WIRESHARK

Case studies

Sake Blok
sake.blok@SYN-bit.nl

$ whoami

- **Relational therapist for computer systems**
  - Solve {application,network,performance}-issues by looking at the communication between systems
- **Member Wireshark core-team since 2007**
- **Started SYN-bit in 2010**
  - Application and Network troubleshooting
  - Protocol and packet analysis
  - Training (Wireshark, TCP, TLS)
- **Wireshark Certified Analyst (WCA) #1** 🎉

Some stories from real cases

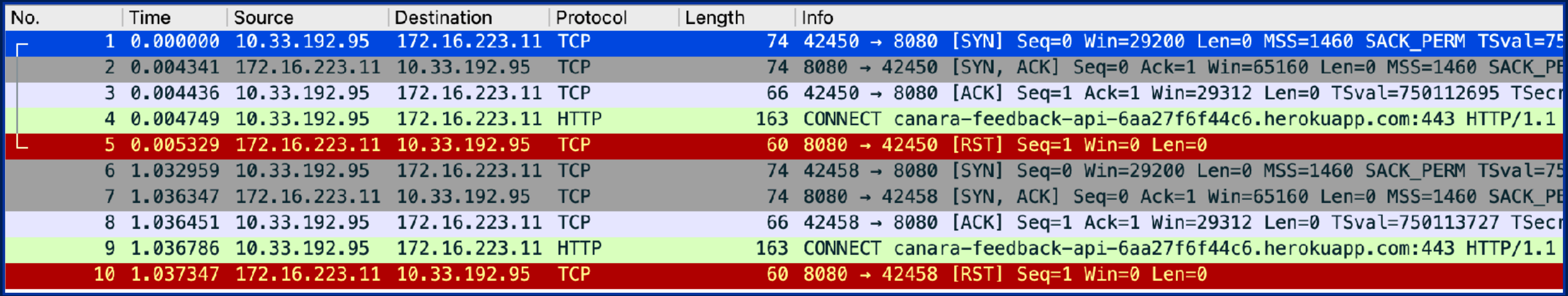
- **Cases**
  - 1: Can't reach a certain site over a proxy (ask.wireshark.org)
  - 2: Long live (video) streams get interrupted
  - 3: Retransmission on a local area without errors
  - 4: Hotel WiFi at Sharkfest '25 US
- **How to optimize Wireshark for specific tasks**
- **What information to look for as "proof"**

https://www.flickr.com/photos/lexnger/2061061452

Case 01: Question on TCP/RST

- Question about "RST: present, Fin: Absent..."
  - Actually comes from tcp.completeness

- But (why) is the session reset by the proxy?



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.33.192.95	172.16.223.11	TCP	74	42450 → 8080 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=75
2	0.004341	172.16.223.11	10.33.192.95	TCP	74	8080 → 42450 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PE
3	0.004436	10.33.192.95	172.16.223.11	TCP	66	42450 → 8080 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=750112695 TSecr
4	0.004749	10.33.192.95	172.16.223.11	HTTP	163	CONNECT canara-feedback-api-6aa27f6f44c6.herokuapp.com:443 HTTP/1.1
5	0.005329	172.16.223.11	10.33.192.95	TCP	60	8080 → 42450 [RST] Seq=1 Win=0 Len=0
6	1.032959	10.33.192.95	172.16.223.11	TCP	74	42458 → 8080 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=75
7	1.036347	172.16.223.11	10.33.192.95	TCP	74	8080 → 42458 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PE
8	1.036451	10.33.192.95	172.16.223.11	TCP	66	42458 → 8080 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=750113727 TSecr
9	1.036786	10.33.192.95	172.16.223.11	HTTP	163	CONNECT canara-feedback-api-6aa27f6f44c6.herokuapp.com:443 HTTP/1.1
10	1.037347	172.16.223.11	10.33.192.95	TCP	60	8080 → 42458 [RST] Seq=1 Win=0 Len=0

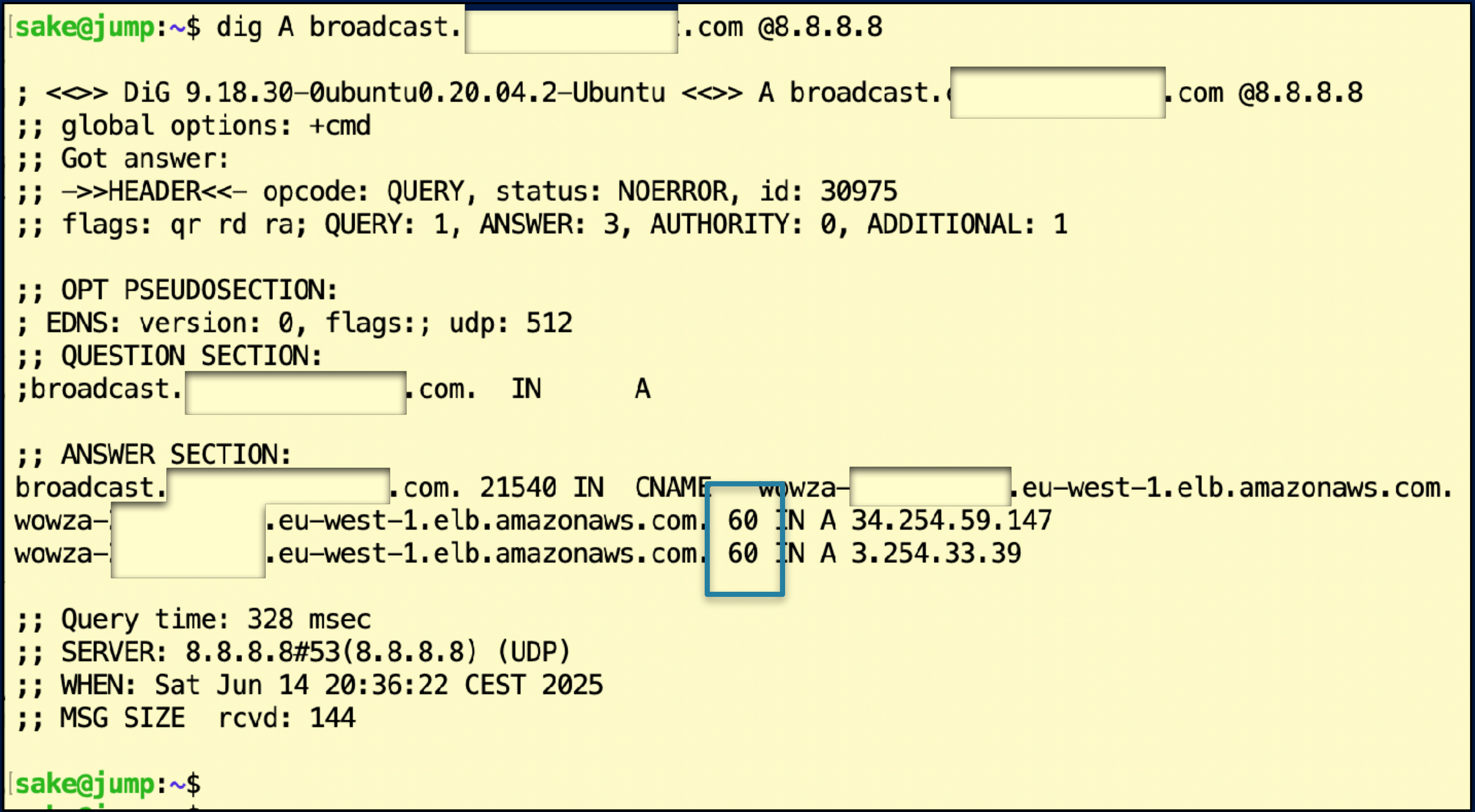
LIVE DEMO

Case 01: Resolution & tips

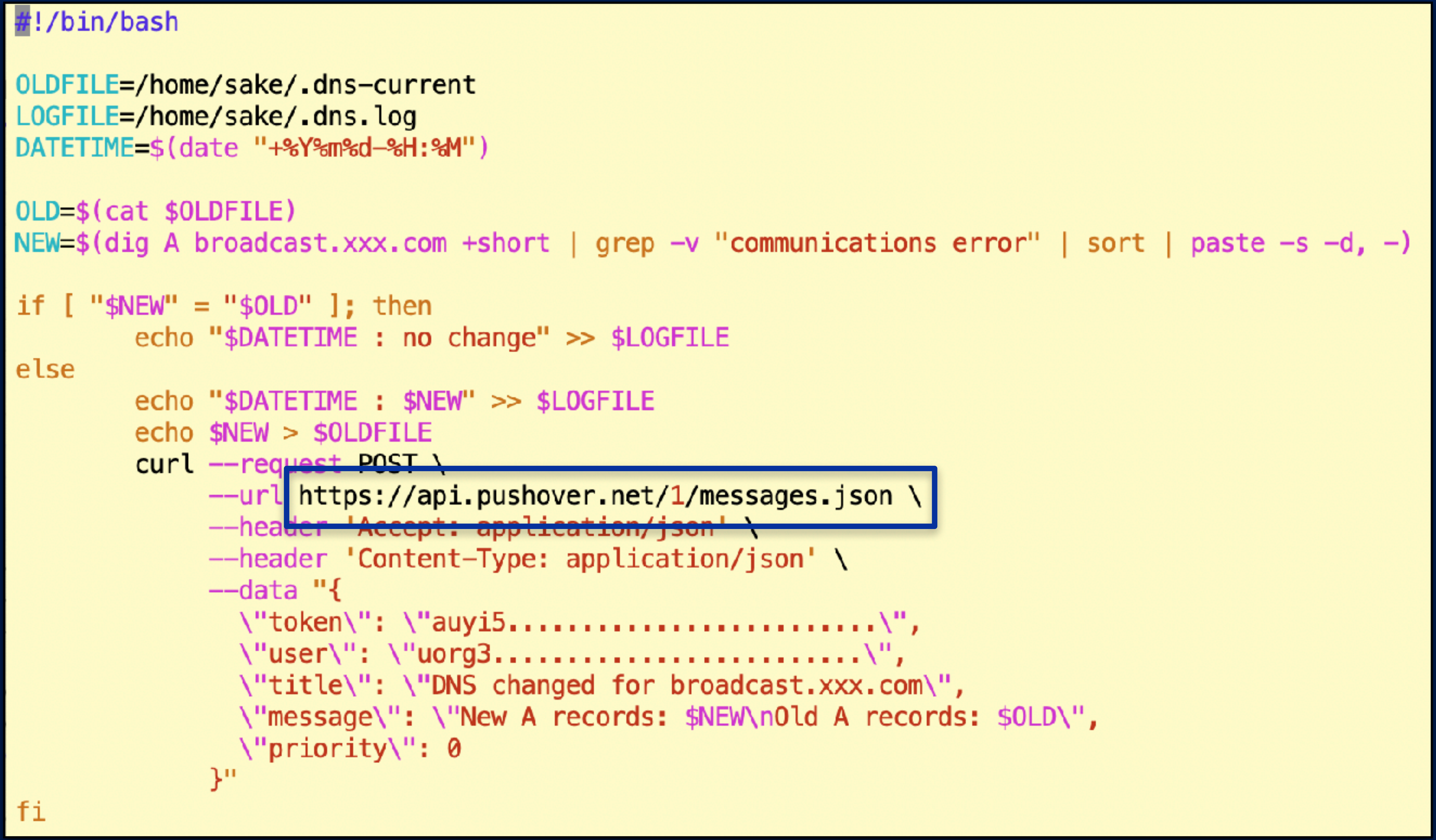
- Firewall has URL filtering enabled and blocked the website
- Add columns of interest, makes life a lot easier
- Timing of packets is important
  - Can things happen at the time they did?
  - Take note of the iRTT and where the capture was taken

Case 02: Broken video streams

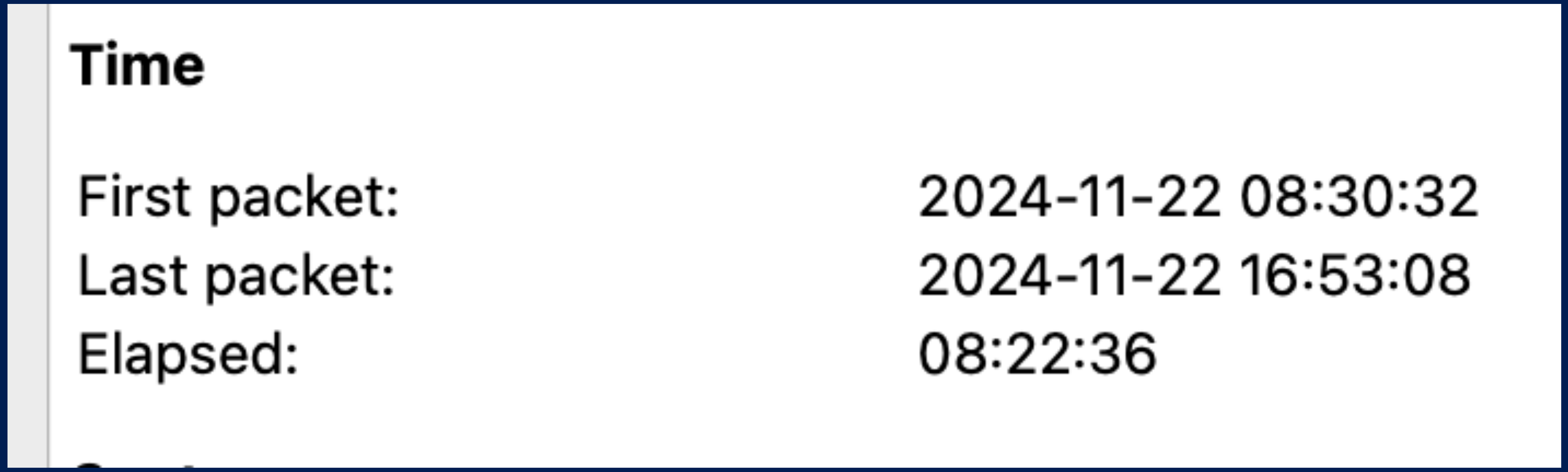
- **Live streaming of city council hearings**
- **Streams break**
  - Usually in long hearings
  - Mostly at ten to the hour (like ~22:50)
- **Cause of issue not clear**
  - client blames streaming provider
  - streaming provider blames clients network
  - Suspicion of DNS involvement
- **TCP sessions of up to 14 hours long**
  - 10-30 GB each! A total of 800 GB was captured
  - 10-100 million packets each



```
sake@jump:~$ dig A broadcast.[redacted].com @8.8.8.8

; <<>> DiG 9.18.30-0ubuntu0.20.04.2-Ubuntu <<>> A broadcast.[redacted].com @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30975
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;broadcast.[redacted].com.   IN      A

;; ANSWER SECTION:
broadcast.[redacted].com. 21540 IN  CNAME   wowza-[redacted].eu-west-1.elb.amazonaws.com.
wowza-[redacted].eu-west-1.elb.amazonaws.com. 60 IN A 34.254.59.147
wowza-[redacted].eu-west-1.elb.amazonaws.com. 60 IN A 3.254.33.39

;; Query time: 328 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Sat Jun 14 20:36:22 CEST 2025
;; MSG SIZE  rcvd: 144

sake@jump:~$
```

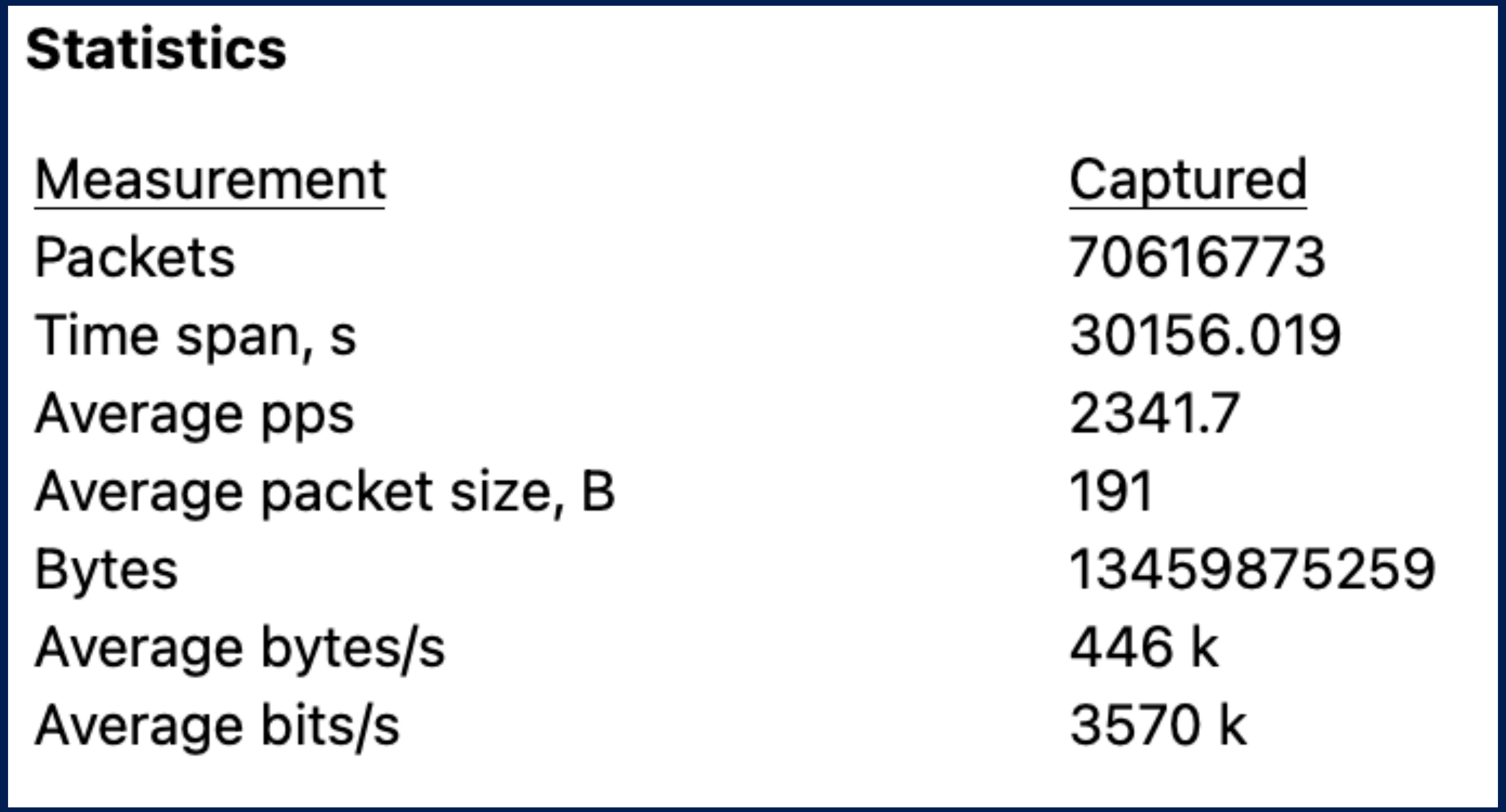


```
#!/bin/bash

OLDFILE=/home/sake/.dns-current
LOGFILE=/home/sake/.dns.log
DATETIME=$(date "+%Y%m%d-%H:%M")

OLD=$(cat $OLDFILE)
NEW=$(dig A broadcast.xxx.com +short | grep -v "communications error" | sort | paste -s -d, -)

if [ "$NEW" = "$OLD" ]; then
        echo "$DATETIME : no change" >> $LOGFILE
else
        echo "$DATETIME : $NEW" >> $LOGFILE
        echo $NEW > $OLDFILE
        curl --request POST \
             --url https://api.pushover.net/1/messages.json \
             --header 'Accept: application/json' \
             --header 'Content-Type: application/json' \
             --data "{
               \"token\": \"auyi5.........................\",
               \"user\": \"uorg3.........................\",
               \"title\": \"DNS changed for broadcast.xxx.com\",
               \"message\": \"New A records: $NEW\nOld A records: $OLD\",
               \"priority\": 0
             }"
fi
```



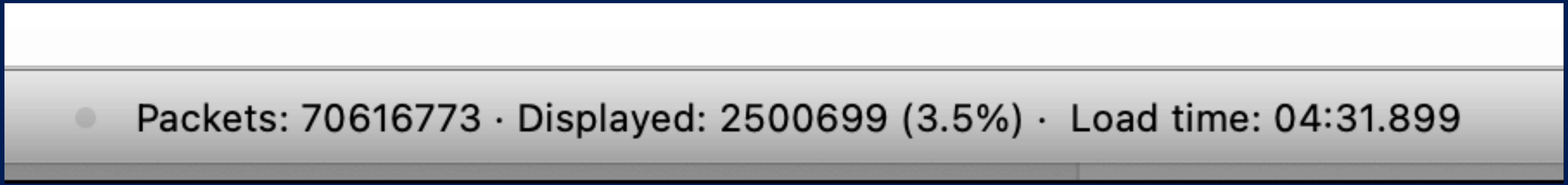
Time

First packet:	2024-11-22 08:30:32
Last packet:	2024-11-22 16:53:08
Elapsed:	08:22:36

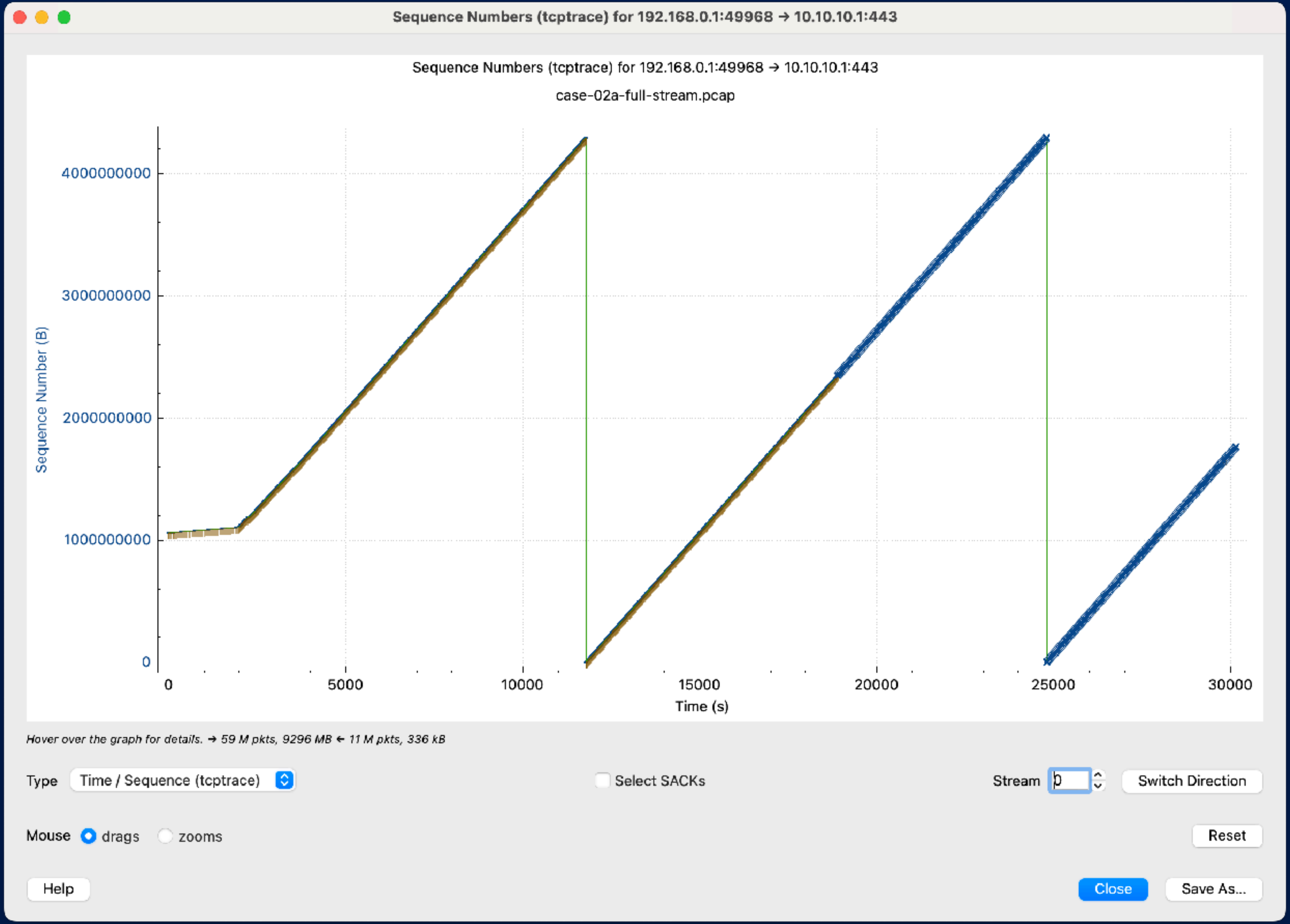


Statistics

Measurement	Captured
Packets	70616773
Time span, s	30156.019
Average pps	2341.7
Average packet size, B	191
Bytes	13459875259
Average bytes/s	446 k
Average bits/s	3570 k



Packets: 70616773 · Displayed: 2500699 (3.5%) · Load time: 04:31.899



Sequence Numbers (tcptrace) for 192.168.0.1:49968 → 10.10.10.1:443

Sequence Numbers (tcptrace) for 192.168.0.1:49968 → 10.10.10.1:443

case-02a-full-stream.pcap

Hover over the graph for details. → 59 M pkts, 9296 MB ← 11 M pkts, 336 kB

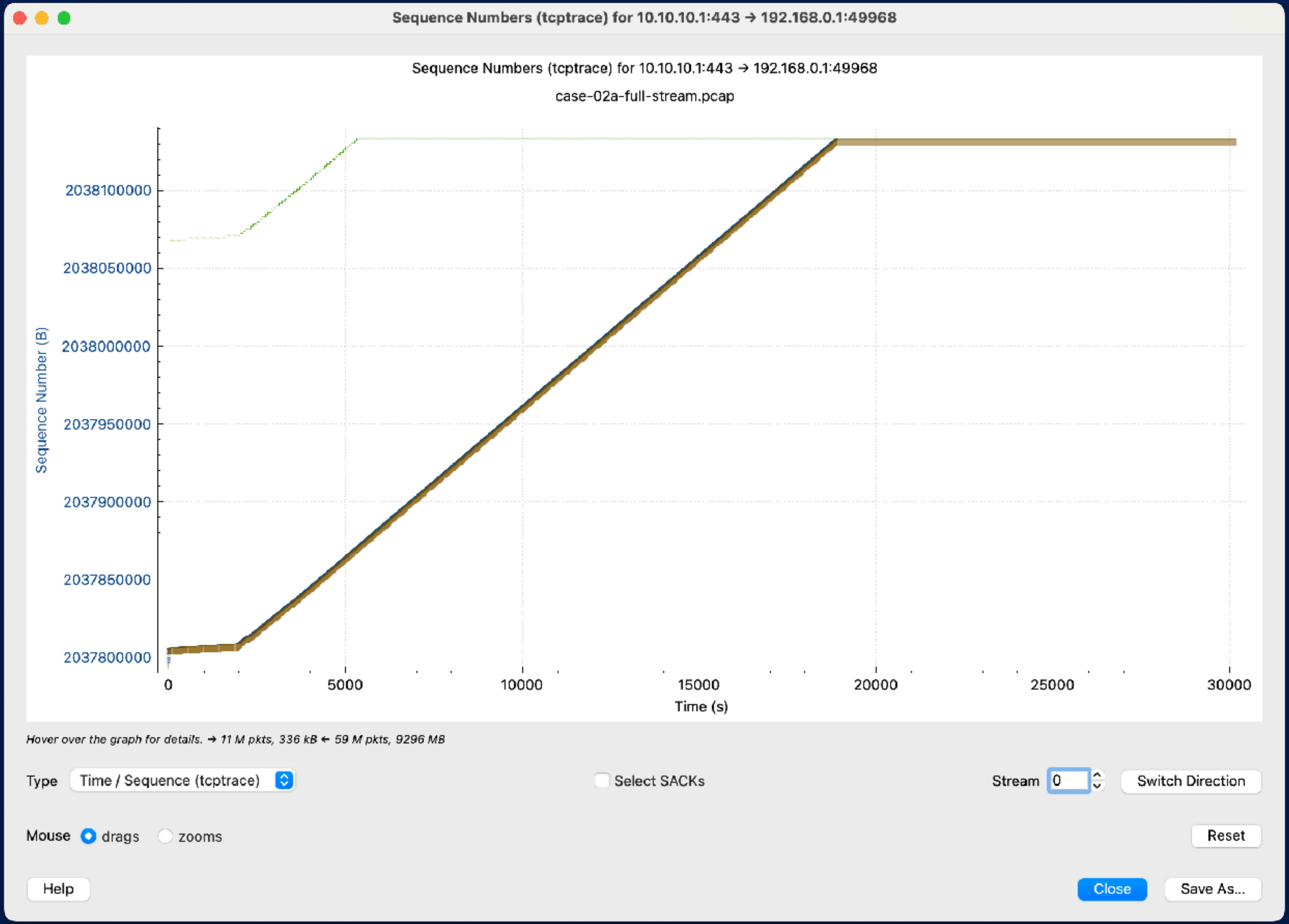
Type Time / Sequence (tcptrace)

□ Select SACKs

Stream 0

Mouse ● drags ○ zooms

LIVE DEMO



Sequence Numbers (tcptrace) for 10.10.10.1:443 → 192.168.0.1:49968

Sequence Numbers (tcptrace) for 10.10.10.1:443 → 192.168.0.1:49968

case-02a-full-stream.pcap

Hover over the graph for details. → 11 M pkts, 336 kB ← 59 M pkts, 9296 MB

Type: Time / Sequence (tcptrace)

□ Select SACKs

Stream: 0

Mouse: ● drags ○ zooms

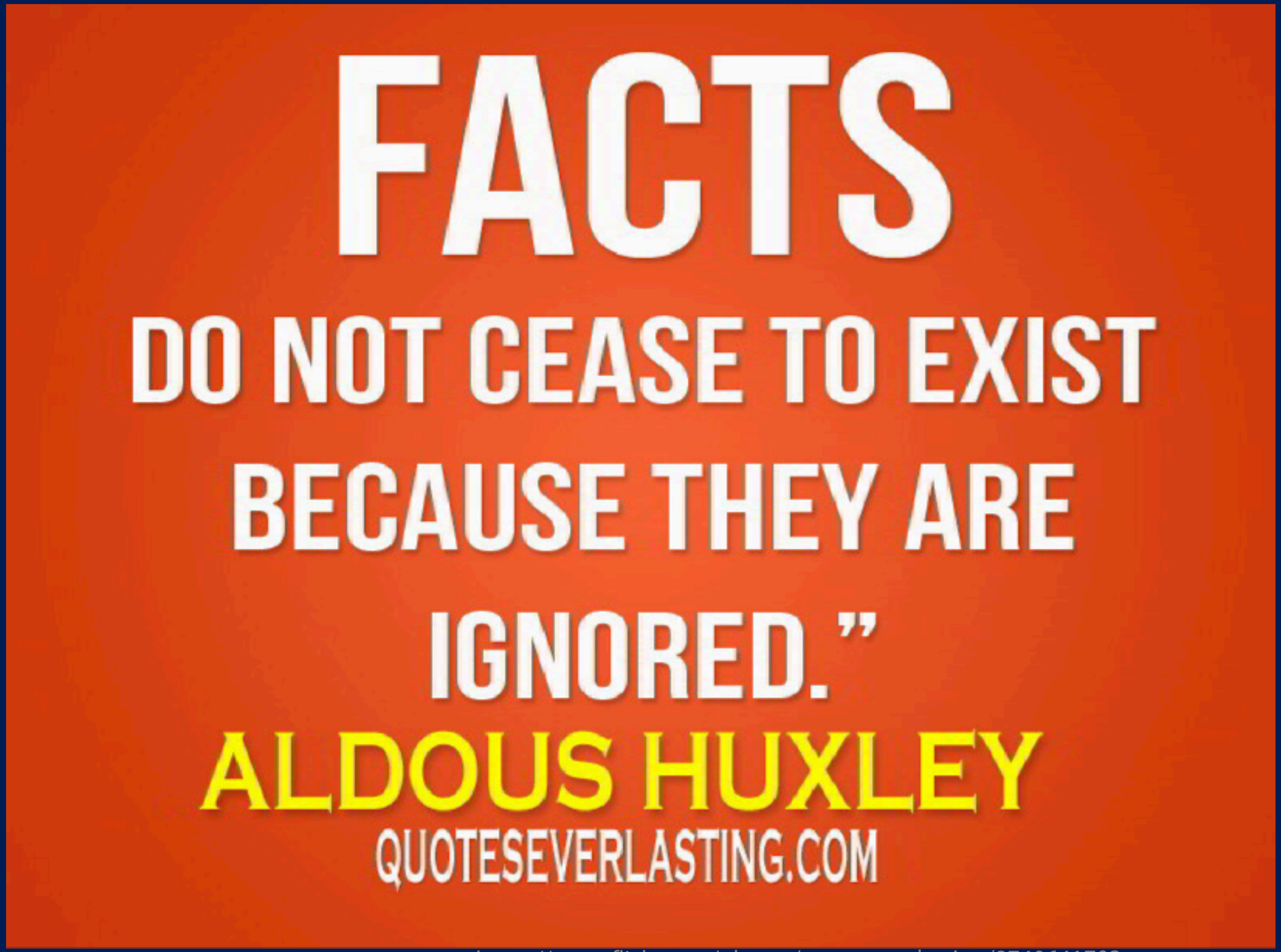
LIVE DEMO

Real time	Source	Info
09:30:03,999	CwcEncodingManager.log	INFO CWCEncodingManager.WebcastManager - Time: server = 11/22/2024 08:30:04, encoder = 11/22/2024 09:30:00, difference = 00:59:56.6088423 (0.99905801175h)
09:30:10,492	H264EncodingService.log	INFO Cwc.H264EncodingService.Models.Writer - Writer starting C:\Program Files\xxx\CwcEncodingService\archive\xxx_20241122_1-part0.ts
09:30:11,147	H264EncodingService.log	INFO Cwc.H264EncodingService.Models.Writer - Writer starting rtmps://broadcast.xxx.com:443/live/CWCENCODER-xxx/<UUID>
09:30:12,182	PCAP	DNS: broadcast.xxx.com => 34.241.39.86, 54.155.185.97
09:30:12,185	PCAP	TCP: SYN 10.x.x.x:57525 -> 34.241.39.86:443, MSS=1460/1460, WS=256/256
09:30:13,259	H264EncodingService.log	INFO Cwc.H264EncodingService.Models.Writer - Starttime 2024-11-22T08:30:12.2880000Z
10:57:06,351	PCAP	Last window-size increase (1020 -> 1026, ie 261120 -> 262656)
10:57:20,912	PCAP	Start final window-size decline to 0 (1026 -> 1025, ie 262656 -> 262400)
15:45:00,000	DNS-check	DNS changed to 52.212.68.153,63.34.179.176

Real time	Source	Info
16:01:57,402	PCAP	Last data from server
16:01:57,419	PCAP	Start zero-window condition in client
18:18:12,486	H264EncodingService.log	WARN Cwc.H264EncodingService.Models.Writer - @Gap 500ms.
18:18:13,126	H264EncodingService.log	INFO Cwc.H264EncodingService.Models.Writer - NewTotalReportedOffset: 1000
22:52:50,136	PCAP	Last ACK from the server
22:52:50,386	PCAP	First retransmissions from the client
22:52:51,134	H264EncodingService.log	WARN Cwc.H264EncodingService.Models.Writer - @Gap 1083ms.
22:52:52,635	H264EncodingService.log	INFO Cwc.H264EncodingService.Models.Writer - NewTotalReportedOffset: 2000
22:52:59,687	PCAP	Last retransmission from the client
22:53:07,640	H264EncodingService.log	WARN Cwc.H264EncodingService.Models.Writer - @Stream interrupted, restarting. Possible causes: bad connection or invalid api key.
22:53:09,287	PCAP	TCP/RST from the client

Case 02: facts so far...

- Analysed 16 full video streams
- Only broken when DNS change during stream
  - But DNS change not always breaks the stream
  - The stream breaks long time after DNS change
- Failing retransmissions -> RST
- Firewall (Fortigate) rules based on FQDN
  - temporarily changing 1 test encoder to "ALL"
  - problem does not occur for that encoder
- Still puzzled by the "ten to the hour" timing
- Is this a BUG or a FEATURE (config issue!)?



FACTS DO NOT CEASE TO EXIST BECAUSE THEY ARE IGNORED."

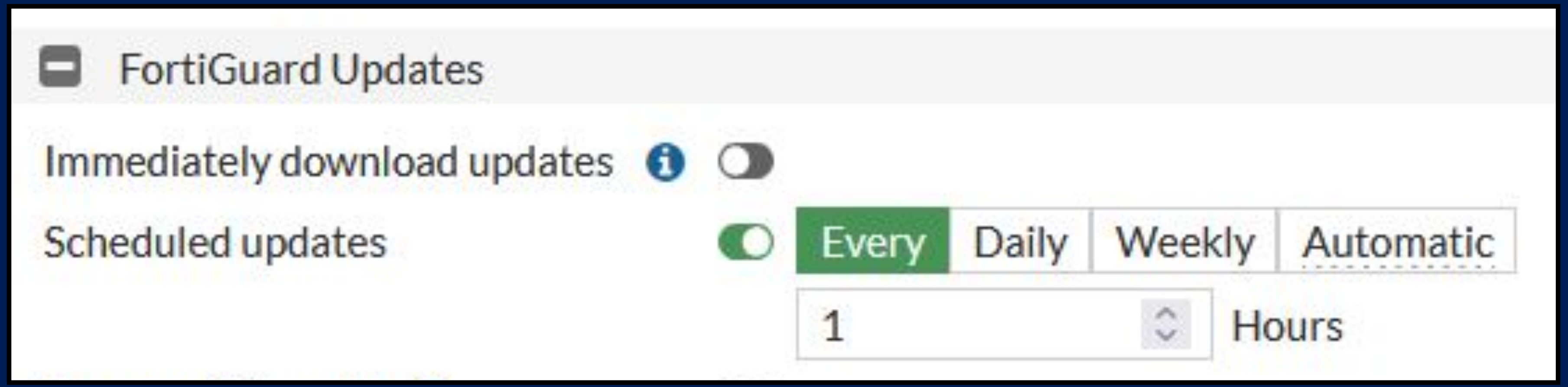
ALDOUS HUXLEY

QUOTESEVERLASTING.COM

https://www.flickr.com/photos/quoteseverlasting/8740641703

Case 02: ... the resolution!

- **May-dirty?**
  - https://community.fortinet.com/t5/FortiGate/Technical-Tip-Dirty-session/ta-p/197748
  - https://community.fortinet.com/t5/FortiGate/Technical-Tip-Explanation-of-the-FQDN-nbsp-default-nbsp-cache/ta-p/213280
  - https://community.fortinet.com/t5/FortiGate/Technical-Tip-Information-about-firewall-session-dirty/ta-p/195802
- **Feature to re-evaluate the policy rules under certain conditions**
  - FortiGuard update being one of them
  - Configured hourly update interval at...
    - ... "ten to the hour" !!!
- **So when...**
  - There was a DNS change during the video stream...
  - ... and an update to the FortigGuard files...
  - ... the session is marked "dirty" and the policy rule gets re-evaluated...
  - ... and because the IP address does not match the ones in DNS...
  - ... the packets are dropped, causing retransmissions... and finally a TCP/RST

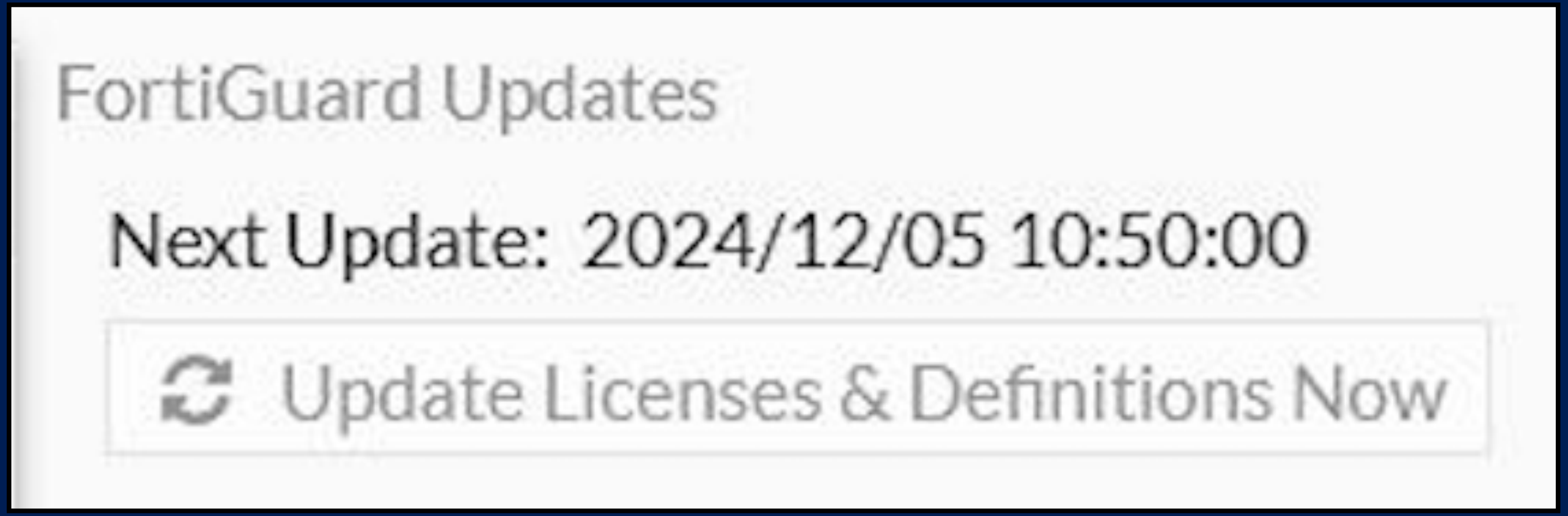


FortiGuard Updates

Immediately download updates ○

Scheduled updates ● Every | Daily | Weekly | Automatic

1 Hours



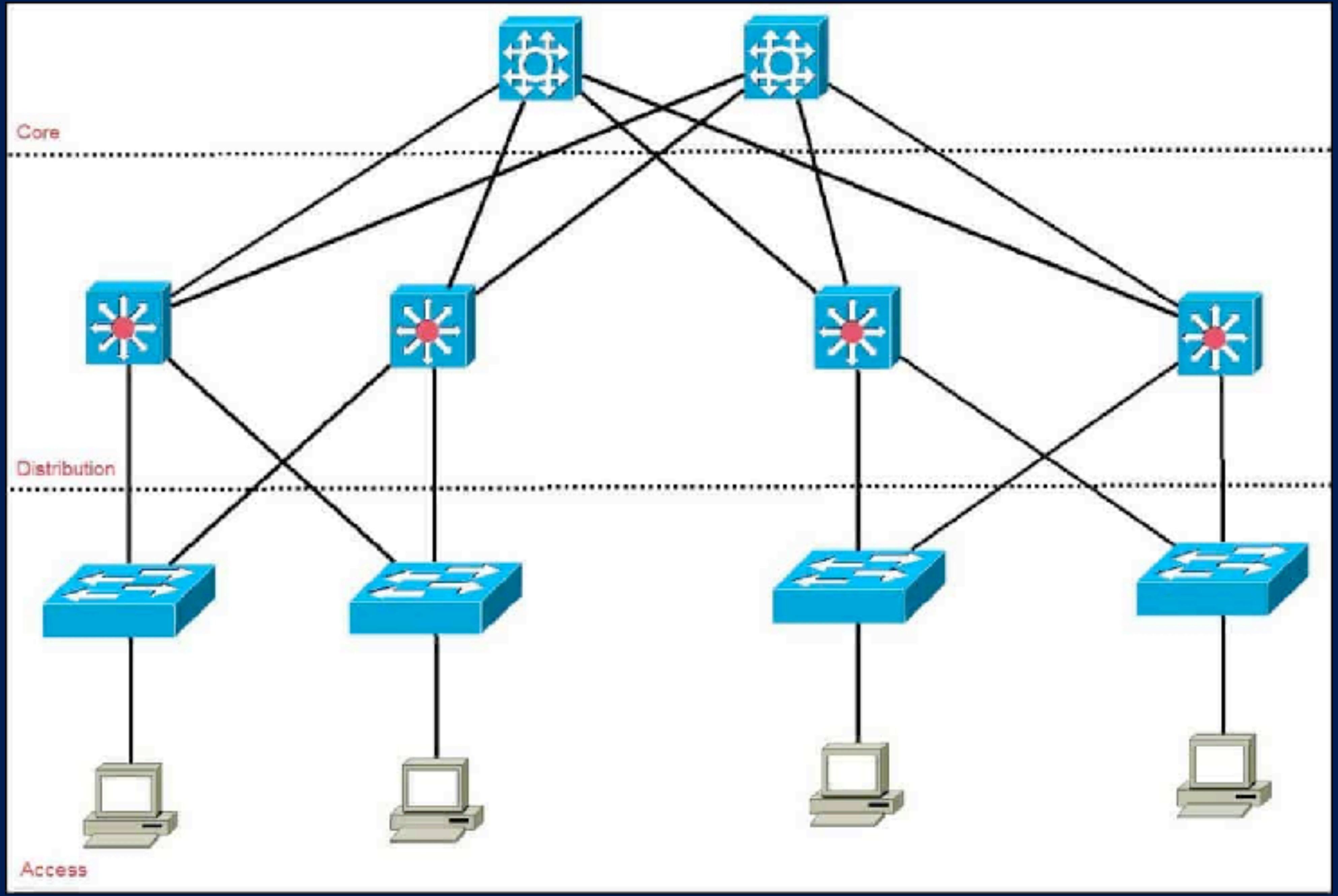
FortiGuard Updates

Next Update: 2024/12/05 10:50:00

Case 03: TCP retransmissions

- Two servers are connected to different server switches
- The two server switches use the same distribution switch
- The distribution switch is connected to the core switch that provides the routing function in this network

- *Many retransmissions are detected on the servers (TCP stats) the source of the retransmissions needs to be found*



Case 03: Packet Hero Quiz Time

- **Which of the following statements is true (choose one):**

  A. The packet with sequence number 49 (frame 4) was received out of order by host 10.0.0.1

  B. Frame 14 is a retransmission of frames 4, 5, 6, 7, 8 and 11
  because these frames were not received by host 10.0.0.1

  C. Frame 4 was dropped by an intermediate network device, therefor frames 5, 6 and 7 generated 3
  duplicate ACKs that triggered the fast-retransmission in frame 14

  D. The ICMP redirect in frame 16 was caused by a wrongly configured subnet-mask on host 10.0.0.2

LIVE DEMO

Case 03: True or False

- **Which of the following statements is true (choose one):**

A. ✅ The packet with sequence number 49 (frame 4) was received out of order by host 10.0.0.1

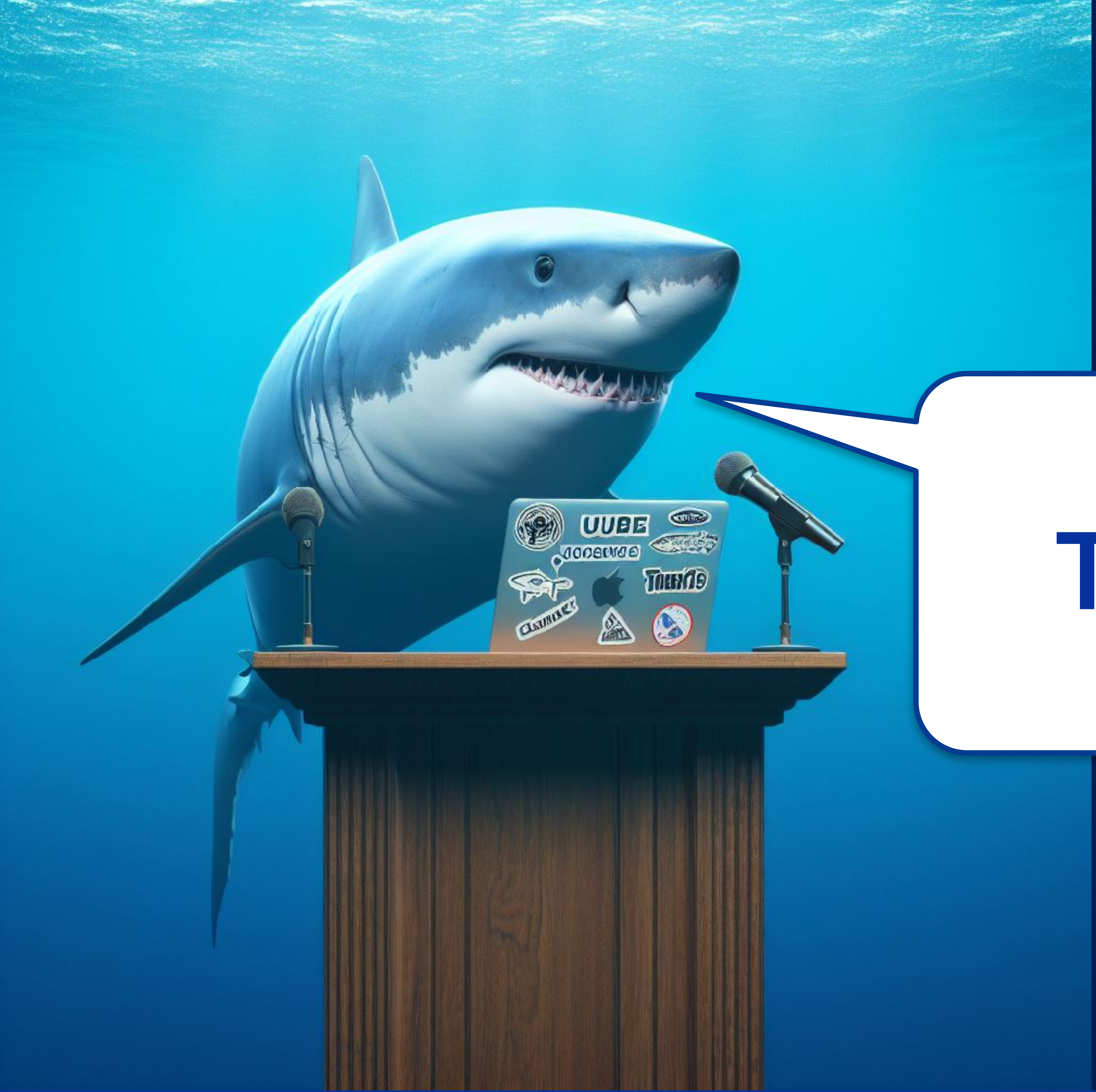
B. ✅ Frame 14 is a retransmission of frames 4, 5, 6, 7, 8 and 11
❌ because these frames were not received by host 10.0.0.1

C. ❌ Frame 4 was dropped by an intermediate network device, therefor
✅ Frames 5, 6 and 7 generated 3 duplicate ACKs that triggered the fast-retransmission in frame 14

D. ❌ The ICMP redirect in frame 16 was caused by a wrongly configured subnet-mask on host 10.0.0.2

Case 03: Resolution & tips

- Static (host) routes on all systems, even though in the same subnet
    - Lazy standardised deployment scenario
- Each second, 1 ICMP redirect message
    - packet is routed over the CPU to generate the ICMP message
    - Process switched packets are slower (1 ms!)
- Use the right columns
- Use (temporary) coloring
- ICMP is your friend
- Using the ip.id field in troubleshooting can help
    - But beware of the different ip.id numbering strategies



Time for Q & A

Still questions?
sake.blok@SYN-bit.nl

SYN-bit
deep traffic analysis