pcapFS
Mounting Network Data for On-the-Fly Analysis

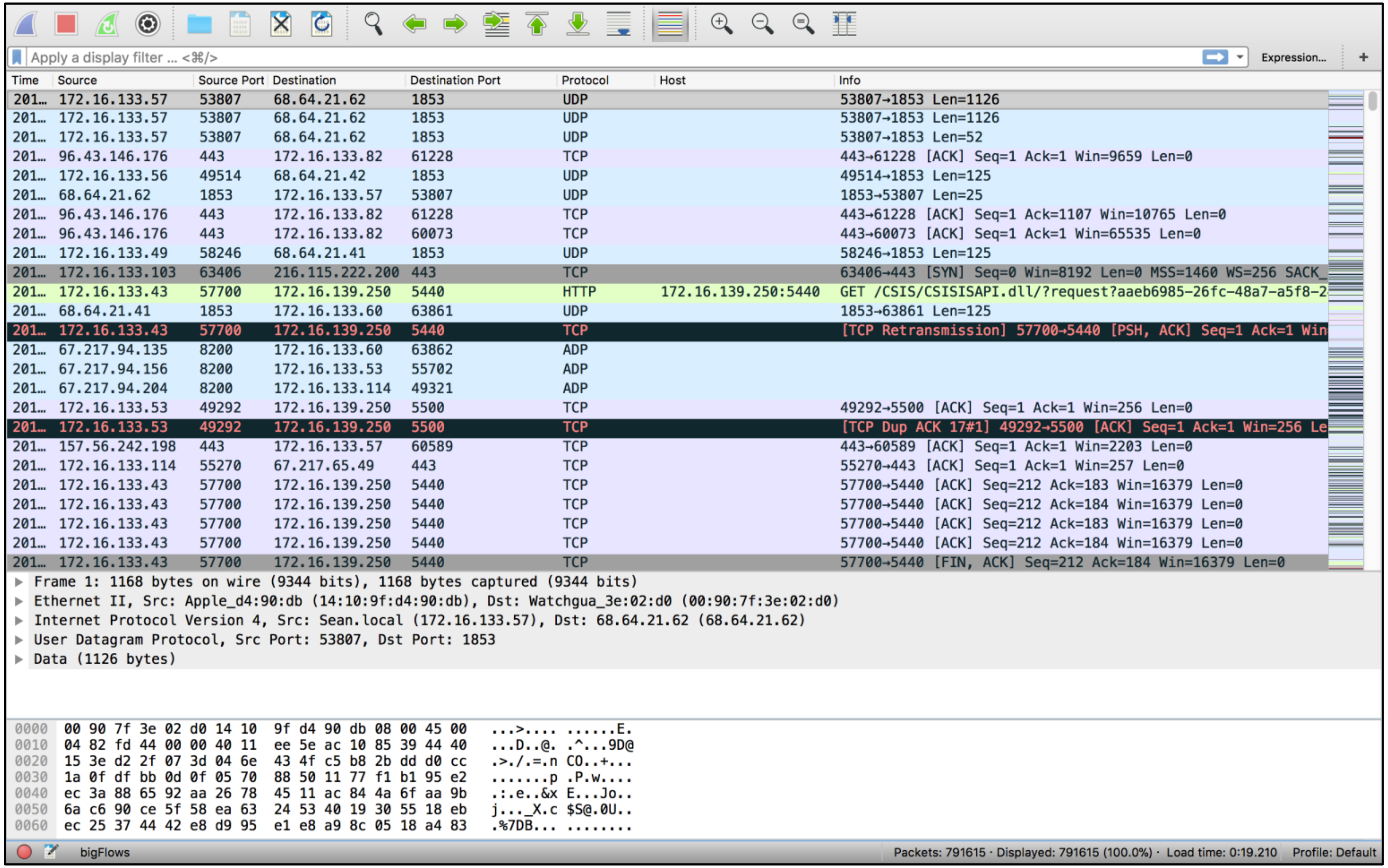
Fraunhofer Institute for Communication, Information Processing and Ergonomics

OSDFCon 2018
October 17th

Jan-Niclas Hilgert* jan-niclas.hilgert@fkie.fraunhofer.de
Martin Lambertz martin.lambertz@fkie.fraunhofer.de

State-of-the-art network forensics

- Wireshark is great, but



Time	Source	Source Port	Destination	Destination Port	Protocol	Host	Info
201…	172.16.133.57	53807	68.64.21.62	1853	UDP		53807→1853 Len=1126
201…	172.16.133.57	53807	68.64.21.62	1853	UDP		53807→1853 Len=1126
201…	172.16.133.57	53807	68.64.21.62	1853	UDP		53807→1853 Len=52
201…	96.43.146.176	443	172.16.133.82	61228	TCP		443→61228 [ACK] Seq=1 Ack=1 Win=9659 Len=0
201…	172.16.133.56	49514	68.64.21.42	1853	UDP		49514→1853 Len=125
201…	68.64.21.62	1853	172.16.133.57	53807	UDP		1853→53807 Len=25
201…	96.43.146.176	443	172.16.133.82	61228	TCP		443→61228 [ACK] Seq=1 Ack=1107 Win=10765 Len=0
201…	96.43.146.176	443	172.16.133.82	60073	TCP		443→60073 [ACK] Seq=1 Ack=1 Win=65535 Len=0
201…	172.16.133.49	58246	68.64.21.41	1853	UDP		58246→1853 Len=125
201…	172.16.133.103	63406	216.115.222.200	443	TCP		63406→443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_
201…	172.16.133.43	57700	172.16.139.250	5440	HTTP	172.16.139.250:5440	GET /CSIS/CSISISAPI.dll/?request?aaeb6985-26fc-48a7-a5f8-2
201…	68.64.21.41	1853	172.16.133.60	63861	UDP		1853→63861 Len=125
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		[TCP Retransmission] 57700→5440 [PSH, ACK] Seq=1 Ack=1 Win
201…	67.217.94.135	8200	172.16.133.60	63862	ADP		
201…	67.217.94.156	8200	172.16.133.53	55702	ADP		
201…	67.217.94.204	8200	172.16.133.114	49321	ADP		
201…	172.16.133.53	49292	172.16.139.250	5500	TCP		49292→5500 [ACK] Seq=1 Ack=1 Win=256 Len=0
201…	172.16.133.53	49292	172.16.139.250	5500	TCP		[TCP Dup ACK 17#1] 49292→5500 [ACK] Seq=1 Ack=1 Win=256 Le
201…	157.56.242.198	443	172.16.133.57	60589	TCP		443→60589 [ACK] Seq=1 Ack=1 Win=2203 Len=0
201…	172.16.133.114	55270	67.217.65.49	443	TCP		55270→443 [ACK] Seq=1 Ack=1 Win=257 Len=0
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		57700→5440 [ACK] Seq=212 Ack=183 Win=16379 Len=0
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		57700→5440 [ACK] Seq=212 Ack=184 Win=16379 Len=0
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		57700→5440 [ACK] Seq=212 Ack=183 Win=16379 Len=0
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		57700→5440 [ACK] Seq=212 Ack=184 Win=16379 Len=0
201…	172.16.133.43	57700	172.16.139.250	5440	TCP		57700→5440 [FIN, ACK] Seq=212 Ack=184 Win=16379 Len=0

```
▶ Frame 1: 1168 bytes on wire (9344 bits), 1168 bytes captured (9344 bits)
▶ Ethernet II, Src: Apple_d4:90:db (14:10:9f:d4:90:db), Dst: Watchgua_3e:02:d0 (00:90:7f:3e:02:d0)
▶ Internet Protocol Version 4, Src: Sean.local (172.16.133.57), Dst: 68.64.21.62 (68.64.21.62)
▶ User Datagram Protocol, Src Port: 53807, Dst Port: 1853
▶ Data (1126 bytes)
```

```
0000  00 90 7f 3e 02 d0 14 10  9f d4 90 db 08 00 45 00   ...>.... ......E.
0010  04 82 fd 44 00 00 40 11  ee 5e ac 10 85 39 44 40   ...D..@. .^...9D@
0020  15 3e d2 2f 07 3d 04 6e  43 4f c5 b8 2b dd d0 cc   .>./.=.n CO..+...
0030  1a 0f df bb 0d 0f 05 70  88 50 11 77 f1 b1 95 e2   .......p .P.w....
0040  ec 3a 88 65 92 aa 26 78  45 11 ac 84 4a 6f aa 9b   .:.e..&x E...Jo..
0050  6a c6 90 ce 5f 58 ea 63  24 53 40 19 30 55 18 eb   j..._X.c $S@.0U..
0060  ec 25 37 44 42 e8 d9 95  e1 e8 a9 8c 05 18 a4 83   .%7DB... ........
```

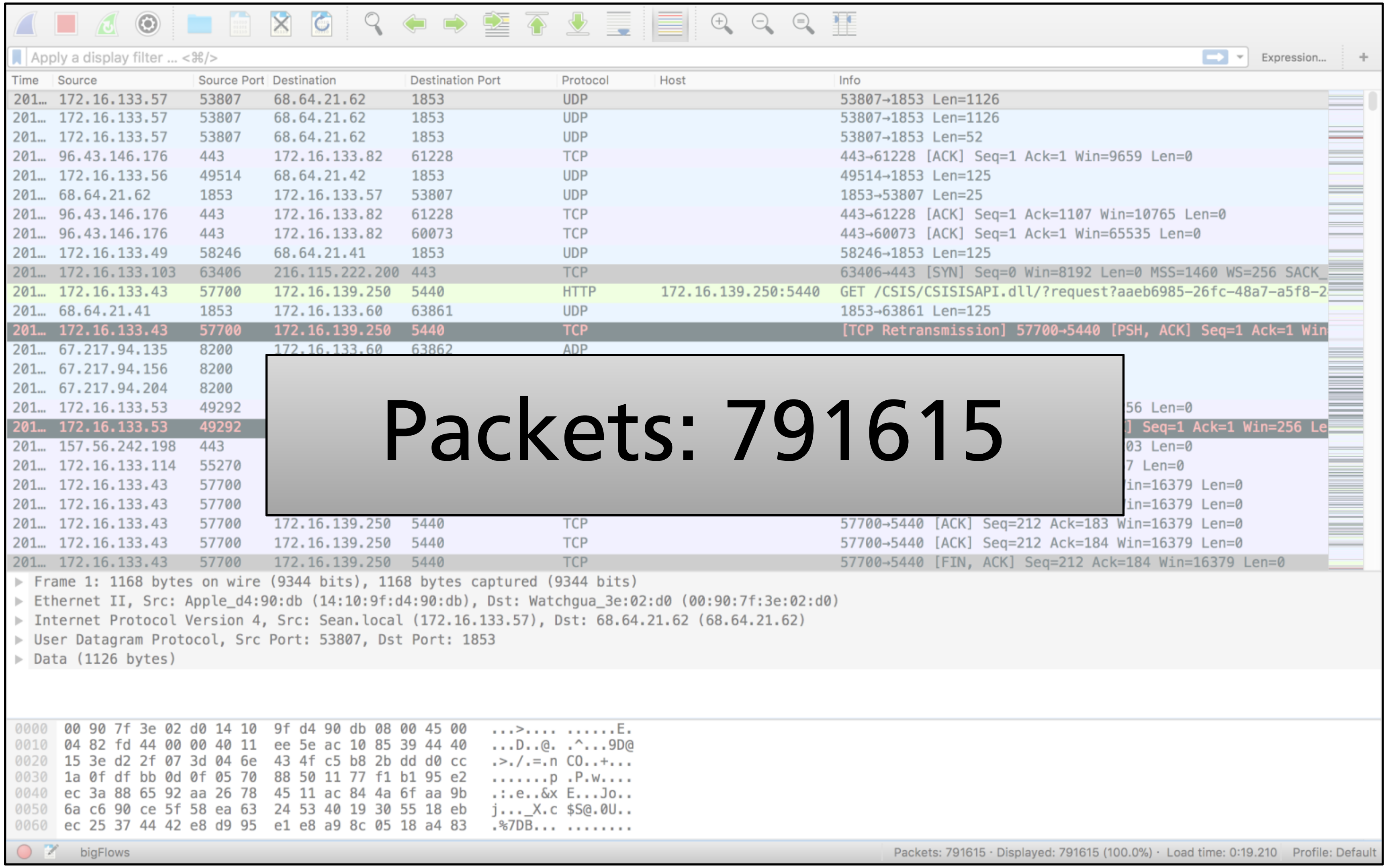
bigFlows

Packets: 791615 · Displayed: 791615 (100.0%) · Load time: 0:19.210   Profile: Default

State-of-the-art network forensics

- Wireshark is great, but

1. Usability



```
Time Source          Source Port Destination     Destination Port Protocol Host                 Info
201… 172.16.133.57   53807       68.64.21.62     1853             UDP                           53807→1853 Len=1126
201… 172.16.133.57   53807       68.64.21.62     1853             UDP                           53807→1853 Len=1126
201… 172.16.133.57   53807       68.64.21.62     1853             UDP                           53807→1853 Len=52
201… 96.43.146.176   443         172.16.133.82   61228            TCP                           443→61228 [ACK] Seq=1 Ack=1 Win=9659 Len=0
201… 172.16.133.56   49514       68.64.21.42     1853             UDP                           49514→1853 Len=125
201… 68.64.21.62     1853        172.16.133.57   53807            UDP                           1853→53807 Len=25
201… 96.43.146.176   443         172.16.133.82   61228            TCP                           443→61228 [ACK] Seq=1 Ack=1107 Win=10765 Len=0
201… 96.43.146.176   443         172.16.133.82   60073            TCP                           443→60073 [ACK] Seq=1 Ack=1 Win=65535 Len=0
201… 172.16.133.49   58246       68.64.21.41     1853             UDP                           58246→1853 Len=125
201… 172.16.133.103  63406       216.115.222.200 443              TCP                           63406→443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_
201… 172.16.133.43   57700       172.16.139.250  5440             HTTP     172.16.139.250:5440  GET /CSIS/CSISISAPI.dll/?request?aaeb6985-26fc-48a7-a5f8-2
201… 68.64.21.41     1853        172.16.133.60   63861            UDP                           1853→63861 Len=125
201… 172.16.133.43   57700       172.16.139.250  5440             TCP                           [TCP Retransmission] 57700→5440 [PSH, ACK] Seq=1 Ack=1 Win
201… 67.217.94.135   8200        172.16.133.60   63862            ADP
201… 67.217.94.156   8200
201… 67.217.94.204   8200
201… 172.16.133.53   49292                                                                      56 Len=0
201… 172.16.133.53   49292                                                                      ] Seq=1 Ack=1 Win=256 Le
201… 157.56.242.198  443                                                                        03 Len=0
201… 172.16.133.114  55270                                                                      7 Len=0
201… 172.16.133.43   57700                                                                      in=16379 Len=0
201… 172.16.133.43   57700                                                                      in=16379 Len=0
201… 172.16.133.43   57700       172.16.139.250  5440             TCP                           57700→5440 [ACK] Seq=212 Ack=183 Win=16379 Len=0
201… 172.16.133.43   57700       172.16.139.250  5440             TCP                           57700→5440 [ACK] Seq=212 Ack=184 Win=16379 Len=0
201… 172.16.133.43   57700       172.16.139.250  5440             TCP                           57700→5440 [FIN, ACK] Seq=212 Ack=184 Win=16379 Len=0
```

Packets: 791615

```
▶ Frame 1: 1168 bytes on wire (9344 bits), 1168 bytes captured (9344 bits)
▶ Ethernet II, Src: Apple_d4:90:db (14:10:9f:d4:90:db), Dst: Watchgua_3e:02:d0 (00:90:7f:3e:02:d0)
▶ Internet Protocol Version 4, Src: Sean.local (172.16.133.57), Dst: 68.64.21.62 (68.64.21.62)
▶ User Datagram Protocol, Src Port: 53807, Dst Port: 1853
▶ Data (1126 bytes)
```

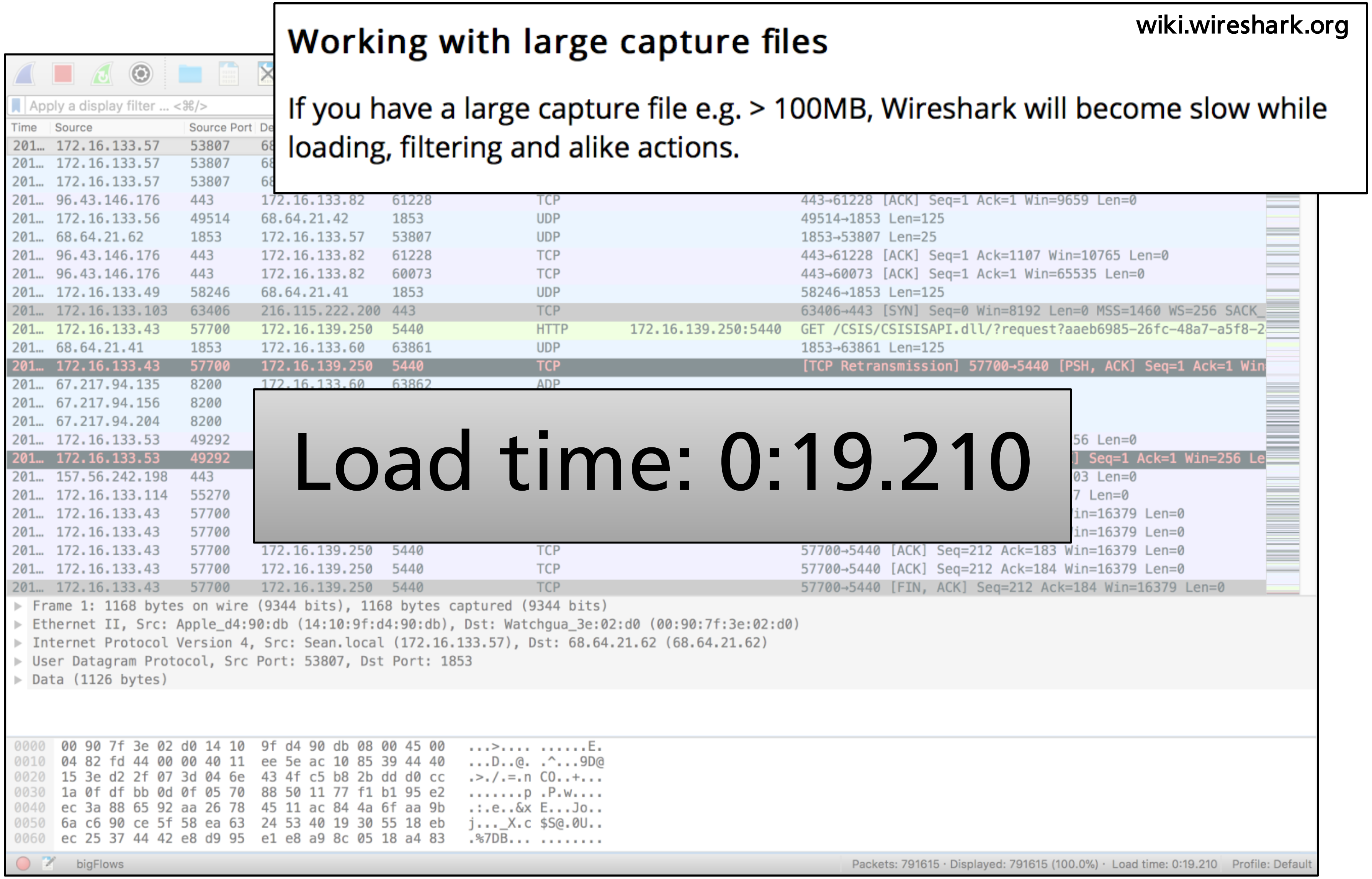
```
0000  00 90 7f 3e 02 d0 14 10  9f d4 90 db 08 00 45 00   ...>.... ......E.
0010  04 82 fd 44 00 00 40 11  ee 5e ac 10 85 39 44 40   ...D..@. .^...9D@
0020  15 3e d2 2f 07 3d 04 6e  43 4f c5 b8 2b dd d0 cc   .>./.=.n CO..+...
0030  1a 0f df bb 0d 0f 05 70  88 50 11 77 f1 b1 95 e2   .......p .P.w....
0040  ec 3a 88 65 92 aa 26 78  45 11 ac 84 4a 6f aa 9b   .:.e..&x E...Jo..
0050  6a c6 90 ce 5f 58 ea 63  24 53 40 19 30 55 18 eb   j..._X.c $S@.0U..
0060  ec 25 37 44 42 e8 d9 95  e1 e8 a9 8c 05 18 a4 83   .%7DB... ........
```

bigFlows   Packets: 791615 · Displayed: 791615 (100.0%) · Load time: 0:19.210   Profile: Default

State-of-the-art network forensics

- Wireshark is great, but

1. Usability
2. Performance



wiki.wireshark.org

Working with large capture files

If you have a large capture file e.g. > 100MB, Wireshark will become slow while loading, filtering and alike actions.

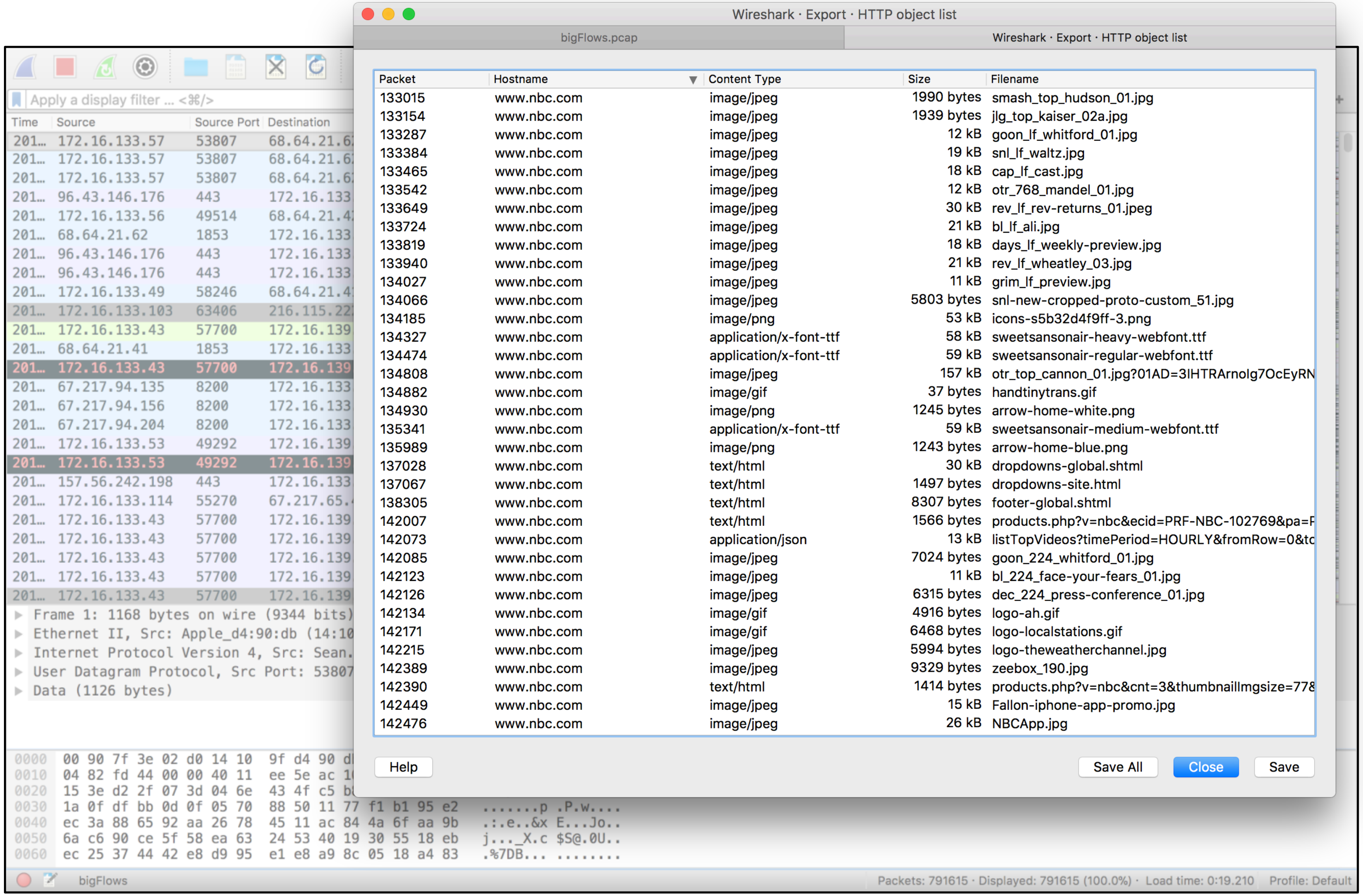
Load time: 0:19.210

```
Apply a display filter ... <⌘/>
Time  Source           Source Port De
201…  172.16.133.57    53807       68
201…  172.16.133.57    53807       68
201…  172.16.133.57    53807       68
201…  96.43.146.176    443         172.16.133.82    61228    TCP                           443→61228 [ACK] Seq=1 Ack=1 Win=9659 Len=0
201…  172.16.133.56    49514       68.64.21.42      1853     UDP                           49514→1853 Len=125
201…  68.64.21.62      1853        172.16.133.57    53807    UDP                           1853→53807 Len=25
201…  96.43.146.176    443         172.16.133.82    61228    TCP                           443→61228 [ACK] Seq=1 Ack=1107 Win=10765 Len=0
201…  96.43.146.176    443         172.16.133.82    60073    TCP                           443→60073 [ACK] Seq=1 Ack=1 Win=65535 Len=0
201…  172.16.133.49    58246       68.64.21.41      1853     UDP                           58246→1853 Len=125
201…  172.16.133.103   63406       216.115.222.200  443      TCP                           63406→443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_
201…  172.16.133.43    57700       172.16.139.250   5440     HTTP   172.16.139.250:5440    GET /CSIS/CSISISAPI.dll/?request?aaeb6985-26fc-48a7-a5f8-2
201…  68.64.21.41      1853        172.16.133.60    63861    UDP                           1853→63861 Len=125
201…  172.16.133.43    57700       172.16.139.250   5440     TCP                           [TCP Retransmission] 57700→5440 [PSH, ACK] Seq=1 Ack=1 Win
201…  67.217.94.135    8200        172.16.133.60    63862    ADP
201…  67.217.94.156    8200
201…  67.217.94.204    8200
201…  172.16.133.53    49292                                                                                     56 Len=0
201…  172.16.133.53    49292                                                                                     ] Seq=1 Ack=1 Win=256 Le
201…  157.56.242.198   443                                                                                       03 Len=0
201…  172.16.133.114   55270                                                                                     7 Len=0
201…  172.16.133.43    57700                                                                                     in=16379 Len=0
201…  172.16.133.43    57700                                                                                     in=16379 Len=0
201…  172.16.133.43    57700       172.16.139.250   5440     TCP                           57700→5440 [ACK] Seq=212 Ack=183 Win=16379 Len=0
201…  172.16.133.43    57700       172.16.139.250   5440     TCP                           57700→5440 [ACK] Seq=212 Ack=184 Win=16379 Len=0
201…  172.16.133.43    57700       172.16.139.250   5440     TCP                           57700→5440 [FIN, ACK] Seq=212 Ack=184 Win=16379 Len=0
▶ Frame 1: 1168 bytes on wire (9344 bits), 1168 bytes captured (9344 bits)
▶ Ethernet II, Src: Apple_d4:90:db (14:10:9f:d4:90:db), Dst: Watchgua_3e:02:d0 (00:90:7f:3e:02:d0)
▶ Internet Protocol Version 4, Src: Sean.local (172.16.133.57), Dst: 68.64.21.62 (68.64.21.62)
▶ User Datagram Protocol, Src Port: 53807, Dst Port: 1853
▶ Data (1126 bytes)

0000  00 90 7f 3e 02 d0 14 10  9f d4 90 db 08 00 45 00   ...>.... ......E.
0010  04 82 fd 44 00 00 40 11  ee 5e ac 10 85 39 44 40   ...D..@. .^...9D@
0020  15 3e d2 2f 07 3d 04 6e  43 4f c5 b8 2b dd d0 cc   .>./.=.n CO..+...
0030  1a 0f df bb 0d 0f 05 70  88 50 11 77 f1 b1 95 e2   .......p .P.w....
0040  ec 3a 88 65 92 aa 26 78  45 11 ac 84 4a 6f aa 9b   .:.e..&x E...Jo..
0050  6a c6 90 ce 5f 58 ea 63  24 53 40 19 30 55 18 eb   j..._X.c $S@.0U..
0060  ec 25 37 44 42 e8 d9 95  e1 e8 a9 8c 05 18 a4 83   .%7DB... ........

bigFlows          Packets: 791615 · Displayed: 791615 (100.0%) · Load time: 0:19.210    Profile: Default
```

State-of-the-art network forensics

- Wireshark is great, but

1. Usability
2. Performance
3. Resources



Wireshark · Export · HTTP object list
bigFlows.pcap
Wireshark · Export · HTTP object list
Apply a display filter ... <⌘/>
Help
Save All
Close
Save

```
Time   Source           Source Port  Destination
201…   172.16.133.57    53807        68.64.21.6
201…   172.16.133.57    53807        68.64.21.6
201…   172.16.133.57    53807        68.64.21.6
201…   96.43.146.176    443          172.16.133
201…   172.16.133.56    49514        68.64.21.4
201…   68.64.21.62      1853         172.16.133
201…   96.43.146.176    443          172.16.133
201…   96.43.146.176    443          172.16.133
201…   172.16.133.49    58246        68.64.21.4
201…   172.16.133.103   63406        216.115.22
201…   172.16.133.43    57700        172.16.139
201…   68.64.21.41      1853         172.16.133
201…   172.16.133.43    57700        172.16.139
201…   67.217.94.135    8200         172.16.133
201…   67.217.94.156    8200         172.16.133
201…   67.217.94.204    8200         172.16.133
201…   172.16.133.53    49292        172.16.139
201…   172.16.133.53    49292        172.16.139
201…   157.56.242.198   443          172.16.133
201…   172.16.133.114   55270        67.217.65.
201…   172.16.133.43    57700        172.16.139
201…   172.16.133.43    57700        172.16.139
201…   172.16.133.43    57700        172.16.139
201…   172.16.133.43    57700        172.16.139
201…   172.16.133.43    57700        172.16.139
▶ Frame 1: 1168 bytes on wire (9344 bits)
▶ Ethernet II, Src: Apple_d4:90:db (14:10
▶ Internet Protocol Version 4, Src: Sean.
▶ User Datagram Protocol, Src Port: 53807
▶ Data (1126 bytes)

0000  00 90 7f 3e 02 d0 14 10  9f d4 90 d
0010  04 82 fd 44 00 00 40 11  ee 5e ac 1
0020  15 3e d2 2f 07 3d 04 6e  43 4f c5 b
0030  1a 0f df bb 0d 0f 05 70  88 50 11 77 f1 b1 95 e2   .......p .P.w....
0040  ec 3a 88 65 92 aa 26 78  45 11 ac 84 4a 6f aa 9b   .:.e..&x E...Jo..
0050  6a c6 90 ce 5f 58 ea 63  24 53 40 19 30 55 18 eb   j..._X.c $S@.0U..
0060  ec 25 37 44 42 e8 d9 95  e1 e8 a9 8c 05 18 a4 83   .%7DB... ........

bigFlows        Packets: 791615 · Displayed: 791615 (100.0%) · Load time: 0:19.210   Profile: Default
```

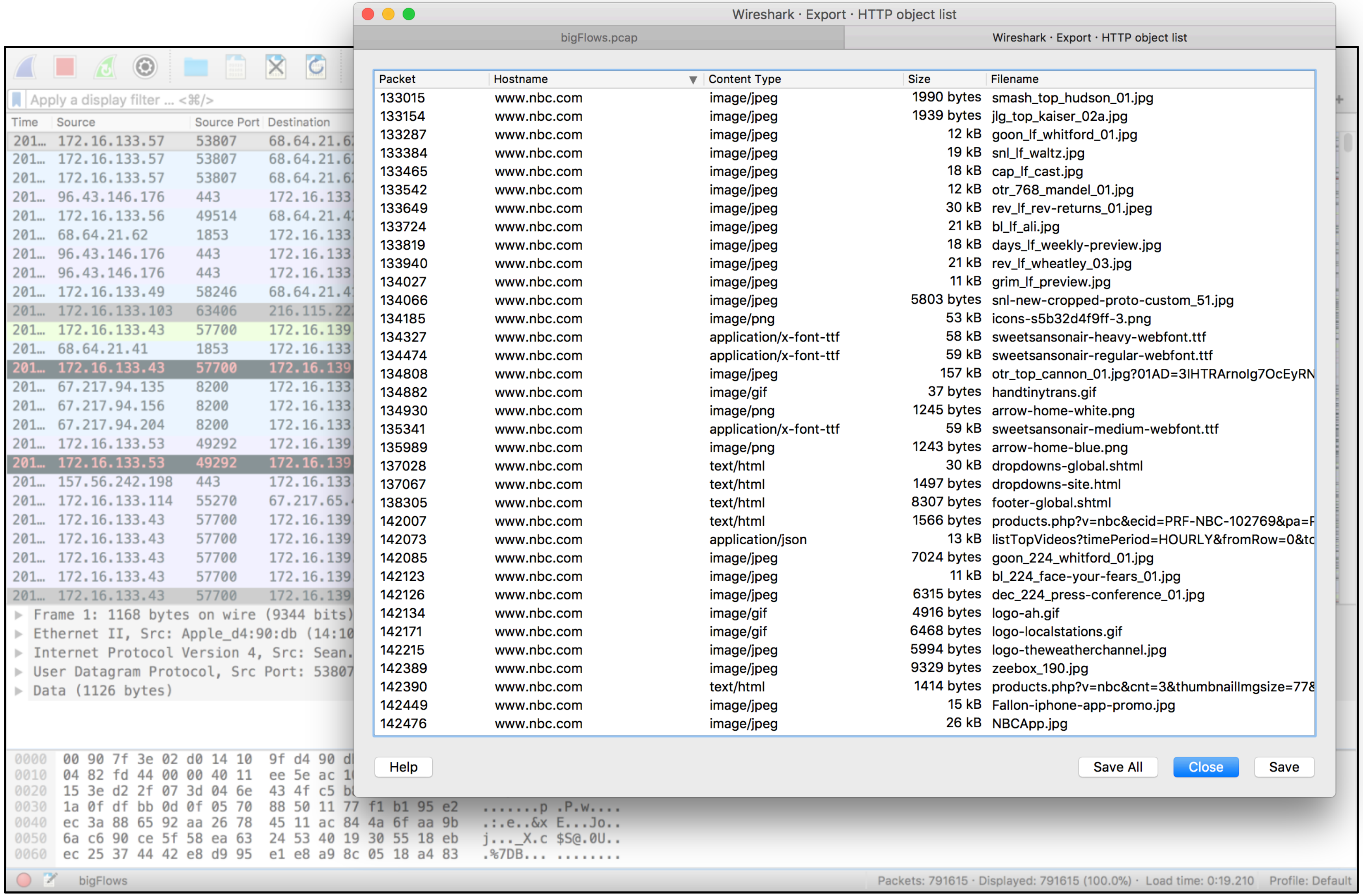
Packet	Hostname	Content Type	Size	Filename
133015	www.nbc.com	image/jpeg	1990 bytes	smash_top_hudson_01.jpg
133154	www.nbc.com	image/jpeg	1939 bytes	jlg_top_kaiser_02a.jpg
133287	www.nbc.com	image/jpeg	12 kB	goon_lf_whitford_01.jpg
133384	www.nbc.com	image/jpeg	19 kB	snl_lf_waltz.jpg
133465	www.nbc.com	image/jpeg	18 kB	cap_lf_cast.jpg
133542	www.nbc.com	image/jpeg	12 kB	otr_768_mandel_01.jpg
133649	www.nbc.com	image/jpeg	30 kB	rev_lf_rev-returns_01.jpeg
133724	www.nbc.com	image/jpeg	21 kB	bl_lf_ali.jpg
133819	www.nbc.com	image/jpeg	18 kB	days_lf_weekly-preview.jpg
133940	www.nbc.com	image/jpeg	21 kB	rev_lf_wheatley_03.jpg
134027	www.nbc.com	image/jpeg	11 kB	grim_lf_preview.jpg
134066	www.nbc.com	image/jpeg	5803 bytes	snl-new-cropped-proto-custom_51.jpg
134185	www.nbc.com	image/png	53 kB	icons-s5b32d4f9ff-3.png
134327	www.nbc.com	application/x-font-ttf	58 kB	sweetsansonair-heavy-webfont.ttf
134474	www.nbc.com	application/x-font-ttf	59 kB	sweetsansonair-regular-webfont.ttf
134808	www.nbc.com	image/jpeg	157 kB	otr_top_cannon_01.jpg?01AD=3IHTRArnolg7OcEyRN
134882	www.nbc.com	image/gif	37 bytes	handtinytrans.gif
134930	www.nbc.com	image/png	1245 bytes	arrow-home-white.png
135341	www.nbc.com	application/x-font-ttf	59 kB	sweetsansonair-medium-webfont.ttf
135989	www.nbc.com	image/png	1243 bytes	arrow-home-blue.png
137028	www.nbc.com	text/html	30 kB	dropdowns-global.shtml
137067	www.nbc.com	text/html	1497 bytes	dropdowns-site.html
138305	www.nbc.com	text/html	8307 bytes	footer-global.shtml
142007	www.nbc.com	text/html	1566 bytes	products.php?v=nbc&ecid=PRF-NBC-102769&pa=F
142073	www.nbc.com	application/json	13 kB	listTopVideos?timePeriod=HOURLY&fromRow=0&tc
142085	www.nbc.com	image/jpeg	7024 bytes	goon_224_whitford_01.jpg
142123	www.nbc.com	image/jpeg	11 kB	bl_224_face-your-fears_01.jpg
142126	www.nbc.com	image/jpeg	6315 bytes	dec_224_press-conference_01.jpg
142134	www.nbc.com	image/gif	4916 bytes	logo-ah.gif
142171	www.nbc.com	image/gif	6468 bytes	logo-localstations.gif
142215	www.nbc.com	image/jpeg	5994 bytes	logo-theweatherchannel.jpg
142389	www.nbc.com	image/jpeg	9329 bytes	zeebox_190.jpg
142390	www.nbc.com	text/html	1414 bytes	products.php?v=nbc&cnt=3&thumbnailImgsize=77&
142449	www.nbc.com	image/jpeg	15 kB	Fallon-iphone-app-promo.jpg
142476	www.nbc.com	image/jpeg	26 kB	NBCApp.jpg

State-of-the-art network forensics

- Wireshark is great, but

1. Usability
2. Performance
3. Resources

- How else can you access a pcap?



```
Time  Source           Source Port  Destination
201…  172.16.133.57    53807        68.64.21.6
201…  172.16.133.57    53807        68.64.21.6
201…  172.16.133.57    53807        68.64.21.6
201…  96.43.146.176    443          172.16.133
201…  172.16.133.56    49514        68.64.21.4
201…  68.64.21.62      1853         172.16.133
201…  96.43.146.176    443          172.16.133
201…  96.43.146.176    443          172.16.133
201…  172.16.133.49    58246        68.64.21.4
201…  172.16.133.103   63406        216.115.22
201…  172.16.133.43    57700        172.16.139
201…  68.64.21.41      1853         172.16.133
201…  172.16.133.43    57700        172.16.139
201…  67.217.94.135    8200         172.16.133
201…  67.217.94.156    8200         172.16.133
201…  67.217.94.204    8200         172.16.133
201…  172.16.133.53    49292        172.16.139
201…  172.16.133.53    49292        172.16.139
201…  157.56.242.198   443          172.16.133
201…  172.16.133.114   55270        67.217.65.
201…  172.16.133.43    57700        172.16.139
201…  172.16.133.43    57700        172.16.139
201…  172.16.133.43    57700        172.16.139
201…  172.16.133.43    57700        172.16.139
201…  172.16.133.43    57700        172.16.139
▶ Frame 1: 1168 bytes on wire (9344 bits)
▶ Ethernet II, Src: Apple_d4:90:db (14:10
▶ Internet Protocol Version 4, Src: Sean.
▶ User Datagram Protocol, Src Port: 53807
▶ Data (1126 bytes)

0000  00 90 7f 3e 02 d0 14 10  9f d4 90 d
0010  04 82 fd 44 00 00 40 11  ee 5e ac 1
0020  15 3e d2 2f 07 3d 04 6e  43 4f c5 b
0030  1a 0f df bb 0d 0f 05 70  88 50 11 77 f1 b1 95 e2   .......p .P.w....
0040  ec 3a 88 65 92 aa 26 78  45 11 ac 84 4a 6f aa 9b   .:.e..&x E...Jo..
0050  6a c6 90 ce 5f 58 ea 63  24 53 40 19 30 55 18 eb   j..._X.c $S@.0U..
0060  ec 25 37 44 42 e8 d9 95  e1 e8 a9 8c 05 18 a4 83   .%7DB... ........
```

bigFlows   Packets: 791615 · Displayed: 791615 (100.0%) · Load time: 0:19.210   Profile: Default

Wireshark · Export · HTTP object list

bigFlows.pcap | Wireshark · Export · HTTP object list

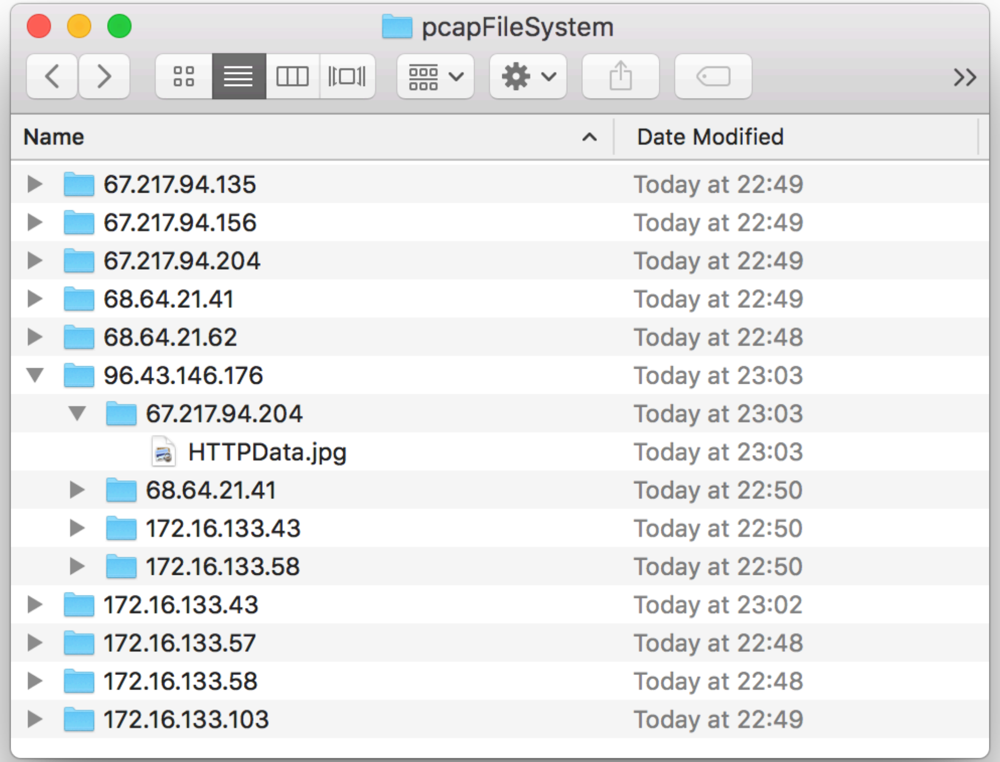
Packet	Hostname	Content Type	Size	Filename
133015	www.nbc.com	image/jpeg	1990 bytes	smash_top_hudson_01.jpg
133154	www.nbc.com	image/jpeg	1939 bytes	jlg_top_kaiser_02a.jpg
133287	www.nbc.com	image/jpeg	12 kB	goon_lf_whitford_01.jpg
133384	www.nbc.com	image/jpeg	19 kB	snl_lf_waltz.jpg
133465	www.nbc.com	image/jpeg	18 kB	cap_lf_cast.jpg
133542	www.nbc.com	image/jpeg	12 kB	otr_768_mandel_01.jpg
133649	www.nbc.com	image/jpeg	30 kB	rev_lf_rev-returns_01.jpeg
133724	www.nbc.com	image/jpeg	21 kB	bl_lf_ali.jpg
133819	www.nbc.com	image/jpeg	18 kB	days_lf_weekly-preview.jpg
133940	www.nbc.com	image/jpeg	21 kB	rev_lf_wheatley_03.jpg
134027	www.nbc.com	image/jpeg	11 kB	grim_lf_preview.jpg
134066	www.nbc.com	image/jpeg	5803 bytes	snl-new-cropped-proto-custom_51.jpg
134185	www.nbc.com	image/png	53 kB	icons-s5b32d4f9ff-3.png
134327	www.nbc.com	application/x-font-ttf	58 kB	sweetsansonair-heavy-webfont.ttf
134474	www.nbc.com	application/x-font-ttf	59 kB	sweetsansonair-regular-webfont.ttf
134808	www.nbc.com	image/jpeg	157 kB	otr_top_cannon_01.jpg?01AD=3IHTRArnolg7OcEyRN
134882	www.nbc.com	image/gif	37 bytes	handtinytrans.gif
134930	www.nbc.com	image/png	1245 bytes	arrow-home-white.png
135341	www.nbc.com	application/x-font-ttf	59 kB	sweetsansonair-medium-webfont.ttf
135989	www.nbc.com	image/png	1243 bytes	arrow-home-blue.png
137028	www.nbc.com	text/html	30 kB	dropdowns-global.shtml
137067	www.nbc.com	text/html	1497 bytes	dropdowns-site.html
138305	www.nbc.com	text/html	8307 bytes	footer-global.shtml
142007	www.nbc.com	text/html	1566 bytes	products.php?v=nbc&ecid=PRF-NBC-102769&pa=F
142073	www.nbc.com	application/json	13 kB	listTopVideos?timePeriod=HOURLY&fromRow=0&tc
142085	www.nbc.com	image/jpeg	7024 bytes	goon_224_whitford_01.jpg
142123	www.nbc.com	image/jpeg	11 kB	bl_224_face-your-fears_01.jpg
142126	www.nbc.com	image/jpeg	6315 bytes	dec_224_press-conference_01.jpg
142134	www.nbc.com	image/gif	4916 bytes	logo-ah.gif
142171	www.nbc.com	image/gif	6468 bytes	logo-localstations.gif
142215	www.nbc.com	image/jpeg	5994 bytes	logo-theweatherchannel.jpg
142389	www.nbc.com	image/jpeg	9329 bytes	zeebox_190.jpg
142390	www.nbc.com	text/html	1414 bytes	products.php?v=nbc&cnt=3&thumbnailImgsize=778
142449	www.nbc.com	image/jpeg	15 kB	Fallon-iphone-app-promo.jpg
142476	www.nbc.com	image/jpeg	26 kB	NBCApp.jpg

Fraunhofer FKIE

Idea

20180706.pcap

- File systems organize unstructured data and make them available to the user
  - Create a file system for pcaps



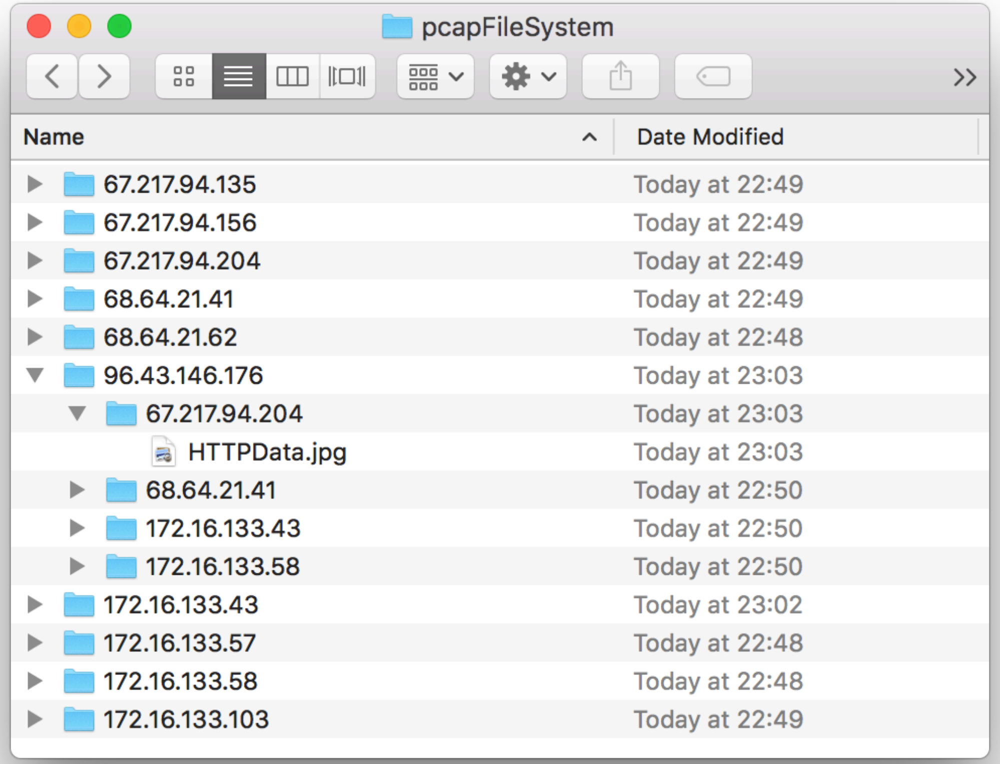
```
Name                          Date Modified
▶ 67.217.94.135               Today at 22:49
▶ 67.217.94.156               Today at 22:49
▶ 67.217.94.204               Today at 22:49
▶ 68.64.21.41                 Today at 22:49
▶ 68.64.21.62                 Today at 22:48
▼ 96.43.146.176               Today at 23:03
    ▼ 67.217.94.204           Today at 23:03
          HTTPData.jpg        Today at 23:03
    ▶ 68.64.21.41             Today at 22:50
    ▶ 172.16.133.43           Today at 22:50
    ▶ 172.16.133.58           Today at 22:50
▶ 172.16.133.43               Today at 23:02
▶ 172.16.133.57               Today at 22:48
▶ 172.16.133.58               Today at 22:48
▶ 172.16.133.103              Today at 22:49
```

Idea

- File systems organize unstructured data and make them available to the user
  - Create a file system for pcaps
- Create a structure, which can be used when accessing the same network capture again
  - Create an index file keeping track of the files in the file system

20180706.index

20180706.pcap

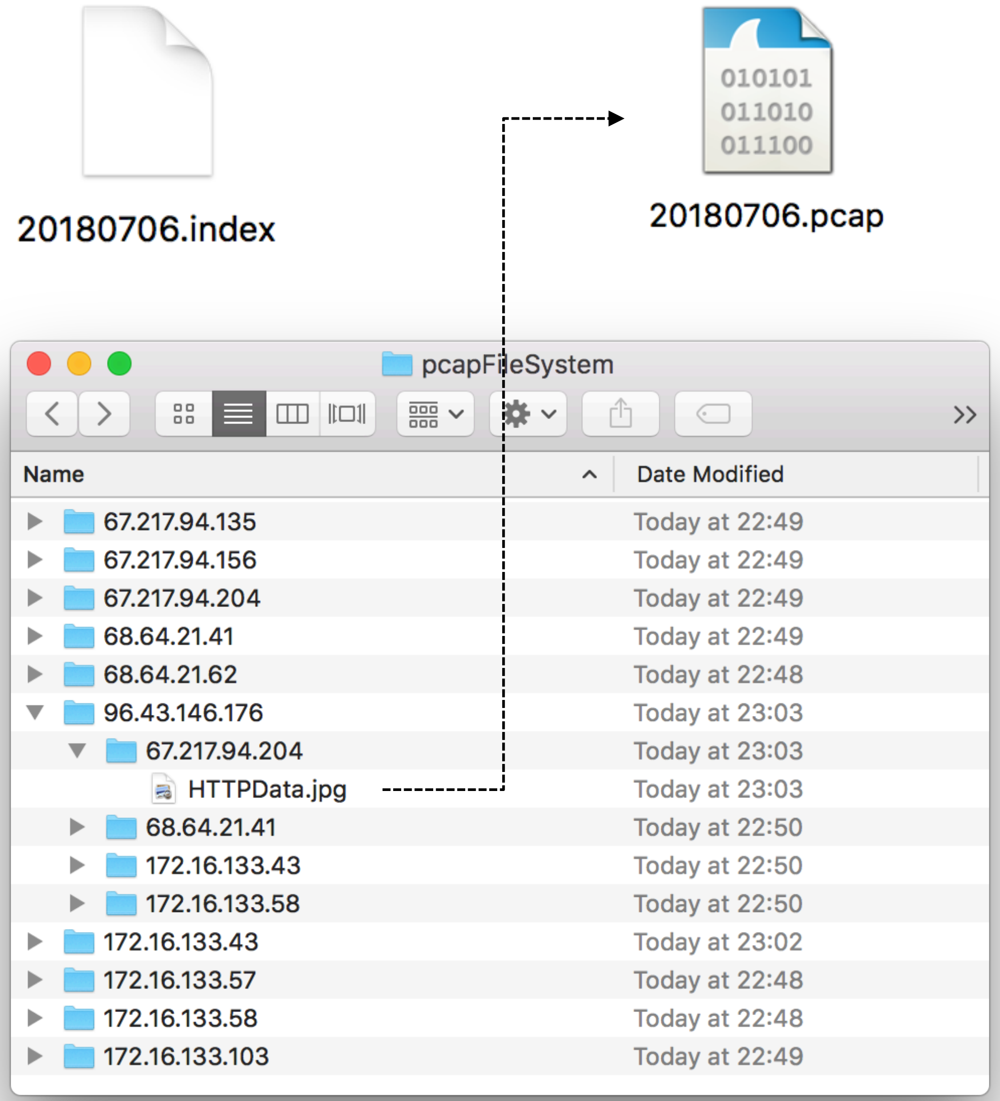


pcapFileSystem

Name	Date Modified
▶ 67.217.94.135	Today at 22:49
▶ 67.217.94.156	Today at 22:49
▶ 67.217.94.204	Today at 22:49
▶ 68.64.21.41	Today at 22:49
▶ 68.64.21.62	Today at 22:48
▼ 96.43.146.176	Today at 23:03
▼ 67.217.94.204	Today at 23:03
HTTPData.jpg	Today at 23:03
▶ 68.64.21.41	Today at 22:50
▶ 172.16.133.43	Today at 22:50
▶ 172.16.133.58	Today at 22:50
▶ 172.16.133.43	Today at 23:02
▶ 172.16.133.57	Today at 22:48
▶ 172.16.133.58	Today at 22:48
▶ 172.16.133.103	Today at 22:49

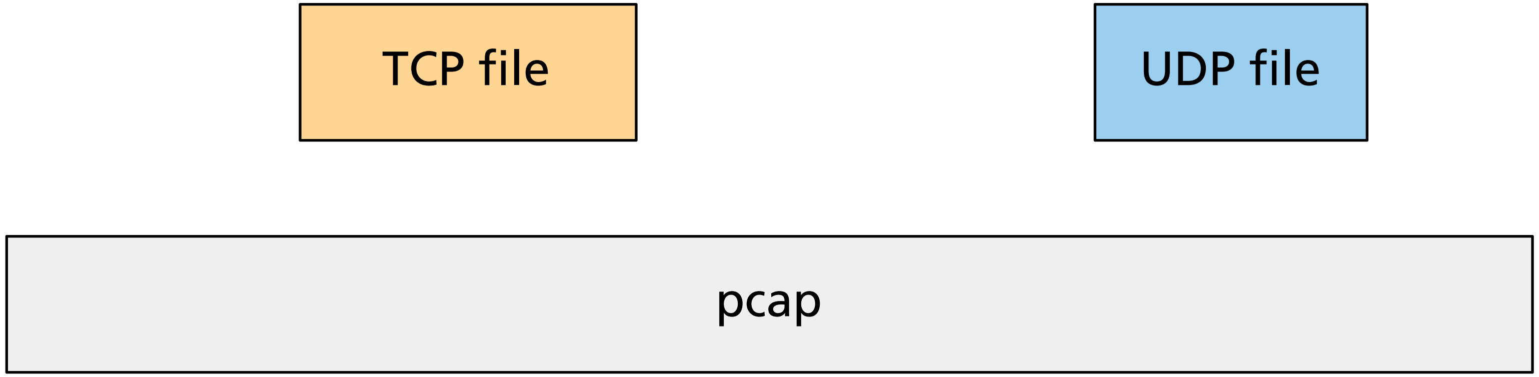
Idea

- File systems organize unstructured data and make them available to the user
  - Create a file system for pcaps
- Create a structure, which can be used when accessing the same network capture again
  - Create an index file keeping track of the files in the file system
- Extracting data in order to process it creates unnecessary overhead
  - Point directly into the data in the pcap

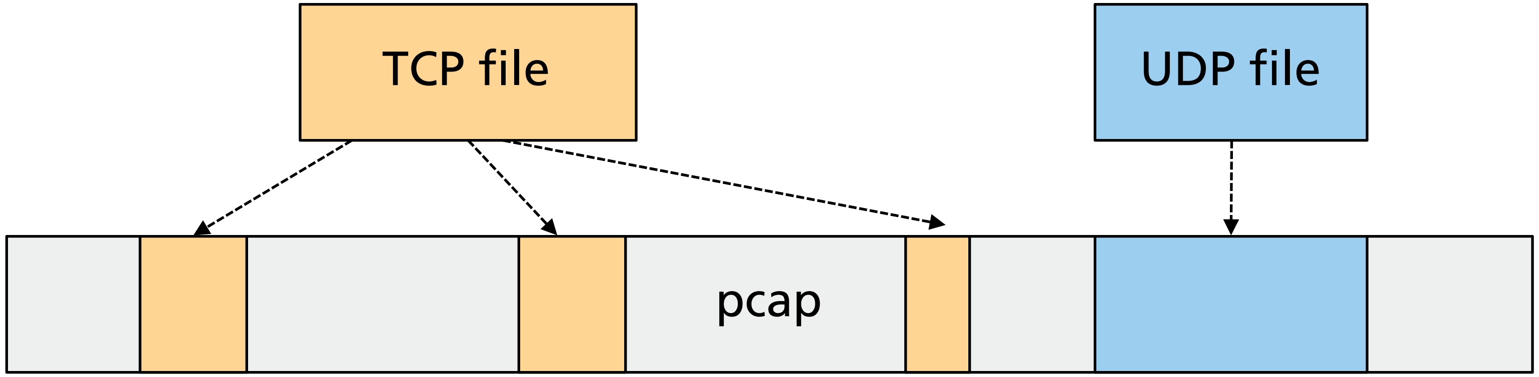


```
Name                              Date Modified
▶ 67.217.94.135                   Today at 22:49
▶ 67.217.94.156                   Today at 22:49
▶ 67.217.94.204                   Today at 22:49
▶ 68.64.21.41                     Today at 22:49
▶ 68.64.21.62                     Today at 22:48
▼ 96.43.146.176                   Today at 23:03
    ▼ 67.217.94.204               Today at 23:03
          HTTPData.jpg            Today at 23:03
    ▶ 68.64.21.41                 Today at 22:50
    ▶ 172.16.133.43               Today at 22:50
    ▶ 172.16.133.58               Today at 22:50
▶ 172.16.133.43                   Today at 23:02
▶ 172.16.133.57                   Today at 22:48
▶ 172.16.133.58                   Today at 22:48
▶ 172.16.133.103                  Today at 22:49
```

Concept

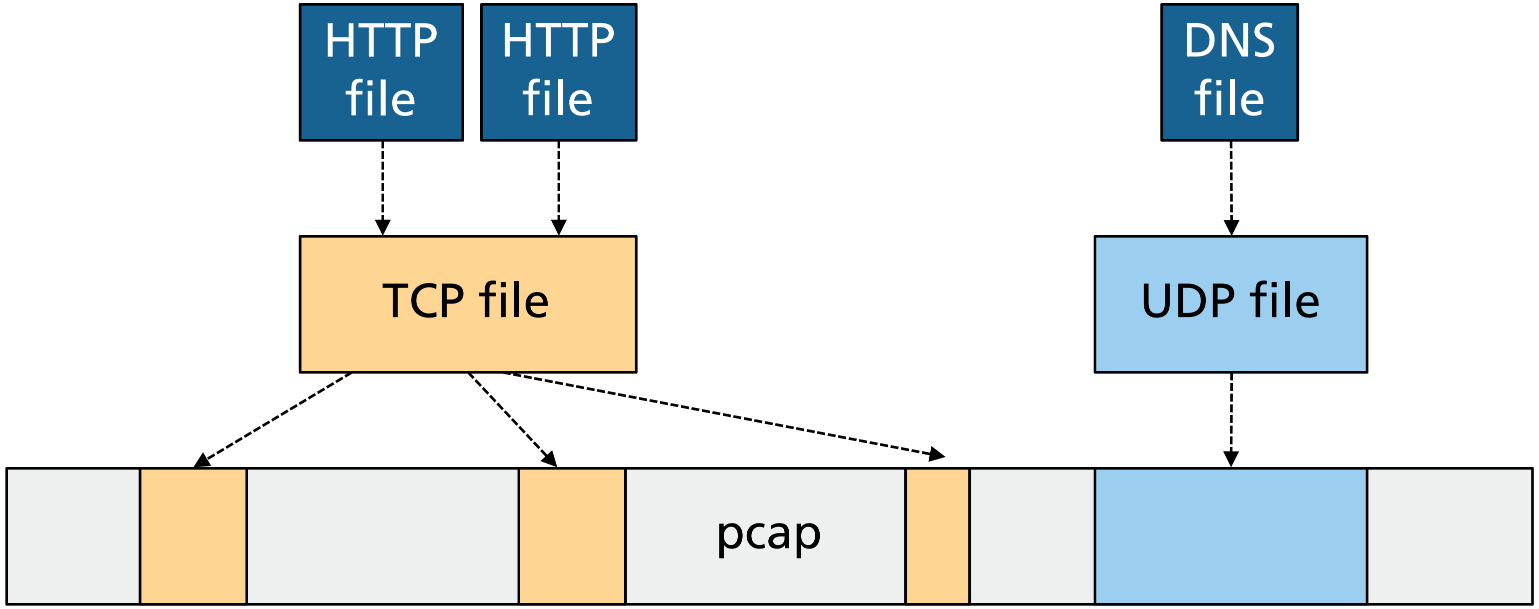


Concept



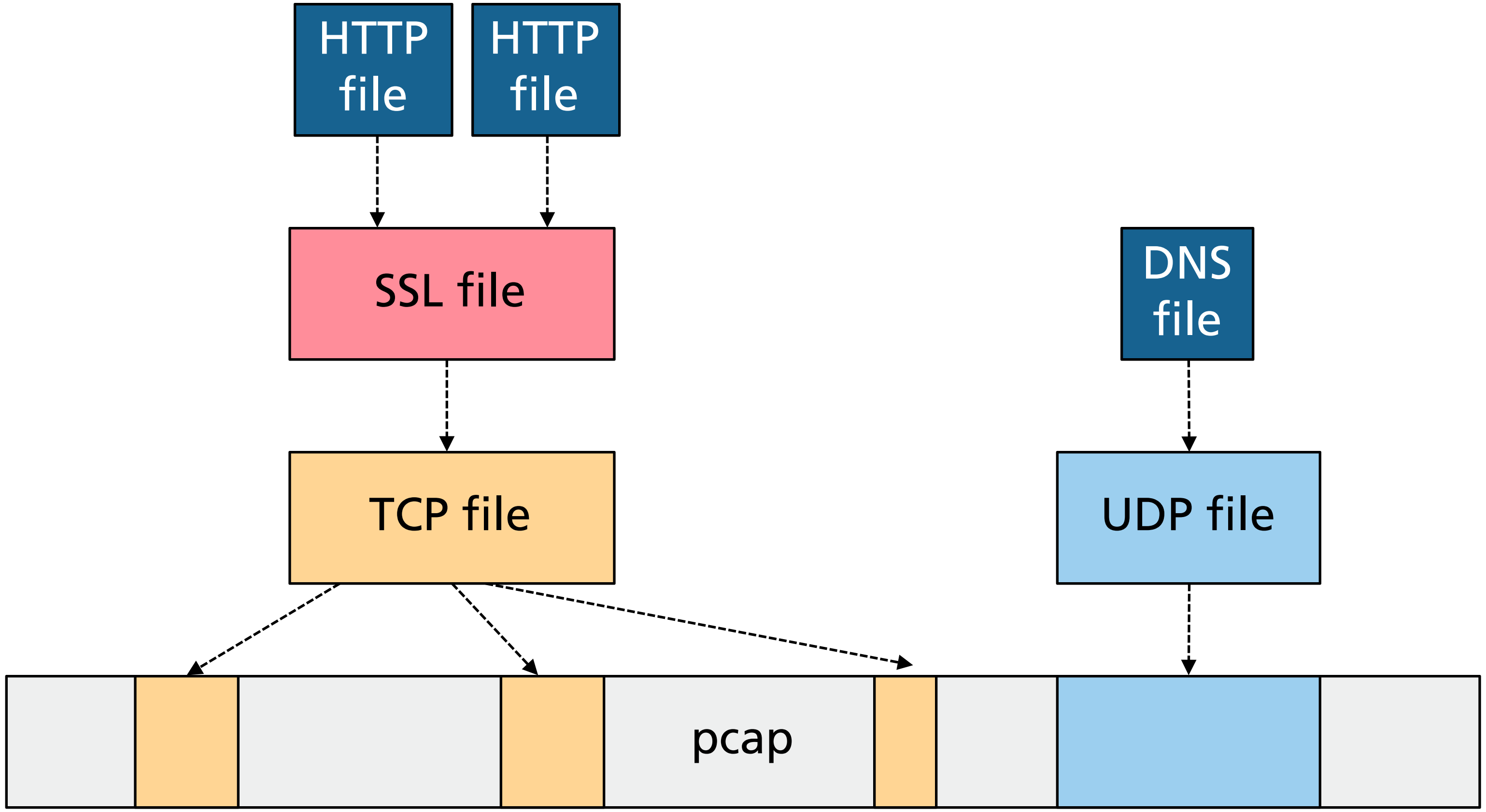
TCP and UDP files point directly into the pcap

Concept



Application protocols can then point into the TCP and UDP files

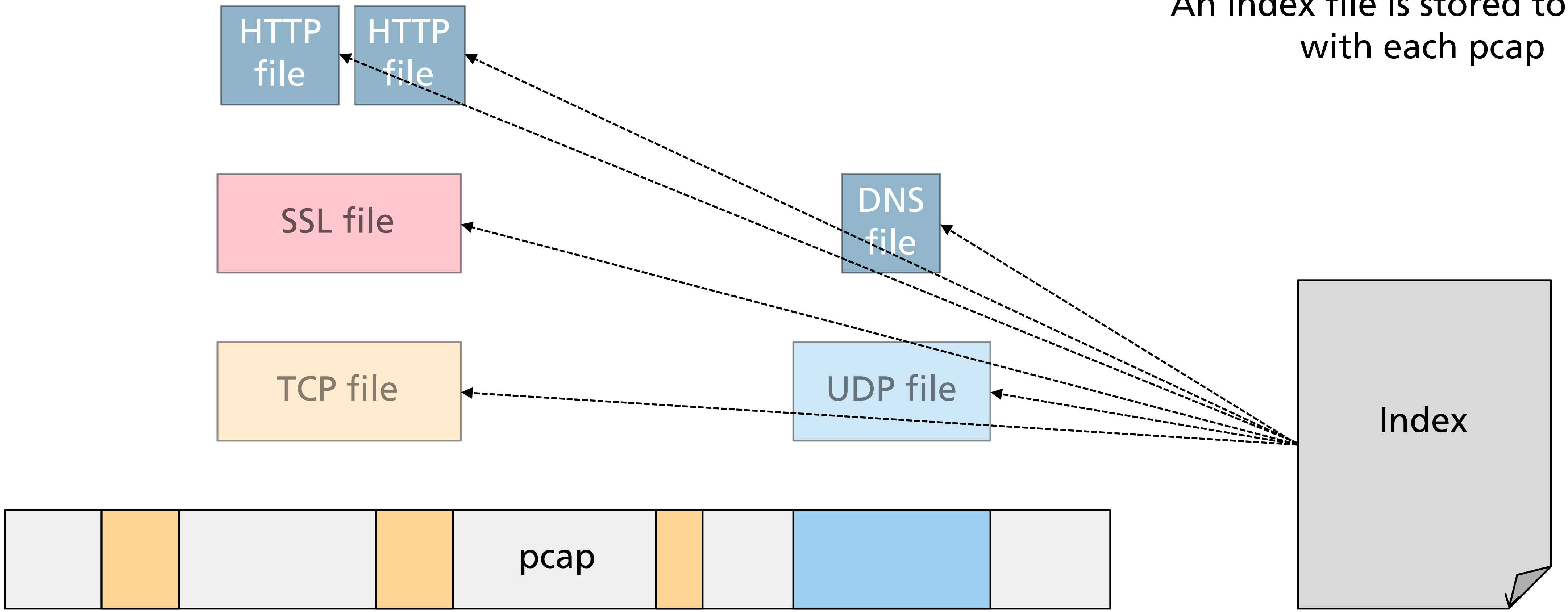
Concept



Other protocols add new virtual layers in between

Concept

An index file is stored together with each pcap



pcapFS

- pcapFS is a FUSE module mounting captured network data as a virtual file system
  - Filesystem in Userspace is part of the Linux kernel and available for multiple operating systems including FreeBSD, OpenBSD and MacOS
  - Another ”pcapFS” was already released as part of the PyFlag framework by Michael Cohen
    - Unfortunately deprecated and not maintained ☹
- Index files can be stored in memory or on disk for future mounts
- Protocols are implemented by virtual file classes

Demo

pcapFS vs. Wireshark

Demo

Demo

pcapFS vs. Wireshark

- Usability
  - Data is presented using the virtual file system
  - Its hierarchy can be specified using multiple sorting options
- Performance
  - First mount of a pcap creates an index file
  - Browsing through the mounted data takes almost no time
  - Mounting with a used index is significantly faster than Wireshark
- Resources
  - Files in pcapFS point directly into the pcap or other virtual files
  - They are only extracted on demand

Demo

Beyond Wireshark

Demo

Demo

Beyond Wireshark

- pcapFS supports mounting of split pcap files
- File system level tools can be used on the mounted data without any extraction
- Metadata can be preprocessed and displayed as an own file as for example:
  - HTTP header
  - DNS requests and responses (e.g. as JSON)
- Missing data in streams can easily be padded for reconstruction

Demo

Working with pcapFS

Demo

Demo

Working with pcapFS

- Decryption of data by providing the corresponding key files
  - More cipher suites for SSL will be added in the future
  - Key files can be implemented for multiple protocols
- Configuration files force a protocol decoding on files with specified properties:
  - e.g. `XOR dstPort 31489 protocol http`

Summary

- pcapFS gives investigators the possibility to
  - quickly take a look at the relevant data of a network capture
  - order the data by different criteria
  - use file system level tools for their analysis
- Keeping an index file for each pcap significantly increases the performance of analyzing pcaps
- Using virtual files eliminates the overhead of extracting data out of pcaps

Future Plans

- Add support for more protocols (wishes are more than welcome!)
  - Particularly add support for other cipher suites in SSL
  - BitTorrent, HTTP2, SMB
- Add support for more metadata
  - e.g. SSL certificates
- Make use of Symbolic Links (e.g. reverse connections)
- Add support for pcapng

Thanks for your attention!

https://github.com/fkie-cad/pcapfs

jan-niclas.hilgert@fkie.fraunhofer.de