# DATA WRANGLING: MAKING SENSE OF DATA IN LINUX

**Argandov**

April, 2021

# $ cat about_me.txt

Originally a Mechanical Engineer, I started my journey into Cybersecurity around 2019 and haven’t looked back since.

I’m interested in Stoic and Eastern philosophies, which I believe can greatly improve our thinking. I’ve also been playing guitar and bass for 22 years now, 10 of them professionally in a Heavy Metal band🤘

# What we’ll be doing here

**Pcap analysis:**

- Tcpdump
- NetworkMiner or Wireshark

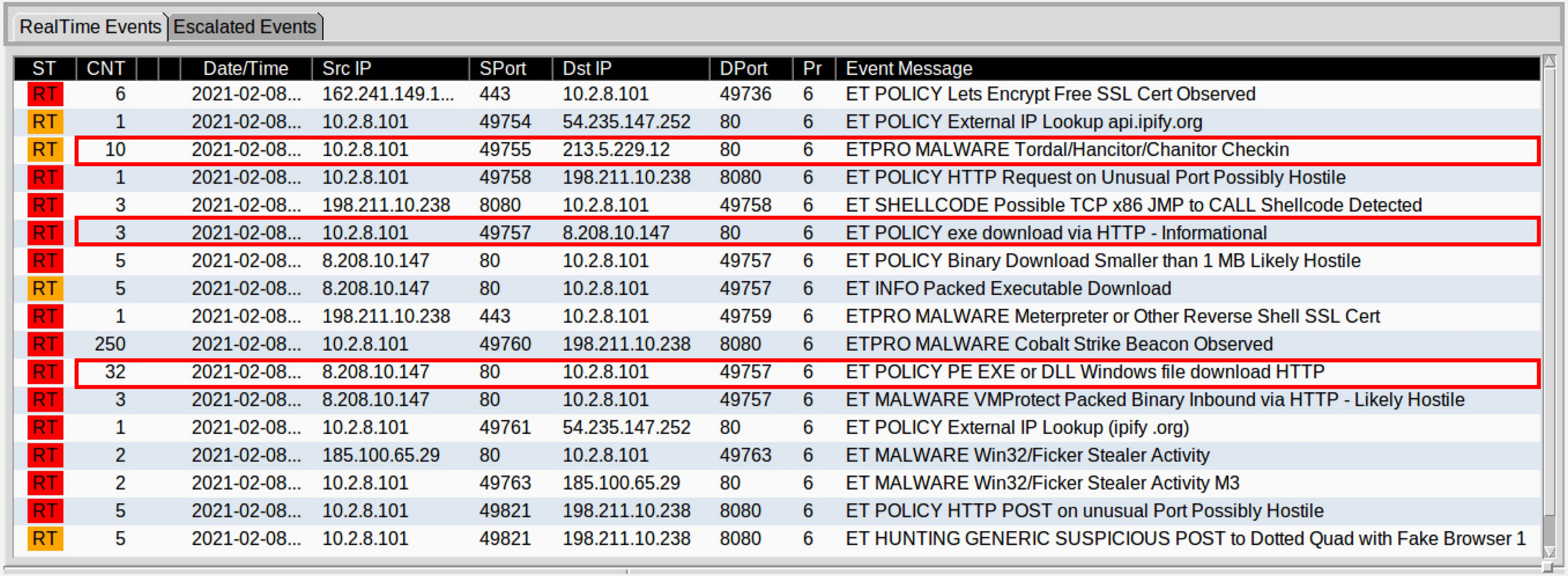
**Data we’re provided with:**

- Alerts from an NSM for us to investigate
- Pcap file, captured at the moment of the events

**Data we expect to gather (IoCs):**

- Malicious files being transfered to the system
- Confirmation of Communications with C&C (Command & control)

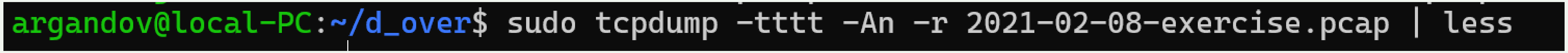
# Sguil Alerts



| ST | CNT | | | Date/Time | Src IP | SPort | Dst IP | DPort | Pr | Event Message |
|---|---|---|---|---|---|---|---|---|---|---|
| RT | 6 | | | 2021-02-08... | 162.241.149.1... | 443 | 10.2.8.101 | 49736 | 6 | ET POLICY Lets Encrypt Free SSL Cert Observed |
| RT | 1 | | | 2021-02-08... | 10.2.8.101 | 49754 | 54.235.147.252 | 80 | 6 | ET POLICY External IP Lookup api.ipify.org |
| RT | 10 | | | 2021-02-08... | 10.2.8.101 | 49755 | 213.5.229.12 | 80 | 6 | ETPRO MALWARE Tordal/Hancitor/Chanitor Checkin |
| RT | 1 | | | 2021-02-08... | 10.2.8.101 | 49758 | 198.211.10.238 | 8080 | 6 | ET POLICY HTTP Request on Unusual Port Possibly Hostile |
| RT | 3 | | | 2021-02-08... | 198.211.10.238 | 8080 | 10.2.8.101 | 49758 | 6 | ET SHELLCODE Possible TCP x86 JMP to CALL Shellcode Detected |
| RT | 3 | | | 2021-02-08... | 10.2.8.101 | 49757 | 8.208.10.147 | 80 | 6 | ET POLICY exe download via HTTP - Informational |
| RT | 5 | | | 2021-02-08... | 8.208.10.147 | 80 | 10.2.8.101 | 49757 | 6 | ET POLICY Binary Download Smaller than 1 MB Likely Hostile |
| RT | 5 | | | 2021-02-08... | 8.208.10.147 | 80 | 10.2.8.101 | 49757 | 6 | ET INFO Packed Executable Download |
| RT | 1 | | | 2021-02-08... | 198.211.10.238 | 443 | 10.2.8.101 | 49759 | 6 | ETPRO MALWARE Meterpreter or Other Reverse Shell SSL Cert |
| RT | 250 | | | 2021-02-08... | 10.2.8.101 | 49760 | 198.211.10.238 | 8080 | 6 | ETPRO MALWARE Cobalt Strike Beacon Observed |
| RT | 32 | | | 2021-02-08... | 8.208.10.147 | 80 | 10.2.8.101 | 49757 | 6 | ET POLICY PE EXE or DLL Windows file download HTTP |
| RT | 3 | | | 2021-02-08... | 8.208.10.147 | 80 | 10.2.8.101 | 49757 | 6 | ET MALWARE VMProtect Packed Binary Inbound via HTTP - Likely Hostile |
| RT | 1 | | | 2021-02-08... | 10.2.8.101 | 49761 | 54.235.147.252 | 80 | 6 | ET POLICY External IP Lookup (ipify .org) |
| RT | 2 | | | 2021-02-08... | 185.100.65.29 | 80 | 10.2.8.101 | 49763 | 6 | ET MALWARE Win32/Ficker Stealer Activity |
| RT | 2 | | | 2021-02-08... | 10.2.8.101 | 49763 | 185.100.65.29 | 80 | 6 | ET MALWARE Win32/Ficker Stealer Activity M3 |
| RT | 5 | | | 2021-02-08... | 10.2.8.101 | 49821 | 198.211.10.238 | 8080 | 6 | ET POLICY HTTP POST on unusual Port Possibly Hostile |
| RT | 5 | | | 2021-02-08... | 10.2.8.101 | 49821 | 198.211.10.238 | 8080 | 6 | ET HUNTING GENERIC SUSPICIOUS POST to Dotted Quad with Fake Browser 1 |

➢ **Let’s pick 3 alerts and work on them for now**

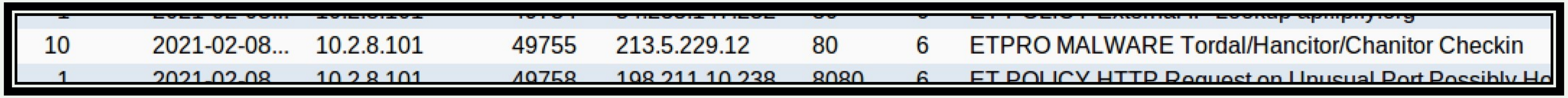
# First we read our pcap file with tcpdump



```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | less
```

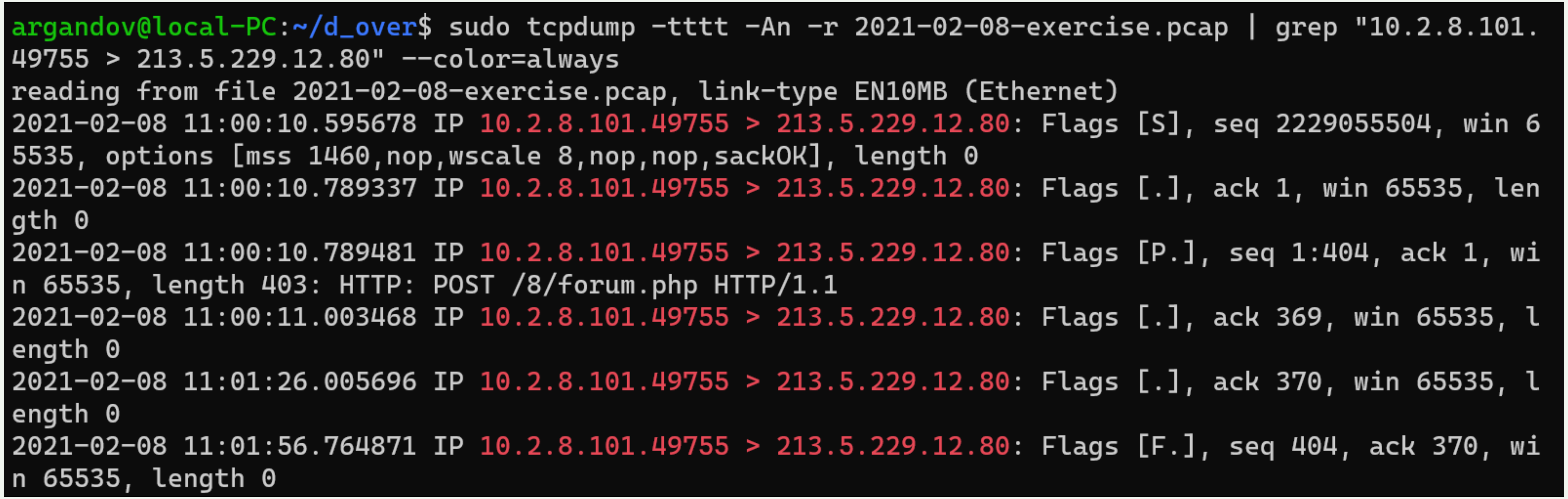
- We take a first glance at the file.
- Get familiar with the file format and the way info is presented.

# Let’s look at the first alert and the CLI process



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 10 | 2021-02-08... | 10.2.8.101 | 49755 | 213.5.229.12 | 80 | 6 | ETPRO MALWARE Tordal/Hancitor/Chanitor Checkin |
| 1 | 2021-02-08 | 10.2.8.101 | 49758 | 198.211.10.238 | 8080 | 6 | ET POLICY HTTP Request on Unusual Port Possibly Ho |

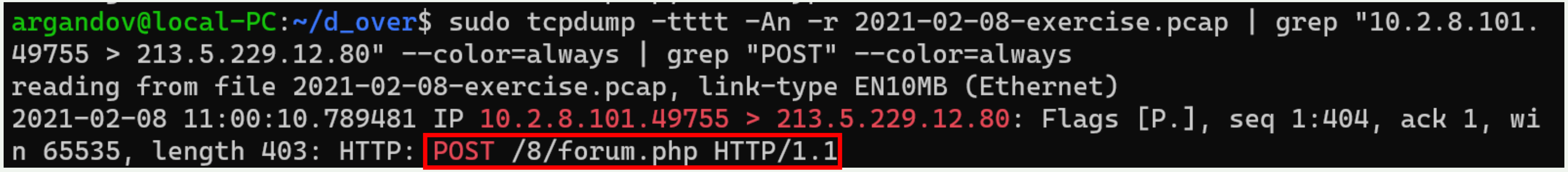
- ➢ **Let’s do an initial grep through the file, with the knowledge we now have:**



```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | grep "10.2.8.101.
49755 > 213.5.229.12.80" --color=always
reading from file 2021-02-08-exercise.pcap, link-type EN10MB (Ethernet)
2021-02-08 11:00:10.595678 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [S], seq 2229055504, win 6
5535, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0
2021-02-08 11:00:10.789337 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [.], ack 1, win 65535, len
gth 0
2021-02-08 11:00:10.789481 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [P.], seq 1:404, ack 1, wi
n 65535, length 403: HTTP: POST /8/forum.php HTTP/1.1
2021-02-08 11:00:11.003468 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [.], ack 369, win 65535, l
ength 0
2021-02-08 11:01:26.005696 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [.], ack 370, win 65535, l
ength 0
2021-02-08 11:01:56.764871 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [F.], seq 404, ack 370, wi
n 65535, length 0
```

# Any interesting results? Yes, and very useful!

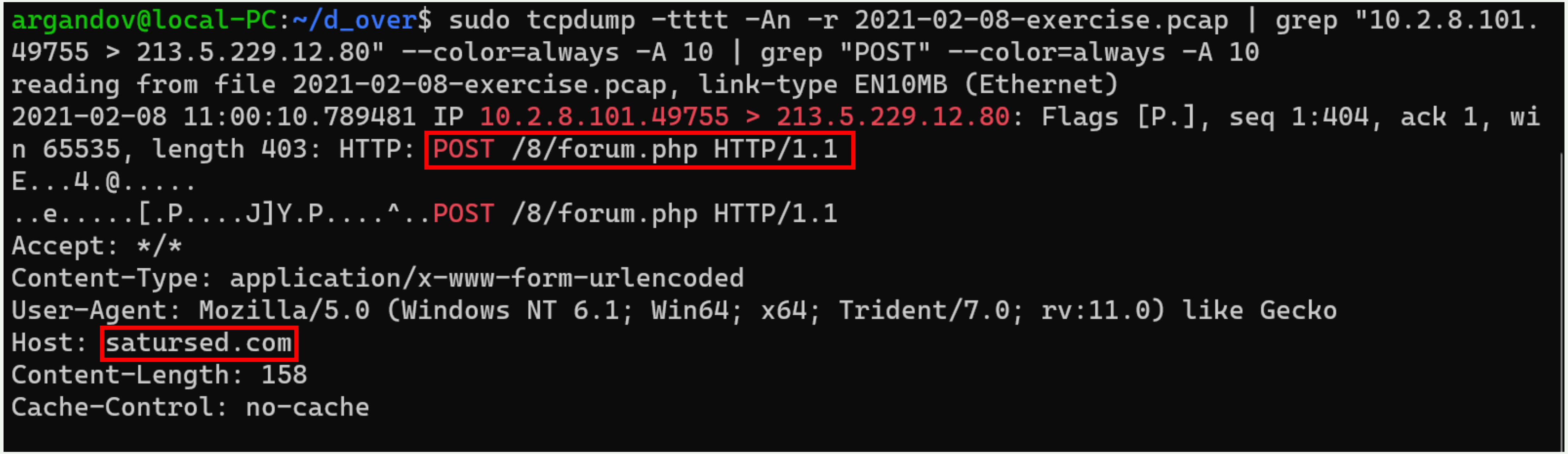
- ➢ **Let’s isolate the interesting part:**



```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | grep "10.2.8.101.
49755 > 213.5.229.12.80" --color=always | grep "POST" --color=always
reading from file 2021-02-08-exercise.pcap, link-type EN10MB (Ethernet)
2021-02-08 11:00:10.789481 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [P.], seq 1:404, ack 1, wi
n 65535, length 403: HTTP: POST /8/forum.php HTTP/1.1
```

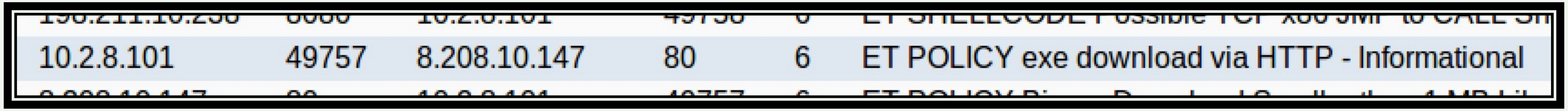
- ➢ **Remember: “grep” Will only display at stdout the line in which our search matches. We will need more context.**

# Getting more context from our previous results

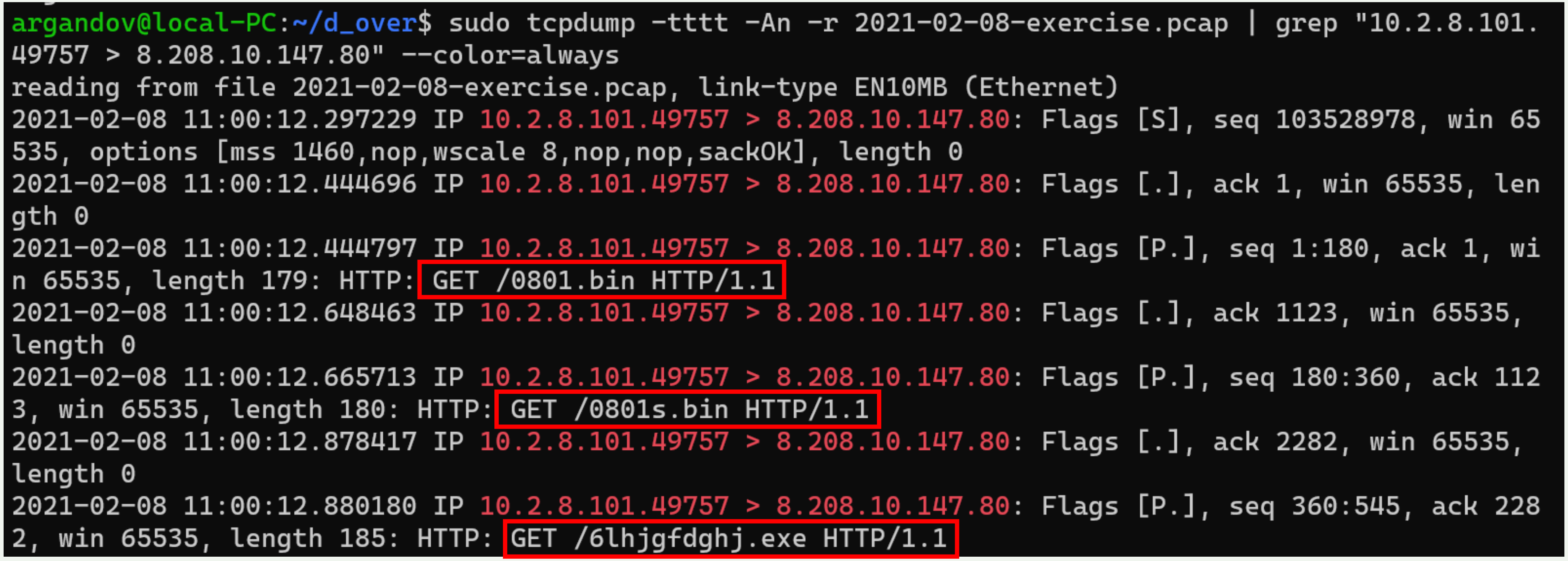


```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | grep "10.2.8.101.
49755 > 213.5.229.12.80" --color=always -A 10 | grep "POST" --color=always -A 10
reading from file 2021-02-08-exercise.pcap, link-type EN10MB (Ethernet)
2021-02-08 11:00:10.789481 IP 10.2.8.101.49755 > 213.5.229.12.80: Flags [P.], seq 1:404, ack 1, wi
n 65535, length 403: HTTP: POST /8/forum.php HTTP/1.1
E...4.@.....
..e.....[.P....J]Y.P....^..POST /8/forum.php HTTP/1.1
Accept: */*
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; Trident/7.0; rv:11.0) like Gecko
Host: satursed.com
Content-Length: 158
Cache-Control: no-cache
```

# On our second alert, strange binaries appear!

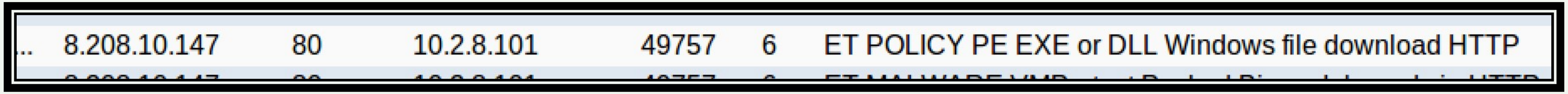


| | | | | | |
|---|---|---|---|---|---|
| 10.2.8.101 | 49757 | 8.208.10.147 | 80 | 6 | ET POLICY exe download via HTTP - Informational |

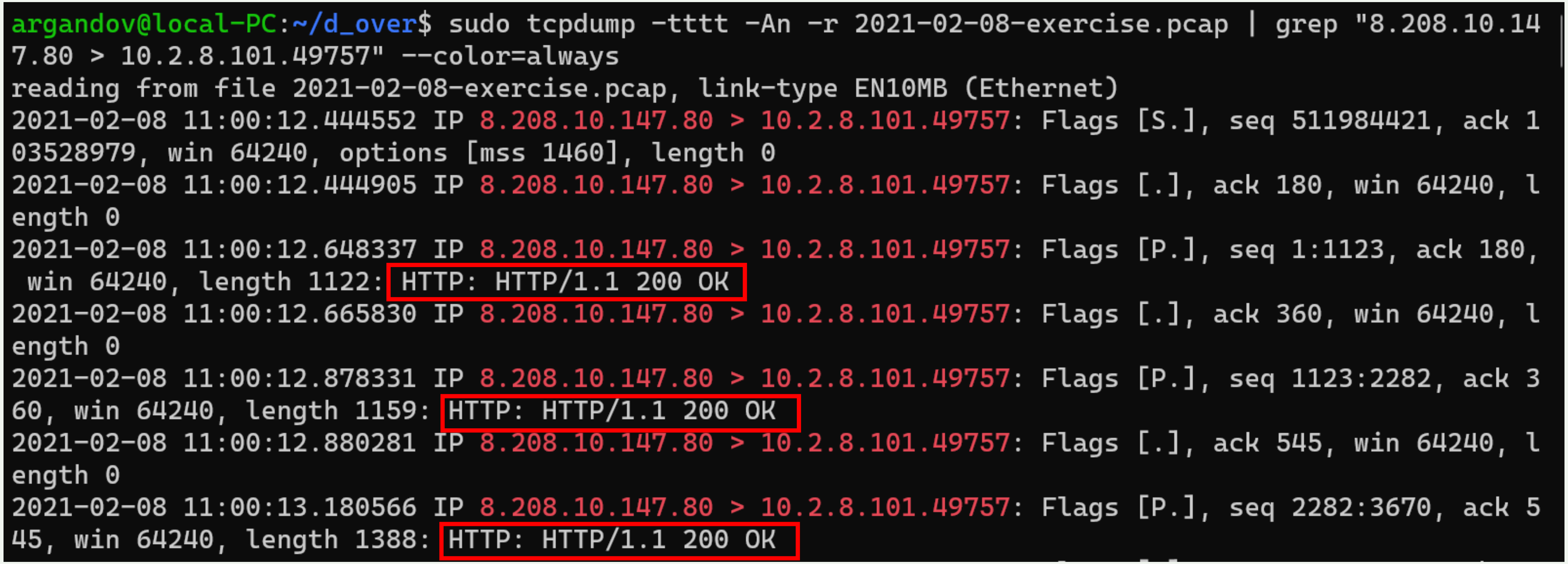


```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | grep "10.2.8.101.
49757 > 8.208.10.147.80" --color=always
reading from file 2021-02-08-exercise.pcap, link-type EN10MB (Ethernet)
2021-02-08 11:00:12.297229 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [S], seq 103528978, win 65
535, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0
2021-02-08 11:00:12.444696 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [.], ack 1, win 65535, len
gth 0
2021-02-08 11:00:12.444797 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [P.], seq 1:180, ack 1, wi
n 65535, length 179: HTTP: GET /0801.bin HTTP/1.1
2021-02-08 11:00:12.648463 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [.], ack 1123, win 65535,
length 0
2021-02-08 11:00:12.665713 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [P.], seq 180:360, ack 112
3, win 65535, length 180: HTTP: GET /0801s.bin HTTP/1.1
2021-02-08 11:00:12.878417 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [.], ack 2282, win 65535,
length 0
2021-02-08 11:00:12.880180 IP 10.2.8.101.49757 > 8.208.10.147.80: Flags [P.], seq 360:545, ack 228
2, win 65535, length 185: HTTP: GET /6lhjgfdghj.exe HTTP/1.1
```

# On our third alert, our binaries confirmed!



| | | | | | | |
|---|---|---|---|---|---|---|
| ... | 8.208.10.147 | 80 | 10.2.8.101 | 49757 | 6 | ET POLICY PE EXE or DLL Windows file download HTTP |



```
argandov@local-PC:~/d_over$ sudo tcpdump -tttt -An -r 2021-02-08-exercise.pcap | grep "8.208.10.14
7.80 > 10.2.8.101.49757" --color=always
reading from file 2021-02-08-exercise.pcap, link-type EN10MB (Ethernet)
2021-02-08 11:00:12.444552 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [S.], seq 511984421, ack 1
03528979, win 64240, options [mss 1460], length 0
2021-02-08 11:00:12.444905 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [.], ack 180, win 64240, l
ength 0
2021-02-08 11:00:12.648337 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [P.], seq 1:1123, ack 180,
 win 64240, length 1122: HTTP: HTTP/1.1 200 OK
2021-02-08 11:00:12.665830 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [.], ack 360, win 64240, l
ength 0
2021-02-08 11:00:12.878331 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [P.], seq 1123:2282, ack 3
60, win 64240, length 1159: HTTP: HTTP/1.1 200 OK
2021-02-08 11:00:12.880281 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [.], ack 545, win 64240, l
ength 0
2021-02-08 11:00:13.180566 IP 8.208.10.147.80 > 10.2.8.101.49757: Flags [P.], seq 2282:3670, ack 5
45, win 64240, length 1388: HTTP: HTTP/1.1 200 OK
```

# But… Can we retrieve those suspicious files?

## Yes we can, and should!

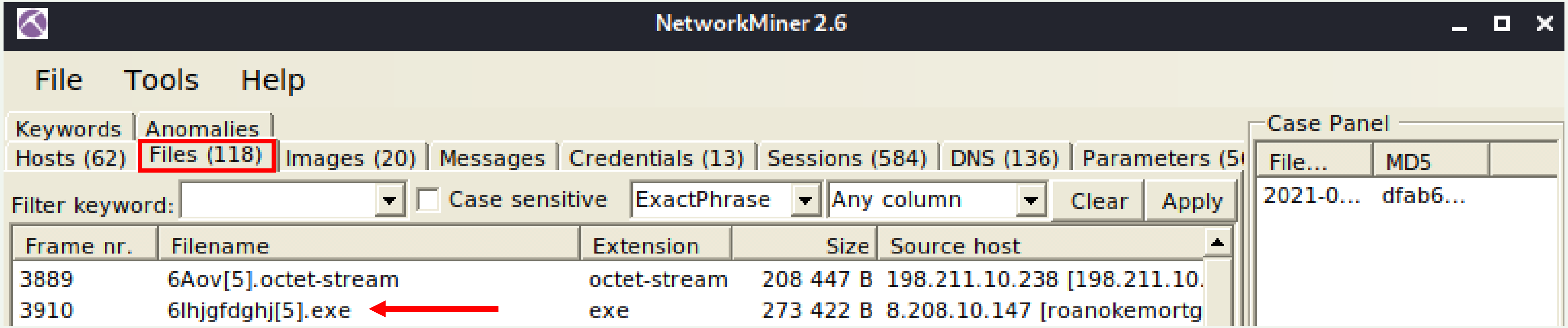
**2 ways of doing that (Of many):**

- **Network miner**
- **Wireshark**

When a malicious site downloads an .exe file into your Linux machine
YOU HAVE NO POWER HERE

**The next section was done in a Linux environment, taking some extra steps to isolate the system to do this safely.**

# Investigating the pcap file, now with some GUI



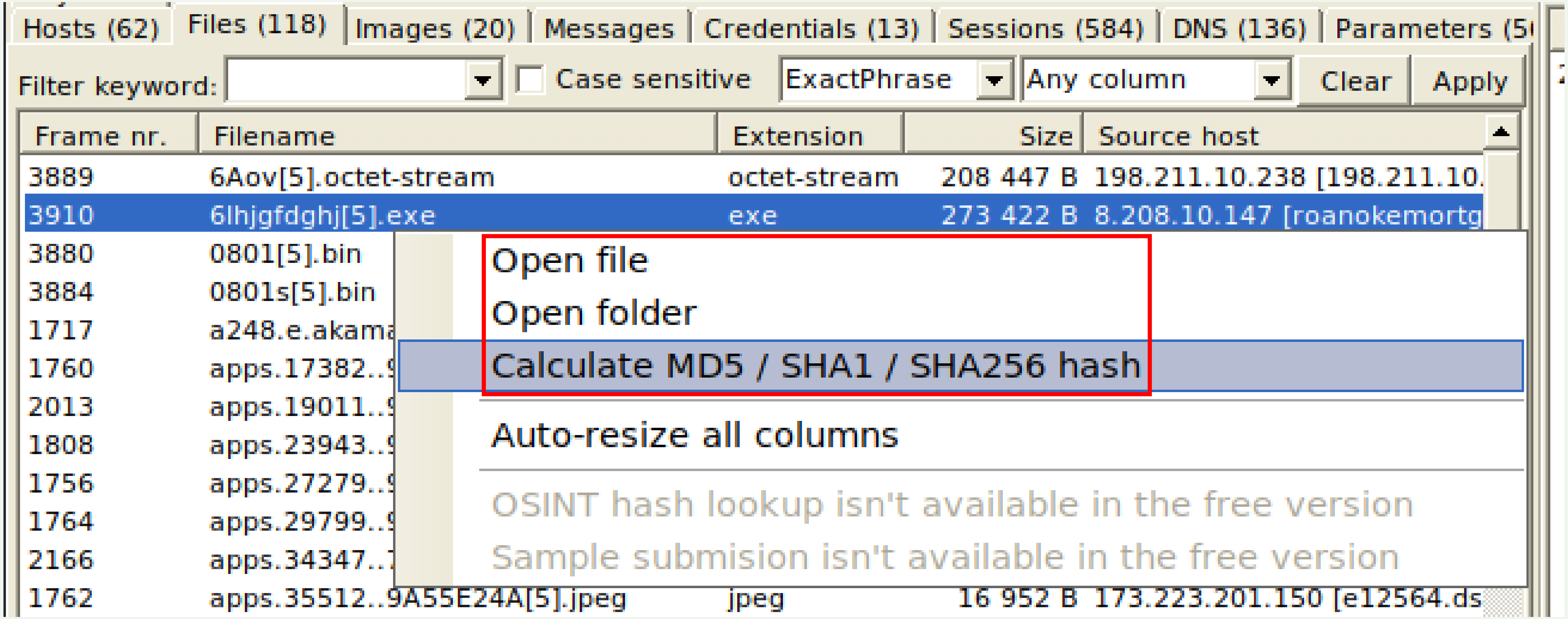
Filter keyword: ☐ Case sensitive ExactPhrase Any column

| Frame nr. | Filename | Extension | Size | Source host |
|---|---|---|---|---|
| 3889 | 6Aov[5].octet-stream | octet-stream | 208 447 B | 198.211.10.238 [198.211.10. |
| 3910 | 6lhjgfdghj[5].exe | exe | 273 422 B | 8.208.10.147 [roanokemortg |

| File... | MD5 |
|---|---|
| 2021-0... | dfab6... |

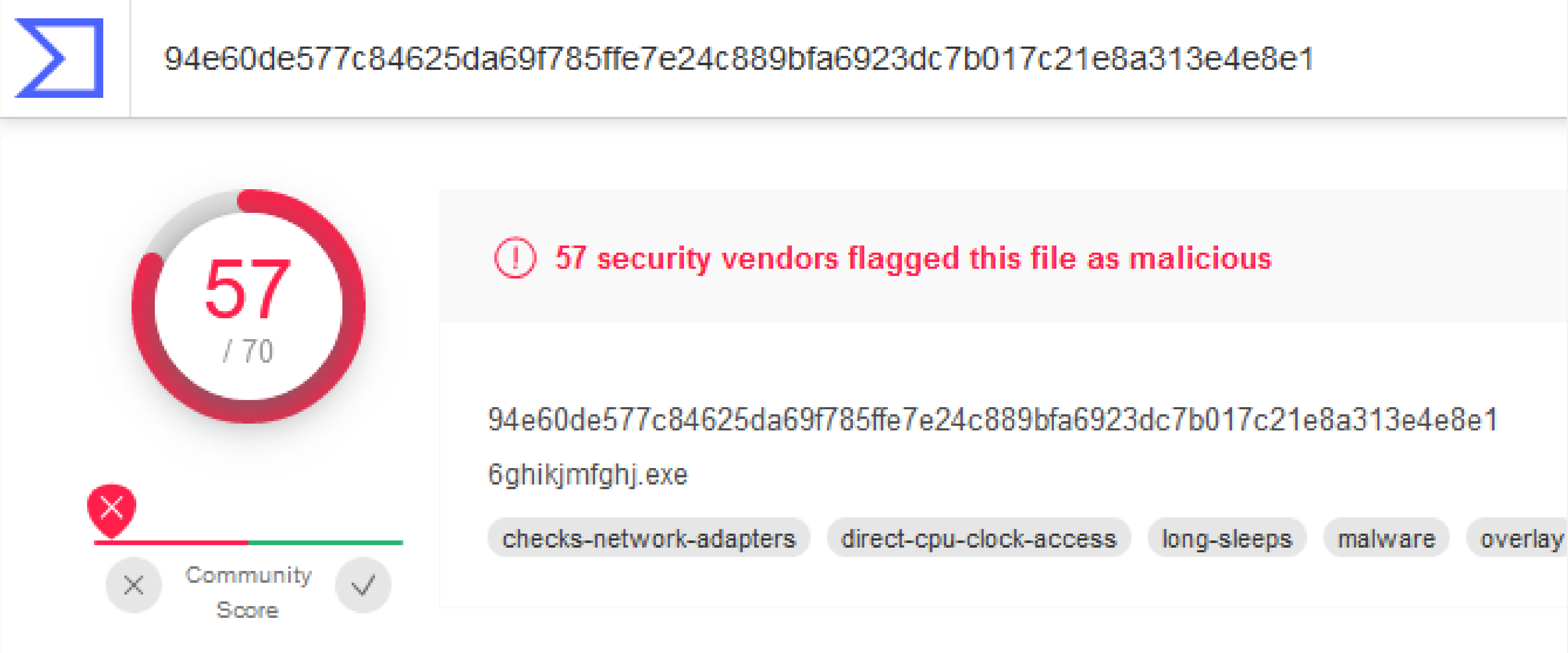
- We can now see some information, such as files transfered, DNS requests, etc. From our .pcap file.
- Something interesting showed up quite quickly.

# Extracting the files from the pcap



| Frame nr. | Filename | Extension | Size | Source host |
|---|---|---|---|---|
| 3889 | 6Aov[5].octet-stream | octet-stream | 208 447 B | 198.211.10.238 [198.211.10. |
| 3910 | 6lhjgfdghj[5].exe | exe | 273 422 B | 8.208.10.147 [roanokemortg |
| 3880 | 0801[5].bin | | | |
| 3884 | 0801s[5].bin | | | |
| 1717 | a248.e.akama | | | |
| 1760 | apps.17382..9 | | | |
| 2013 | apps.19011..9 | | | |
| 1808 | apps.23943..9 | | | |
| 1756 | apps.27279..9 | | | |
| 1764 | apps.29799..9 | | | |
| 2166 | apps.34347..7 | | | |
| 1762 | apps.35512..9A55E24A[5].jpeg | jpeg | 16 952 B | 173.223.201.150 [e12564.ds |

# Confirming our suspicions with Virus Total



94e60de577c84625da69f785ffe7e24c889bfa6923dc7b017c21e8a313e4e8e1

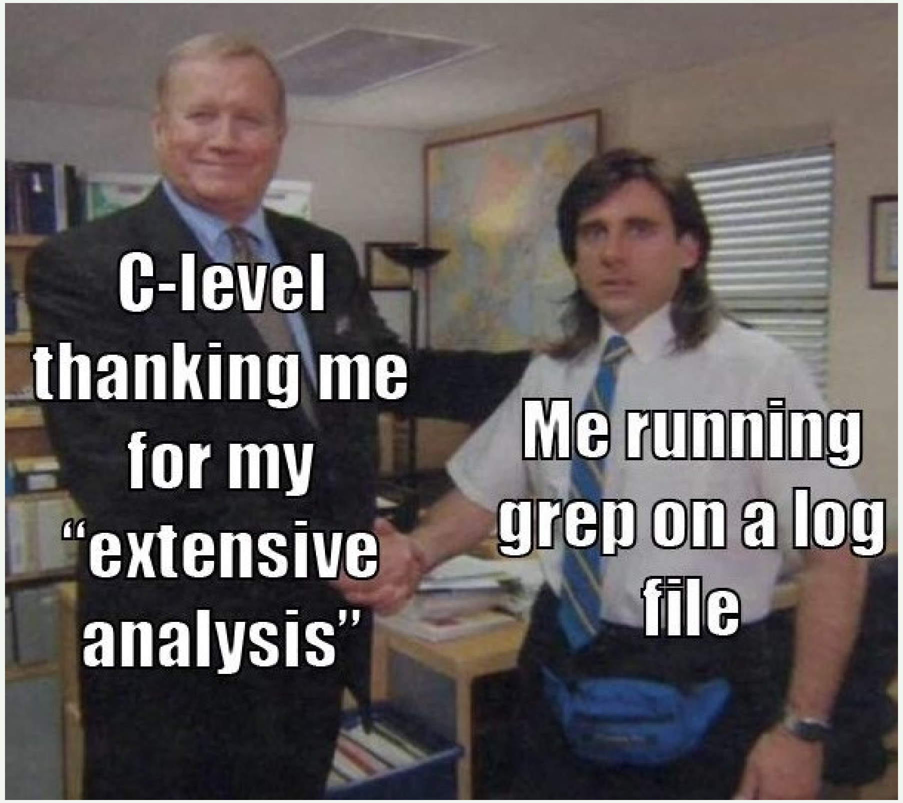
57 security vendors flagged this file as malicious

94e60de577c84625da69f785ffe7e24c889bfa6923dc7b017c21e8a313e4e8e1

6ghikjmfghj.exe

# Wait… Why all this tcpdump and CLI jazz if we have automated GUI applications?

- ➢We now know exactly how pcap files are structured.
- ➢GUI packet sniffers can be greedy in resources
- ➢GUIs are not always available


C-level thanking me for my “extensive analysis”
Me running grep on a log file

## That was easy!

- ➢You now know a couple of tools and their uses. Now let’s become proficient with them!
- ➢As a recommendation, you can follow the full exercise to prove your new knowledge.

# $ cat special_thanks.txt

Guys at Republic of Hackers. It’s a pleasure to be a part of it. You’re doing a lot for the community.

Also, the staff at Digital Overdose Con for making this event, and all your support.

# $ cat contact.txt

Download the exercise and slides:
https://github.com/Argandov/Digital_Overdose_Con_Argandov/blob/main/Resources.md

Linkedin: linkedin.com/in/Argandov

Twitter: @Argandov1

Discord: ArgandoV#0373