py
Volatility 3 Framework 2.27.0
usage: vol.py [-h] [-c CONFIG] [--parallelism [{processes,threads,off}]]
              [-e EXTEND] [-p PLUGIN_DIRS] [-s SYMBOL_DIRS] [-v] [-l LOG]
              [-o OUTPUT_DIR] [-q] [-f FILE] [--write-config]
              [--save-config SAVE_CONFIG] [--clear-cache]
              [--cache-path CACHE_PATH] [--offline | -u URL]
              [--filters FILTERS] [--hide-columns [HIDE_COLUMNS ...]]
              [-r RENDERER] [--single-location SINGLE_LOCATION]
              [--stackers [STACKERS ...]]
              [--single-swap-locations [SINGLE_SWAP_LOCATIONS ...]]
              PLUGIN ...
```

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

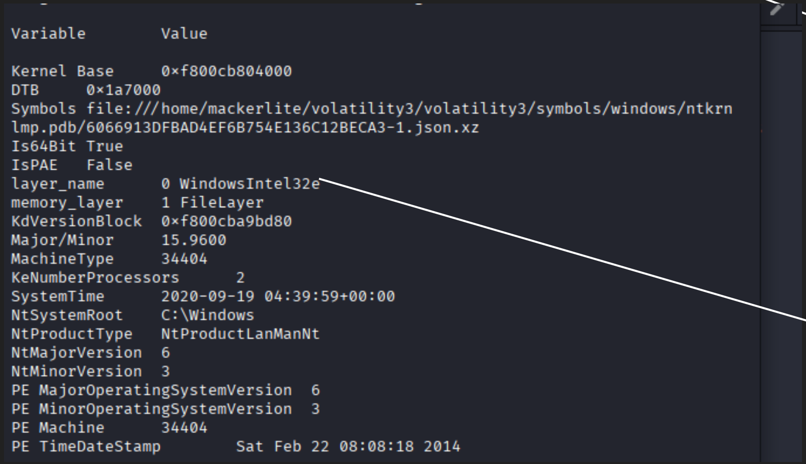
Analyzing our image



```
┌──(mackerlite㉿mack)-[~/volatility3]
└─$ python3 vol.py -f "citadeldc01.mem" windows.info
Volatility 3 Framework 2.27.0
Progress:     0.18               Reading file http://msdl.microsoft.com/downl
Progress:     0.35mp pdb/6066913DReading file http://msdl.microsoft.com/downl
```

Different versions of volatility use different plugin formats version 2 uses “imageinfo” instead.

Volatility works with specific profiles for different operating systems so we must first identify what OS we are dealing with! These can be .mem or .raw files.



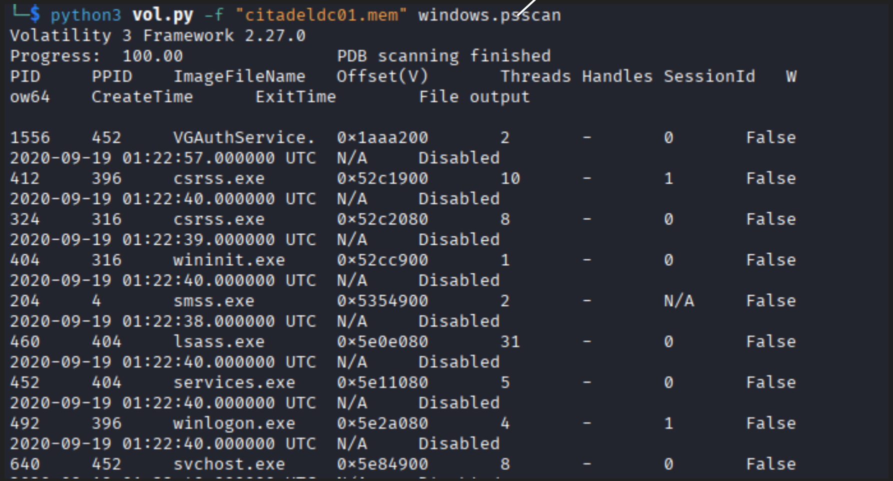
```
Variable        Value

Kernel Base     0×f800cb804000
DTB     0×1a7000
Symbols file:///home/mackerlite/volatility3/volatility3/symbols/windows/ntkrn
lmp.pdb/6066913DFBAD4EF6B754E136C12BECA3-1.json.xz
Is64Bit True
IsPAE   False
layer_name      0 WindowsIntel32e
memory_layer    1 FileLayer
KdVersionBlock  0×f800cba9bd80
Major/Minor     15.9600
MachineType     34404
KeNumberProcessors      2
SystemTime      2020-09-19 04:39:59+00:00
NtSystemRoot    C:\Windows
NtProductType   NtProductLanManNt
NtMajorVersion  6
NtMinorVersion  3
PE MajorOperatingSystemVersion  6
PE MinorOperatingSystemVersion  3
PE Machine      34404
PE TimeDateStamp        Sat Feb 22 08:08:18 2014
```

Windows Intel 32 machine has been found and processed

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

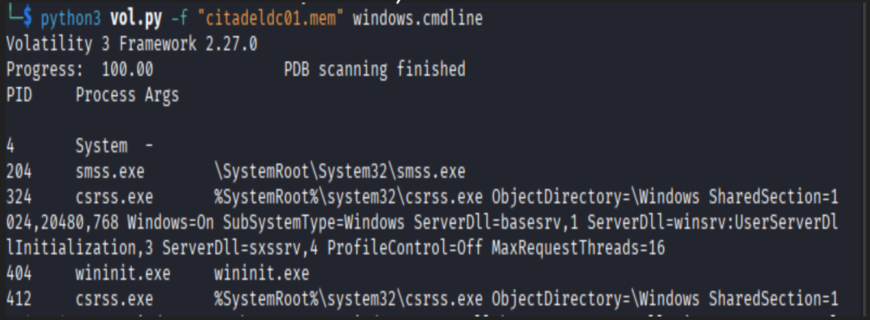
How to hunt in memory

See all running processes that were in memory at the time



```
└─$ python3 vol.py -f "citadeldc01.mem" windows.psscan
Volatility 3 Framework 2.27.0
Progress:  100.00               PDB scanning finished
PID     PPID    ImageFileName   Offset(V)       Threads Handles SessionId   W
ow64    CreateTime      ExitTime        File output

1556    452     VGAuthService.  0x1aaa200       2       -       0       False
2020-09-19 01:22:57.000000 UTC  N/A     Disabled
412     396     csrss.exe       0x52c1900       10      -       1       False
2020-09-19 01:22:40.000000 UTC  N/A     Disabled
324     316     csrss.exe       0x52c2080       8       -       0       False
2020-09-19 01:22:39.000000 UTC  N/A     Disabled
404     316     wininit.exe     0x52cc900       1       -       0       False
2020-09-19 01:22:40.000000 UTC  N/A     Disabled
204     4       smss.exe        0x5354900       2       -       N/A     False
2020-09-19 01:22:38.000000 UTC  N/A     Disabled
460     404     lsass.exe       0x5e0e080       31      -       0       False
2020-09-19 01:22:40.000000 UTC  N/A     Disabled
452     404     services.exe    0x5e11080       5       -       0       False
2020-09-19 01:22:40.000000 UTC  N/A     Disabled
492     396     winlogon.exe    0x5e2a080       4       -       1       False
2020-09-19 01:22:40.000000 UTC  N/A     Disabled
640     452     svchost.exe     0x5e84900       8       -       0       False
```

What commands are currently in use in memory (and what are their arguments)



```
└─$ python3 vol.py -f "citadeldc01.mem" windows.cmdline
Volatility 3 Framework 2.27.0
Progress:  100.00               PDB scanning finished
PID     Process Args

4       System  -
204     smss.exe        \SystemRoot\System32\smss.exe
324     csrss.exe       %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1
024,20480,768 Windows=On SubSystemType=Windows ServerDll=basesrv,1 ServerDll=winsrv:UserServerDl
lInitialization,3 ServerDll=sxssrv,4 ProfileControl=Off MaxRequestThreads=16
404     wininit.exe     wininit.exe
412     csrss.exe       %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1
```

And more….

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Networking primer

- There is a LOT of noise in Wireshark captures so knowing what common network communications are is quite helpful to then find something interesting. All of these be filtered for in Wireshark.

TCP/UDP – Primary networking protocols for communicating data on the internet in 'packets'.
SYN/ACK/SYN-ACK – How two devices agree to communicate over TCP.
ARP – How devices find each other to talk.
DNS – Domain resolutions (think like a phonebook for the internet)
Port – Endpoint for a particular service on a device e.g. DNS runs on port 53. "tcp.port==" is a filter we can use, but what is a filter?

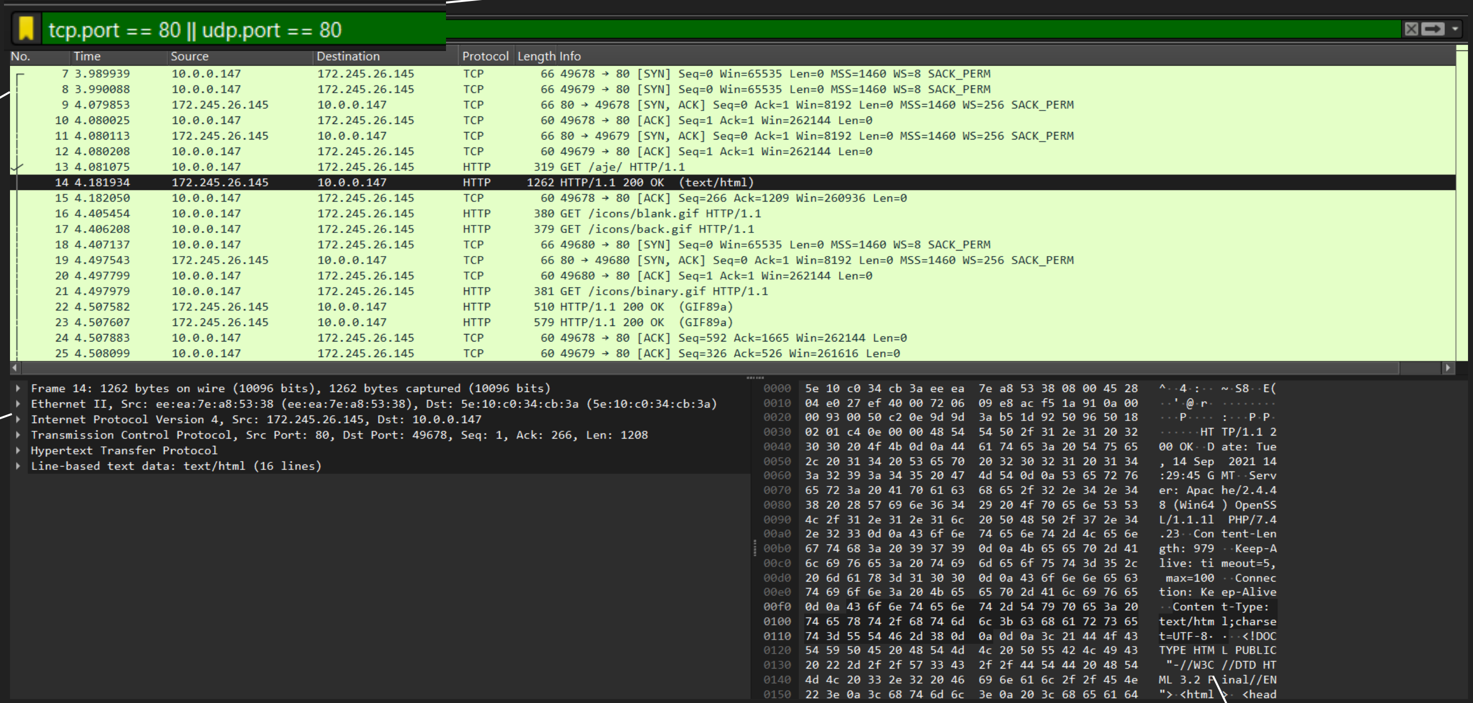
Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Network forensics - Wireshark

Filters! Packet streams can be quite large...

Stream of packets during network capture (.pcap).

Information contained within the packet



```
tcp.port == 80 || udp.port == 80
```

```
No.  Time        Source          Destination     Protocol Length Info
   7 3.989939    10.0.0.147      172.245.26.145  TCP        66 49678 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM
   8 3.990088    10.0.0.147      172.245.26.145  TCP        66 49679 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM
   9 4.079853    172.245.26.145  10.0.0.147      TCP        66 80 → 49678 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
  10 4.080025    10.0.0.147      172.245.26.145  TCP        60 49678 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
  11 4.080113    172.245.26.145  10.0.0.147      TCP        66 80 → 49679 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
  12 4.080208    10.0.0.147      172.245.26.145  TCP        60 49679 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
  13 4.081075    10.0.0.147      172.245.26.145  HTTP      319 GET /aje/ HTTP/1.1
  14 4.181934    172.245.26.145  10.0.0.147      HTTP     1262 HTTP/1.1 200 OK  (text/html)
  15 4.182050    10.0.0.147      172.245.26.145  TCP        60 49678 → 80 [ACK] Seq=266 Ack=1209 Win=260936 Len=0
  16 4.405454    10.0.0.147      172.245.26.145  HTTP      380 GET /icons/blank.gif HTTP/1.1
  17 4.406208    10.0.0.147      172.245.26.145  HTTP      379 GET /icons/back.gif HTTP/1.1
  18 4.407137    10.0.0.147      172.245.26.145  TCP        66 49680 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM
  19 4.497543    172.245.26.145  10.0.0.147      TCP        66 80 → 49680 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
  20 4.497799    10.0.0.147      172.245.26.145  TCP        60 49680 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
  21 4.497979    10.0.0.147      172.245.26.145  HTTP      381 GET /icons/binary.gif HTTP/1.1
  22 4.507582    172.245.26.145  10.0.0.147      HTTP      510 HTTP/1.1 200 OK  (GIF89a)
  23 4.507607    172.245.26.145  10.0.0.147      HTTP      579 HTTP/1.1 200 OK  (GIF89a)
  24 4.507883    10.0.0.147      172.245.26.145  TCP        60 49678 → 80 [ACK] Seq=592 Ack=1665 Win=262144 Len=0
  25 4.508099    10.0.0.147      172.245.26.145  TCP        60 49679 → 80 [ACK] Seq=326 Ack=526 Win=261616 Len=0
```

```
▸ Frame 14: 1262 bytes on wire (10096 bits), 1262 bytes captured (10096 bits)
▸ Ethernet II, Src: ee:ea:7e:a8:53:38 (ee:ea:7e:a8:53:38), Dst: 5e:10:c0:34:cb:3a (5e:10:c0:34:cb:3a)
▸ Internet Protocol Version 4, Src: 172.245.26.145, Dst: 10.0.0.147
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 49678, Seq: 1, Ack: 266, Len: 1208
▸ Hypertext Transfer Protocol
▸ Line-based text data: text/html (16 lines)
```

```
0000  5e 10 c0 34 cb 3a ee ea  7e a8 53 38 08 00 45 28   ^··4·:·· ~·S8··E(
0010  04 e0 27 ef 40 00 72 06  09 e8 ac f5 1a 91 0a 00   ··'·@·r· ········
0020  00 93 00 50 c2 0e 9d 9d  3a b5 1d 92 50 96 50 18   ···P···· :···P·P·
0030  02 01 c4 0e 00 00 48 54  54 50 2f 31 2e 31 20 32   ······HT TP/1.1 2
0040  30 30 20 4f 4b 0d 0a 44  61 74 65 3a 20 54 75 65   00 OK··D ate: Tue
0050  2c 20 31 34 20 53 65 70  20 32 30 32 31 20 31 34   , 14 Sep  2021 14
0060  3a 32 39 3a 34 35 20 47  4d 54 0d 0a 53 65 72 76   :29:45 G MT··Serv
0070  65 72 3a 20 41 70 61 63  68 65 2f 32 2e 34 2e 34   er: Apac he/2.4.4
0080  38 20 28 57 69 6e 36 34  29 20 4f 70 65 6e 53 53   8 (Win64 ) OpenSS
0090  4c 2f 31 2e 31 2e 31 6c  20 50 48 50 2f 37 2e 34   L/1.1.1l  PHP/7.4
00a0  2e 32 33 0d 0a 43 6f 6e  74 65 6e 74 2d 4c 65 6e   .23··Con tent-Len
00b0  67 74 68 3a 20 39 37 39  0d 0a 4b 65 65 70 2d 41   gth: 979 ··Keep-A
00c0  6c 69 76 65 3a 20 74 69  6d 65 6f 75 74 3d 35 2c   live: ti meout=5,
00d0  20 6d 61 78 3d 31 30 30  0d 0a 43 6f 6e 6e 65 63    max=100 ··Connec
00e0  74 69 6f 6e 3a 20 4b 65  65 70 2d 41 6c 69 76 65   tion: Ke ep-Alive
00f0  0d 0a 43 6f 6e 74 65 6e  74 2d 54 79 70 65 3a 20   ··Conten t-Type: 
0100  74 65 78 74 2f 68 74 6d  6c 3b 63 68 61 72 73 65   text/htm l;charse
0110  74 3d 55 54 46 2d 38 0d  0a 0d 0a 3c 21 44 4f 43   t=UTF-8· ···<!DOC
0120  54 59 50 45 20 48 54 4d  4c 20 50 55 42 4c 49 43   TYPE HTM L PUBLIC
0130  20 22 2d 2f 2f 57 33 43  2f 2f 44 54 44 20 48 54    "-//W3C //DTD HT
0140  4d 4c 20 33 2e 32 20 46  69 6e 61 6c 2f 2f 45 4e   ML 3.2 F inal//EN
0150  22 3e 0a 3c 68 74 6d 6c  3e 0a 20 3c 68 65 61 64   ">·<html >· <head
```

Decoding of hex to text

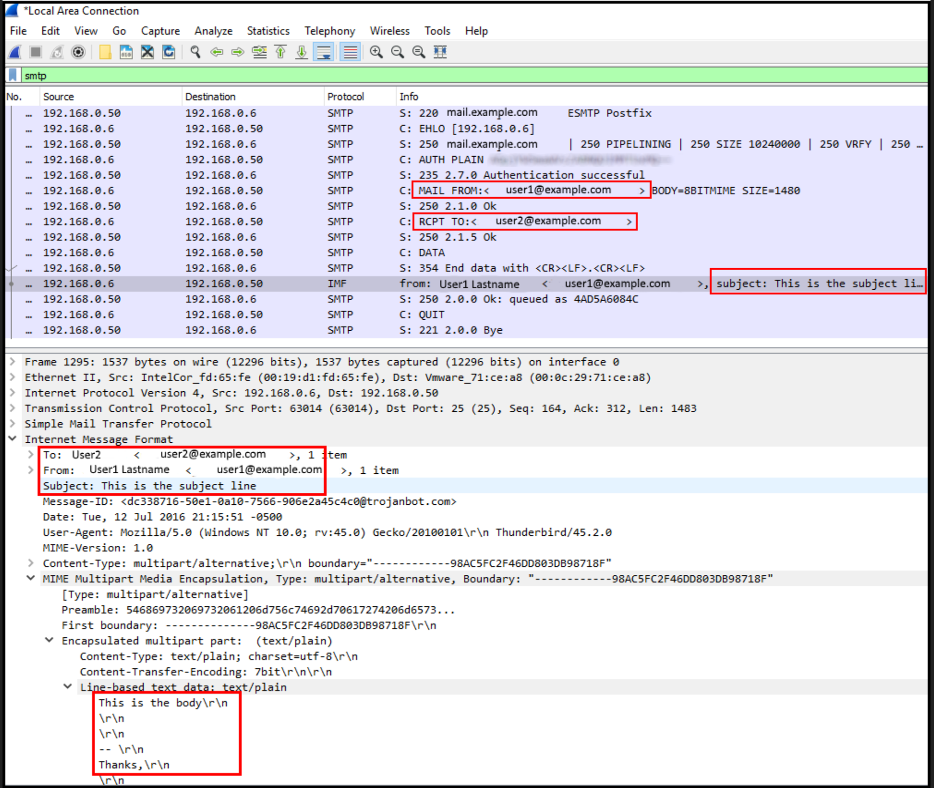
Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Wireshark – Useful filters to find things

smtp– Allows you to peek inside email communications that are not encrypted, see the messages that have been sent across a network.

ftp– See what files have been transferred between two locations. You can even extract passwords used to access FTP services!

icmp– Not all ICMP traffic is always useful, but it can reveal pings that a device can send to each other to check their statuses



smtp

```
No.  Source          Destination     Protocol  Info
 …   192.168.0.50    192.168.0.6     SMTP      S: 220  mail.example.com     ESMTP Postfix
 …   192.168.0.6     192.168.0.50    SMTP      C: EHLO [192.168.0.6]
 …   192.168.0.50    192.168.0.6     SMTP      S: 250  mail.example.com     | 250 PIPELINING | 250 SIZE 10240000 | 250 VRFY | 250 …
 …   192.168.0.6     192.168.0.50    SMTP      C: AUTH PLAIN [redacted]
 …   192.168.0.50    192.168.0.6     SMTP      S: 235 2.7.0 Authentication successful
 …   192.168.0.6     192.168.0.50    SMTP      C: MAIL FROM:<   user1@example.com   > BODY=8BITMIME SIZE=1480
 …   192.168.0.50    192.168.0.6     SMTP      S: 250 2.1.0 Ok
 …   192.168.0.6     192.168.0.50    SMTP      C: RCPT TO:<   user2@example.com   >
 …   192.168.0.50    192.168.0.6     SMTP      S: 250 2.1.5 Ok
 …   192.168.0.6     192.168.0.50    SMTP      C: DATA
 …   192.168.0.50    192.168.0.6     SMTP      S: 354 End data with <CR><LF>.<CR><LF>
 …   192.168.0.6     192.168.0.50    IMF       from: User1 Lastname   <   user1@example.com   >, subject: This is the subject li…
 …   192.168.0.50    192.168.0.6     SMTP      S: 250 2.0.0 Ok: queued as 4AD5A6084C
 …   192.168.0.6     192.168.0.50    SMTP      C: QUIT
 …   192.168.0.50    192.168.0.6     SMTP      S: 221 2.0.0 Bye
```

```
> Frame 1295: 1537 bytes on wire (12296 bits), 1537 bytes captured (12296 bits) on interface 0
> Ethernet II, Src: IntelCor_fd:65:fe (00:19:d1:fd:65:fe), Dst: Vmware_71:ce:a8 (00:0c:29:71:ce:a8)
> Internet Protocol Version 4, Src: 192.168.0.6, Dst: 192.168.0.50
> Transmission Control Protocol, Src Port: 63014 (63014), Dst Port: 25 (25), Seq: 164, Ack: 312, Len: 1483
> Simple Mail Transfer Protocol
v Internet Message Format
  > To:  User2      <     user2@example.com     >, 1 item
  > From:  User1 Lastname  <     user1@example.com     >, 1 item
    Subject: This is the subject line
    Message-ID: <dc338716-50e1-0a10-7566-906e2a45c4c0@trojanbot.com>
    Date: Tue, 12 Jul 2016 21:15:51 -0500
    User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:45.0) Gecko/20100101\r\n Thunderbird/45.2.0
    MIME-Version: 1.0
  > Content-Type: multipart/alternative;\r\n boundary="------------98AC5FC2F46DD803DB98718F"
  v MIME Multipart Media Encapsulation, Type: multipart/alternative, Boundary: "------------98AC5FC2F46DD803DB98718F"
      [Type: multipart/alternative]
      Preamble: 546869732069732061206d756c74692d70617274206d6573...
      First boundary: --------------98AC5FC2F46DD803DB98718F\r\n
    v Encapsulated multipart part:  (text/plain)
        Content-Type: text/plain; charset=utf-8\r\n
        Content-Transfer-Encoding: 7bit\r\n\r\n
      v Line-based text data: text/plain
          This is the body\r\n
          \r\n
          \r\n
          -- \r\n
          Thanks,\r\n
          \r\n
```

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

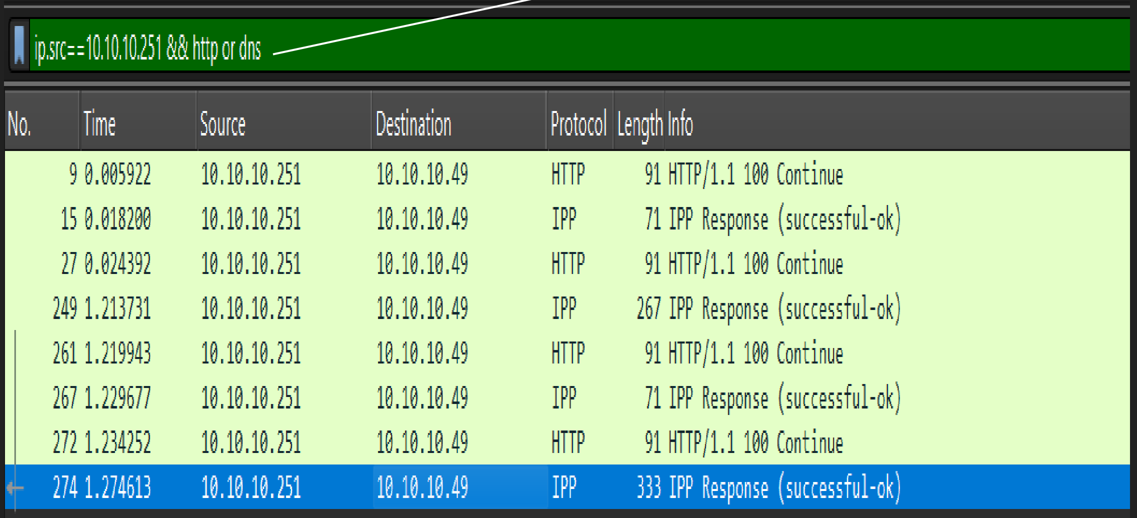
Wireshark – Useful filters to find things

http– Allows you to observe what web traffic is occurring and what has been sent. 'http.request' and 'http.response' can help you filter it down more.

ip.addr==x.x.x.x– See all communications from a set IP you can specify as 'ip.dst=' or 'ip.src=' to specify what a specific IP has sent or received.

Chaining queries – You can combine filters together using '&&', 'or' and '||'

"Traffic sent from IP 10.10.10.251 and that is HTTP or DNS."



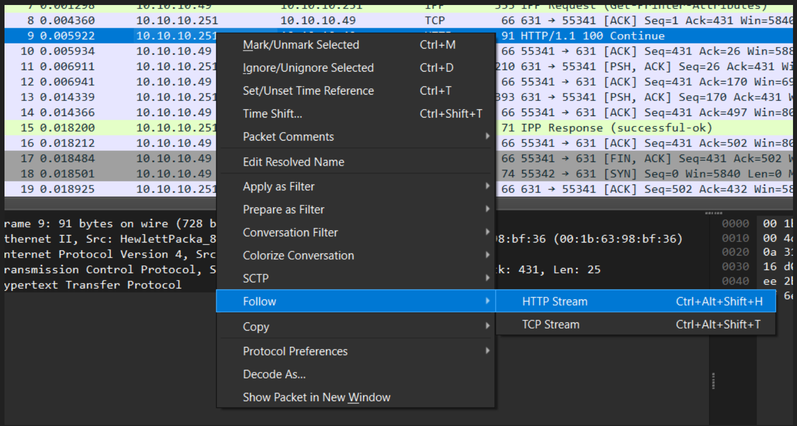
```
ip.src==10.10.10.251 && http or dns
```

No.	Time	Source	Destination	Protocol	Length	Info
9	0.005922	10.10.10.251	10.10.10.49	HTTP	91	HTTP/1.1 100 Continue
15	0.018200	10.10.10.251	10.10.10.49	IPP	71	IPP Response (successful-ok)
27	0.024392	10.10.10.251	10.10.10.49	HTTP	91	HTTP/1.1 100 Continue
249	1.213731	10.10.10.251	10.10.10.49	IPP	267	IPP Response (successful-ok)
261	1.219943	10.10.10.251	10.10.10.49	HTTP	91	HTTP/1.1 100 Continue
267	1.229677	10.10.10.251	10.10.10.49	IPP	71	IPP Response (successful-ok)
272	1.234252	10.10.10.251	10.10.10.49	HTTP	91	HTTP/1.1 100 Continue
274	1.274613	10.10.10.251	10.10.10.49	IPP	333	IPP Response (successful-ok)

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

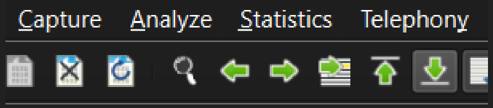
Using Wireshark effectively

- Linking events together in Wireshark can be annoying, the follow feature enables you to track down related packets to build up conversations!



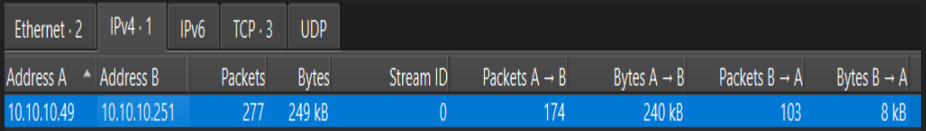
```
  7 0.001298   10.10.10.49    10.10.10.251   IPP   555 IPP Request (Get-Printer-Attributes)
  8 0.004360   10.10.10.251   10.10.10.49    TCP    66 631 → 55341 [ACK] Seq=1 Ack=431 Win=5840
  9 0.005922   10.10.10.251                         91 HTTP/1.1 100 Continue
 10 0.005934   10.10.10.49                          66 55341 → 631 [ACK] Seq=431 Ack=26 Win=588
 11 0.006911   10.10.10.251                        210 631 → 55341 [PSH, ACK] Seq=26 Ack=431 Wi
 12 0.006941   10.10.10.49                          66 55341 → 631 [ACK] Seq=431 Ack=170 Win=69
 13 0.014339   10.10.10.251                        393 631 → 55341 [PSH, ACK] Seq=170 Ack=431 W
 14 0.014366   10.10.10.49                          66 55341 → 631 [ACK] Seq=431 Ack=497 Win=80
 15 0.018200   10.10.10.251                         71 IPP Response (successful-ok)
 16 0.018212   10.10.10.49                          66 55341 → 631 [ACK] Seq=431 Ack=502 Win=80
 17 0.018484   10.10.10.49                          66 55341 → 631 [FIN, ACK] Seq=431 Ack=502 W
 18 0.018501   10.10.10.49                          74 55342 → 631 [SYN] Seq=0 Win=5840 Len=0 M
 19 0.018925   10.10.10.251                         66 631 → 55341 [ACK] Seq=502 Ack=432 Win=58

rame 9: 91 bytes on wire (728 b
thernet II, Src: HewlettPacka_8                   8:bf:36 (00:1b:63:98:bf:36)
nternet Protocol Version 4, Src
ransmission Control Protocol, S                   k: 431, Len: 25
ypertext Transfer Protocol
```

Mark/Unmark Selected Ctrl+M
Ignore/Unignore Selected Ctrl+D
Set/Unset Time Reference Ctrl+T
Time Shift... Ctrl+Shift+T
Packet Comments
Edit Resolved Name
Apply as Filter
Prepare as Filter
Conversation Filter
Colorize Conversation
SCTP
Follow
Copy
Protocol Preferences
Decode As...
Show Packet in New Window
HTTP Stream Ctrl+Alt+Shift+H
TCP Stream Ctrl+Alt+Shift+T



Capture
Analyze
Statistics
Telephony

Statistics tab is useful for telling you lots of things about the capture!



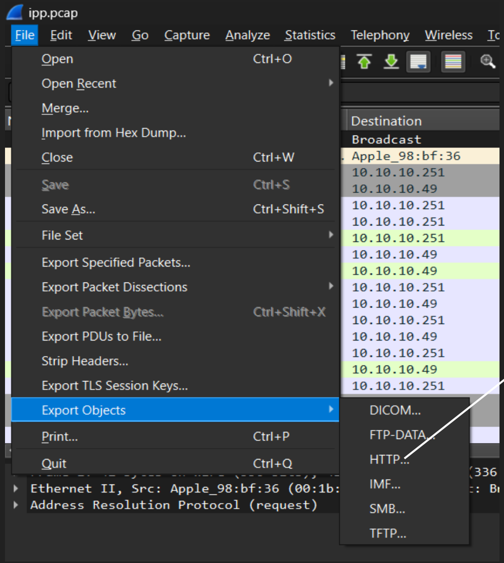
Address A	Address B	Packets	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A
10.10.10.49	10.10.10.251	277	249 kB	0	174	240 kB	103	8 kB

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Going up-streams

- Wireshark also captures file traffic between protocols; you can grab files sent across the network download them to your device (provided they are not sent over an encrypted channel).

This helps especially if it's something malicious were looking for (hint hint)



```
Destination
Broadcast
Apple_98:bf:36
10.10.10.251
10.10.10.49
10.10.10.251
10.10.10.251
10.10.10.251
10.10.10.49
10.10.10.49
10.10.10.251
10.10.10.49
10.10.10.251
10.10.10.49
10.10.10.251
10.10.10.49
10.10.10.251
```

```
(336
Ethernet II, Src: Apple_98:bf:36 (00:1b:          : Br
Address Resolution Protocol (request)
```

All the different types of traffic you can extract files from

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Useful linux commands + tips

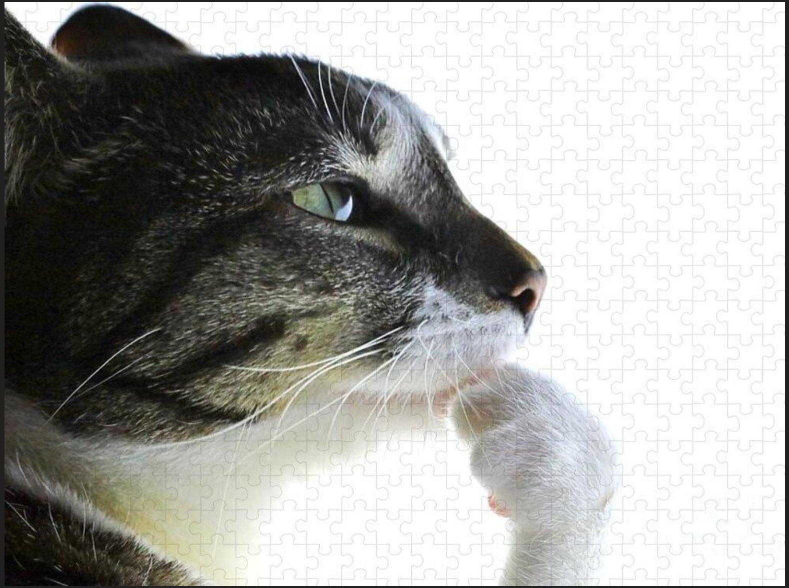
File – This command tells you information on what a file is and its attributes, helps you answer the question of “wtf is this”.

Grep – You can specify this command to search for something specific (or for a pattern using regex) and extract those values, good for log analysis.

Strings – Does a file have any lines of interest that can be pulled out?

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Questions?



Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Disclaimer

- I've given you a lot of information today but its not fully complete otherwise solving the challenges would be pointless…
- Some challenges today may throw curveballs; they may cover stuff from the Ethical hacking demonstration at the start of the year.
- If don't understand something, ask or google is your friend!

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.

Your turn!

https://gym.lilypadd.com/challenges

Challenges:

Funky Traffic Protocol

User Datagram Pilfering

POIP

Wait not MSFVenom?

Picture It

Export – https://bit.ly/4owMwuh+**.**

Do not repeat these activities outside of this session. Warwick Cyber Security Society takes no responsibility for actions performed with this knowledge.