



UNIVERSITY OF LIVERPOOL

CYBER SECURITY SOCIETY

Forensics: How to catch a heker

Disclaimer

Anything you learn in these sessions is FOR EDUCATIONAL PURPOSES ONLY and we are NOT RESPONSIBLE FOR YOUR ACTIONS! The tools we will show you aren't illegal but using them against a network you don't own or where you don't have the explicit written permission to use them is HIGHLY ILLEGAL and almost always against the terms of service.

DO NOT UNDER ANY CIRCUMSTANCES USE THE TOOLS AND TECHNIQUES SHOWN AGAINST ANY UNIVERSITY OWNED PRODUCT, WEBSITE OR NETWORK, YOU WILL BE PUNISHED BY THE DEPARTMENT/UNIVERSITY AND COULD BE PROSECUTED IN SOME CASES.

There are hundreds of websites where you can practice these techniques in a safe, legal environment without the risk of causing real damage or facing prosecution.



Digital Forensics

- Collecting and analysing information from computers
- Who does it
 - Law enforcement
 - Private investigators
- Main types
 - Networking (packets captured)
 - Disk (Entire contents of disk captured)
 - Memory (Memory captured)



Wireshark

- A open-source network forensics tool capable of capturing packets and reading packet capture (pcap) files
- Very useful for manually filtering and analysing packet captures

Sending a packet

- You will see 2 main protocols when looking at network traffic:
- TCP (20 Byte header size)
 - Slow
 - Stable
 - Reliable (Loss & Corruption checks)
 - Typical use: Requesting a web page, email
- UDP (8 Byte header size)
 - Fast
 - Doesn't have built in packet re-transmission
 - Packets can turn up out of order
 - Typical use: Streaming videos, music, playing games



TCP handshake

Starts of TCP streams, you can follow them.

No.	Time	Source	Destination	Protocol	Length	Info
681	18:32:04.615992	192.204.13.153	10.0.0.31	TCP	94	8801 → 11812 [PSH, ACK] Seq=348 Ack=801 Win=16 Len=40
684	18:32:04.637579	10.0.0.31	192.204.13.153	TCP	63	11812 → 8801 [PSH, ACK] Seq=801 Ack=388 Win=253 Len=9
689	18:32:04.683294	192.204.13.153	10.0.0.31	TCP	60	8801 → 11812 [ACK] Seq=388 Ack=810 Win=16 Len=0
769	18:32:05.375910	10.0.0.31	13.59.223.81	TCP	54	7130 → 443 [ACK] Seq=1 Ack=171 Win=253 Len=0
779	18:32:05.448247	13.59.223.81	10.0.0.31	TCP	56	443 → 7130 [ACK] Seq=171 Ack=86 Win=20 Len=0
818	18:32:05.867395	192.204.13.153	10.0.0.31	TCP	94	8801 → 11812 [PSH, ACK] Seq=388 Ack=810 Win=16 Len=40
822	18:32:05.897420	10.0.0.31	192.204.13.153	TCP	63	11812 → 8801 [PSH, ACK] Seq=810 Ack=428 Win=252 Len=9
827	18:32:05.939707	192.204.13.153	10.0.0.31	TCP	56	8801 → 11812 [ACK] Seq=428 Ack=819 Win=16 Len=0
856	18:32:06.176272	10.0.0.31	146.66.71.198	TCP	66	9556 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK
863	18:32:06.225084	146.66.71.198	10.0.0.31	TCP	66	80 → 9556 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460
864	18:32:06.225212	10.0.0.31	146.66.71.198	TCP	54	9556 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
865	18:32:06.225505	10.0.0.31	146.66.71.198	HTTP	388	GET / HTTP/1.1
871	18:32:06.272074	146.66.71.198	10.0.0.31	TCP	56	80 → 9556 [ACK] Seq=1 Ack=335 Win=30464 Len=0
873	18:32:06.279675	146.66.71.198	10.0.0.31	HTTP	739	HTTP/1.1 200 OK (text/html)
880	18:32:06.320004	10.0.0.31	146.66.71.198	TCP	54	9556 → 80 [ACK] Seq=335 Ack=686 Win=64768 Len=0
886	18:32:06.397447	10.0.0.31	146.66.71.198	HTTP	364	GET /images/netlab-labelled-pod1.jpg HTTP/1.1
889	18:32:06.449118	146.66.71.198	10.0.0.31	TCP	1514	80 → 9556 [ACK] Seq=686 Ack=645 Win=31488 Len=1460 [TCP se
890	18:32:06.449119	146.66.71.198	10.0.0.31	TCP	1514	80 → 9556 [ACK] Seq=2146 Ack=645 Win=31488 Len=1460 [TCP se



SYN

SYN-ACK

ACK



Following a stream

The screenshot shows the Wireshark interface with a packet list table. Packet 7 is selected, and a context menu is open over it, with 'Follow' > 'TCP Stream' highlighted. The packet list table is as follows:

No.	Time	Source	Destination	Protocol	Length	Stream index	Info
1	2017-08-20 15:32:26.278220	10.10.10.104	8.8.8.8	ICMP	98		Echo (ping) request id=0xb714, seq=63/16128,
2	2017-08-20 15:32:26.299738	8.8.8.8	10.10.10.104	ICMP	98		Echo (ping) reply id=0xb714, seq=63/16128,
3	2017-08-20 15:32:26.437101	10.10.10.104	10.10.10.127	DNS	77		Standard query 0x7ad3 A didierstevens.com
4	2017-08-20 15:32:26.437236	10.10.10.104	10.10.10.127	DNS	77		Standard query 0xab7 AAAA didierstevens.com
5	2017-08-20 15:32:26.529210	10.10.10.127	10.10.10.104	DNS	93		Standard query response 0x7ad3 A didierstevens.com
6	2017-08-20 15:32:26.547802	10.10.10.127	10.10.10.104	DNS	133		Standard query response 0xab7 AAAA didierstevens.com
7	2017-08-20 15:32:26.548237	10.10.10.104	96.126.103.196	TCP	78	0	53261 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460
8	2017-08-20 15:32:26.687602	10.10.10.104	96.126.103.196	TCP	78		→ 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460
9	2017-08-20 15:32:26.694441	96.126.103.196	10.10.10.104	TCP	78		3261 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0
10	2017-08-20 15:32:26.694535	10.10.10.104	96.126.103.196	TCP	78		Set/Unset Time Reference
11	2017-08-20 15:32:26.694738	10.10.10.104	96.126.103.196	HTTP	120		Time Shift...
12	2017-08-20 15:32:26.833684	96.126.103.196	10.10.10.104	TCP	78		Packet Comment...
13	2017-08-20 15:32:26.833768	96.126.103.196	10.10.10.104	TCP	78		3262 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0
14	2017-08-20 15:32:26.841540	96.126.103.196	10.10.10.104	TCP	78		→ 80 [ACK] Seq=1 Ack=1 Win=132480 Len=0
15	2017-08-20 15:32:26.845071	96.126.103.196	10.10.10.104	TCP	78		3261 [ACK] Seq=1 Ack=392 Win=30000 Len=0
16	2017-08-20 15:32:26.845076	96.126.103.196	10.10.10.104	TCP	78		3261 [ACK] Seq=1 Ack=392 Win=30000 Len=1
17	2017-08-20 15:32:26.845154	10.10.10.104	96.126.103.196	TCP	78		→ 80 [ACK] Seq=392 Ack=2881 Win=129600 Len=0
18	2017-08-20 15:32:26.845942	96.126.103.196	10.10.10.104	HTTP	120		1 200 OK (text/html)

The screenshot shows the 'Follow SIP Stream' dialog box in Wireshark. The details of the selected INVITE message are displayed as follows:

```
INVITE sip:test@10.0.2.15:5060 SIP/2.0
Via: SIP/2.0/UDP 10.0.2.20:5060;branch=z9hG4bK-2134-1-0
From: "G726-16/8000" <sip:sipp@10.0.2.20:5060>;tag=1
To: test <sip:test@10.0.2.15:5060>
Call-ID: 1-2134@10.0.2.20
CSeq: 1 INVITE
Contact: sip:sipp@10.0.2.20:5060
Max-Forwards: 70
Content-Type: application/sdp
Content-Length: 128

v=0
o=- 42 42 IN IP4 10.0.2.20
s=-
c=IN IP4 10.0.2.20
t=0
m=audio 6000 RTP/AVP 99
a=rtpmap:99 G726-16/8000
a=recvnly
SIP/2.0 100 Trying
Via: SIP/2.0/UDP 10.0.2.20:5060;branch=z9hG4bK-2134-1-0
From: "G726-16/8000" <sip:sipp@10.0.2.20:5060>;tag=1
To: test <sip:test@10.0.2.15:5060>
Call-ID: 1-2134@10.0.2.20
CSeq: 1 INVITE
User-Agent: FreeSWITCH-mod_sofia/1.6.12-20-b91a0a6-64bit
Content-Length: 0
```

FTP

- Will also come across FTP
- Communication is in plain text INCLUDING PASSWORD– MITMA
- Follow stream same as TCP
- To get data, save data as raw

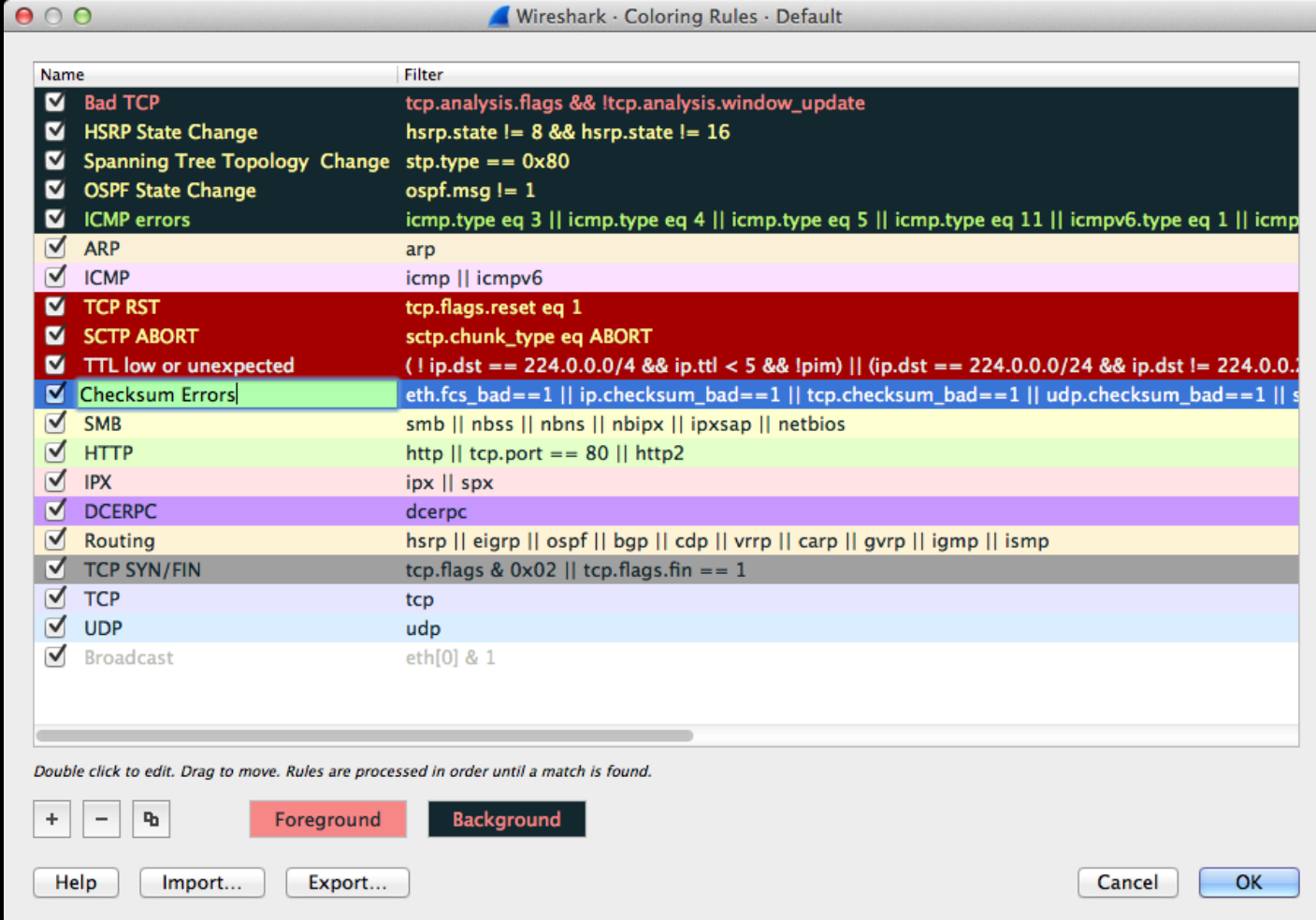
HTTP Status Codes

HTTP STATUS CODES	
2xx Success	
200	Success / OK
3xx Redirection	
301	Permanent Redirect
302	Temporary Redirect
304	Not Modified
4xx Client Error	
401	Unauthorized Error
403	Forbidden
404	Not Found
405	Method Not Allowed
5xx Server Error	
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout

INFODIGIT



Colouring Rules



The image shows the 'Wireshark - Coloring Rules - Default' dialog box. It contains a list of rules with checkboxes and filters. The rules are color-coded: Bad TCP (dark red), HSRP State Change (yellow), Spanning Tree Topology Change (yellow), OSPF State Change (yellow), ICMP errors (green), ARP (yellow), ICMP (yellow), TCP RST (red), SCTP ABORT (red), TTL low or unexpected (red), Checksum Errors (green), SMB (light green), HTTP (light green), IPX (light pink), DCERPC (purple), Routing (yellow), TCP SYN/FIN (grey), TCP (light blue), UDP (light blue), and Broadcast (light blue). The 'Checksum Errors' rule is currently selected and highlighted in blue.

Name	Filter
<input checked="" type="checkbox"/> Bad TCP	tcp.analysis.flags && !tcp.analysis.window_update
<input checked="" type="checkbox"/> HSRP State Change	hsrp.state != 8 && hsrp.state != 16
<input checked="" type="checkbox"/> Spanning Tree Topology Change	stp.type == 0x80
<input checked="" type="checkbox"/> OSPF State Change	ospf.msg != 1
<input checked="" type="checkbox"/> ICMP errors	icmp.type eq 3 icmp.type eq 4 icmp.type eq 5 icmp.type eq 11 icmpv6.type eq 1 icmpv6.type eq 2
<input checked="" type="checkbox"/> ARP	arp
<input checked="" type="checkbox"/> ICMP	icmp icmpv6
<input checked="" type="checkbox"/> TCP RST	tcp.flags.reset eq 1
<input checked="" type="checkbox"/> SCTP ABORT	sctp.chunk_type eq ABORT
<input checked="" type="checkbox"/> TTL low or unexpected	(! ip.dst == 224.0.0.0/4 && ip.ttl < 5 && !pim) (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.1)
<input checked="" type="checkbox"/> Checksum Errors	eth.fcs_bad==1 ip.checksum_bad==1 tcp.checksum_bad==1 udp.checksum_bad==1 sctp.checksum_bad==1
<input checked="" type="checkbox"/> SMB	smb nbss nbns nbpx ipxsap netbios
<input checked="" type="checkbox"/> HTTP	http tcp.port == 80 http2
<input checked="" type="checkbox"/> IPX	ipx spx
<input checked="" type="checkbox"/> DCERPC	dcerpc
<input checked="" type="checkbox"/> Routing	hsrp eigrp ospf bgp cdp vrrp carp gvrp igmp ismp
<input checked="" type="checkbox"/> TCP SYN/FIN	tcp.flags & 0x02 tcp.flags.fin == 1
<input checked="" type="checkbox"/> TCP	tcp
<input checked="" type="checkbox"/> UDP	udp
<input checked="" type="checkbox"/> Broadcast	eth[0] & 1

Double click to edit. Drag to move. Rules are processed in order until a match is found.

+ - [icon] Foreground Background

Help Import... Export... Cancel OK

Filters

The screenshot shows the Wireshark interface with the following details:

- File: traffic-for-wireshark-column-setup.pcap
- Filter: dns and ip.addr != 192.168.10.1
- Packet List Table:

Time	Src	Port	Dst	Port
2018-08-03 19:06:20	192.168.10.195	62006	192.168.10.1	53
2018-08-03 19:06:20	192.168.10.1	53	192.168.10.195	62006
- Packet Details:
 - Frame 1: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
 - Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netge
 - Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.168.10.1
- Packet Bytes:

```
0000  20 e5 2a b6 93 f1 00 08 02 1c 47 ae 08 00 45 00  .*. . . . .
0010  00 42 77 31 00 00 80 11 2d 65 c0 a8 0a c3 c0 a8  .Bw1. . . . .
0020  0a 01 f2 36 00 35 00 2e ae 31 df 27 01 00 00 01  ...6.5. . . . .
```
- Status: Packets: 4448 · Displayed: 99 (2.2%) | Profile: Default

- Can filter through the packets
- Simple filters like 'tcp' or 'dns' will display those types of packets
- More advanced filters like ip.addr != 192.168.10.1
- Can also do things like ==, >, <, ...
- More here:
<https://www.wireshark.org/docs/dfref/>

Exporting a HTTP Object

The screenshot shows the Wireshark interface with the 'Export Objects' menu open. The 'HTTP...' option is selected, which has opened the 'Export - HTTP object list' dialog box. The dialog box contains a table of HTTP objects with columns for Packet No., Hostname, Content Type, Size, and Filename. The first row is selected, and the 'Save' button is highlighted.

Packet No.	Hostname	Content Type	Size	Filename
185	www.paypalaccountsloginn.myddns.com	text/html	15 kB	websrc
389	www.paypalaccountsloginn.myddns.com	application/x-www...	175 bytes	verifychal
409	www.paypalaccountsloginn.myddns.com	text/html	337 bytes	verifychal
627	www.paypalaccountsloginn.myddns.com	application/x-www...	1,205 bytes	webscr?c
636	www.paypalaccountsloginn.myddns.com	text/html	2,146 bytes	webscr?c
640	www.paypalaccountsloginn.myddns.com	text/html	338 bytes	pp_favico
644	www.paypalaccountsloginn.myddns.com	text/html	328 bytes	favi-on.i
648	www.paypalaccountsloginn.myddns.com	text/html	261 bytes	my



Disk image forensics tools

- Image a disk
 - Ftkimager (has free version)
 - safeback
 - EnCase
 - dd

```
dd if=/dev/sda of=sda.img
```

- Inspect an image
 - Sleuthkit/autopsy



Autopsy Forensic Browser x +

localhost:9999/autopsy

Kali Linux Kali Training Kali Tools Kali Forums Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB

WARNING: Your browser currently has Java Script enabled.

You do not need Java Script to use Autopsy and it is recommended that it be turned off for security reasons.

Autopsy Forensic Browser 2.24



<http://www.steuthkit.org/autopsy/>

OPEN CASE NEW CASE HELP

1. **Case Name:** The name of this investigation. It can contain only letters, numbers, and symbols.

2. **Description:** An optional, one line description of this case.

3. **Investigator Names:** The optional names (with no spaces) of the investigators for this case.

a.

b.

c.

d.

e.

f.

g.

h.

i.

j.

NEW CASE

CANCEL

HELP



ADD A NEW HOST

1. **Host Name:** The name of the computer being investigated. It can contain only letters, numbers, and symbols.

2. **Description:** An optional one-line description or note about this computer.

3. **Time zone:** An optional timezone value (i.e. EST5EDT). If not given, it defaults to the local setting. A list of time zones can be found in the help files.

4. **Timeskew Adjustment:** An optional value to describe how many seconds this computer's clock was out of sync. For example, if the computer was 10 seconds fast, then enter -10 to compensate.

5. **Path of Alert Hash Database:** An optional hash database of known bad files.

6. **Path of Ignore Hash Database:** An optional hash database of known good files.



ADD A NEW IMAGE

1. Location

Enter the full path (starting with /) to the image file.
If the image is split (either raw or EnCase), then enter '*' for the extension.

2. Type

Please select if this image file is for a disk or a single partition.

Disk Partition

3. Import Method

To analyze the image file, it must be located in the evidence locker. It can be imported from its current location using a symbolic link, by copying it, or by moving it. Note that if a system failure occurs during the move, then the image could become corrupt.

Symlink Copy Move

NEXT

CANCEL

HELP



Image File Details

Local Name: images/image.E01

File System Details

Analysis of the image file shows the following partitions:

Partition 1 (Type: ntfs)

Mount Point:

File System Type:

ADD

CANCEL

HELP



Select a volume to analyze or add a new image file.

CASE GALLERY

HOST GALLERY

HOST MANAGER

mount	name	fs type	
<input checked="" type="radio"/> C:/	image.E01-0-0	ntfs	details

ANALYZE

ADD IMAGE FILE

CLOSE HOST

HELP

FILE ACTIVITY TIME LINES

IMAGE INTEGRITY

HASH DATABASES

VIEW NOTES

EVENT SEQUENCER



General File System Details

FILE SYSTEM INFORMATION

File System Type: NTFS
Volume Serial Number: BC3E56683E561C28
OEM Name: NTFS
Version: Windows XP

METADATA INFORMATION

First Cluster of MFT: 786432
First Cluster of MFT Mirror: 2
Size of MFT Entries: 1024 bytes
Size of Index Records: 4096 bytes
Range: 0 - 256
Root Directory: 5

CONTENT INFORMATION

Sector Size: 512
Cluster Size: 4096
Total Cluster Range: 0 - 15022074
Total Sector Range: 0 - 120176606

\$AttrDef Attribute Values:
\$STANDARD_INFORMATION (16) Size: 48-72 Flags: Resident
\$ATTRIBUTE_LIST (32) Size: No Limit Flags: Non-resident
\$FILE_NAME (48) Size: 68-578 Flags: Resident,Index
\$OBJECT_ID (64) Size: 0-256 Flags: Resident
\$SECURITY_DESCRIPTOR (80) Size: No Limit Flags: Non-resident
\$VOLUME_NAME (96) Size: 2-256 Flags: Resident
\$VOLUME_INFORMATION (112) Size: 12-12 Flags: Resident
\$DATA (128) Size: No Limit Flags:
\$INDEX_ROOT (144) Size: No Limit Flags: Resident
\$INDEX_ALLOCATION (160) Size: No Limit Flags: Non-resident
\$BITMAP (176) Size: No Limit Flags: Non-resident
\$REPARSE_POINT (192) Size: 0-16384 Flags: Non-resident
\$EA_INFORMATION (208) Size: 8-8 Flags: Resident
\$EA (224) Size: 0-65536 Flags:
\$LOGGED_UTILITY_STREAM (256) Size: 0-65536 Flags: Non-resident



Directory Seek

Enter the name of a directory that you want to view.
 C:/

VIEW

File Name Search

Enter a Perl regular expression for the file names you want to find.

SEARCH

ALL DELETED FILES

EXPAND DIRECTORIES

Current Directory: [C:/](#)

ADD NOTE

GENERATE MD5 LIST OF FILES

DEL	Type	NAME 	WRITTEN	ACCESSED	CHANGED	CREATED	SIZE	UID	GID	META
Error Parsing File (Invalid Characters?): V/V 256: \$OrphanFiles 0000-00-00 00:00:00 (UTC) 0000-00-00 00:00:00 (UTC) 0000-00-00 00:00:00 (UTC) 0000-00-00 00:00:00 (UTC) 0 0 0										
	r / r	\$AttrDef	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2560	0	0	4-128-1
	r / r	\$BadClus	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	0	0	0	8-128-2
	r / r	\$BadClus:\$Bad	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	61530419200	0	0	8-128-1
	r / r	\$Bitmap	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	1877760	0	0	6-128-4
	r / r	\$Boot	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	8192	48	0	7-128-1
	d / d	\$Extend/	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	2021-11-08 20:22:20 (GMT)	656	0	0	11-144-4

File Browsing Mode

In this mode, you can view file and directory contents.

File contents will be shown in this window.
 More file details can be found using the Metadata link at the end of the list (on the right).
 You can also sort the files using the column headers



Memory forensics tools

- Imaging
 - FTK imager
 - MDD (Memory DD)
 - hiberfil.sys
- Analysis
 - Volatility

Challenges

- [ctf.cybersoc.cf](#): Forensics
- Tryhackme
 - volatility
 - linuxserverforensics
 - Overpass 2: hacked
 - autopsy