A12 - (Not So) False Positives in Application Performance Analysis

Christian Landström, Senior Consultant
Airbus Defence and Space

Agenda

- Introduction into false positives
- Analysis Cases
  - “Slow browser performance”
  - Strange packet loss on infrastructure
- This session is going to be 90% hands-on and live analysis
- There are USB-Sticks available with the traces for interactive participation
- Please hand them back after the session ;)

Case #1: Browser-based Downloads

- Users complain about slow downloads
- Sample loads captured using two browsers:
  - Internet Explorer
  - Firefox

- Captured on local client machines, sample file to download provided

→ Basic TCP / HTTP Analysis

What to look for:

- Round-Trip-Time (RTT)
- TCP Retransmissions
- Window Size Problems
- Application Response Time
- ACK Timings
- Other findings

Hands-on: Trace file analysis

- Compare the two samples
  - Browser IE.pcap
  - Browser FF.pcap

Conclusion

- Retransmissions! 400+ packets being lost in such short time have a serious impact on performance
- Zero Window in one session for +3 seconds

  → Regular interpretation: Somethings wrong within the client at that point
- Real cause: “Save as” dialogue in that specific browser version
- No performance problem at the client side
- Roughly **equal** performance in both traces

Case #2: High Packet Drop Rate

- Using TCP traces ONPW
  - Analyse I/O Graph -> Perfect 100MBit straight
  - Hint that there is no Root cause
  - Analyse retransmissions (don’t seem to have effect)
  - -> Show Timimgs until Retrans @Client
  - -> Ask for explaination, Hint at broken Fast Retrans
  - -> Show Timimgs until Retrans @Server
  - -> Zoom in to 1msec IO -> show Microbursts
  - Conclude to always compare capture points

Case #2: High Packet Drop Rate

- Switch statistics show high rate of packet drops
- Performance on User side seemingly not affected too much
- Analysis goal: Clarify if and why there is packet loss

Hands-on: Trace file analysis

- Analyze the two given trace files:
  - “Uplink”, taken at the route down from the distribution layer towards the access layer switches
  - “Client”, taken locally at a lab client during the test runs

Client File Analysis:

- I/O shows perfect steady 100MBit/s
- 42 Retransmissions

  → Obviously no impact on performance

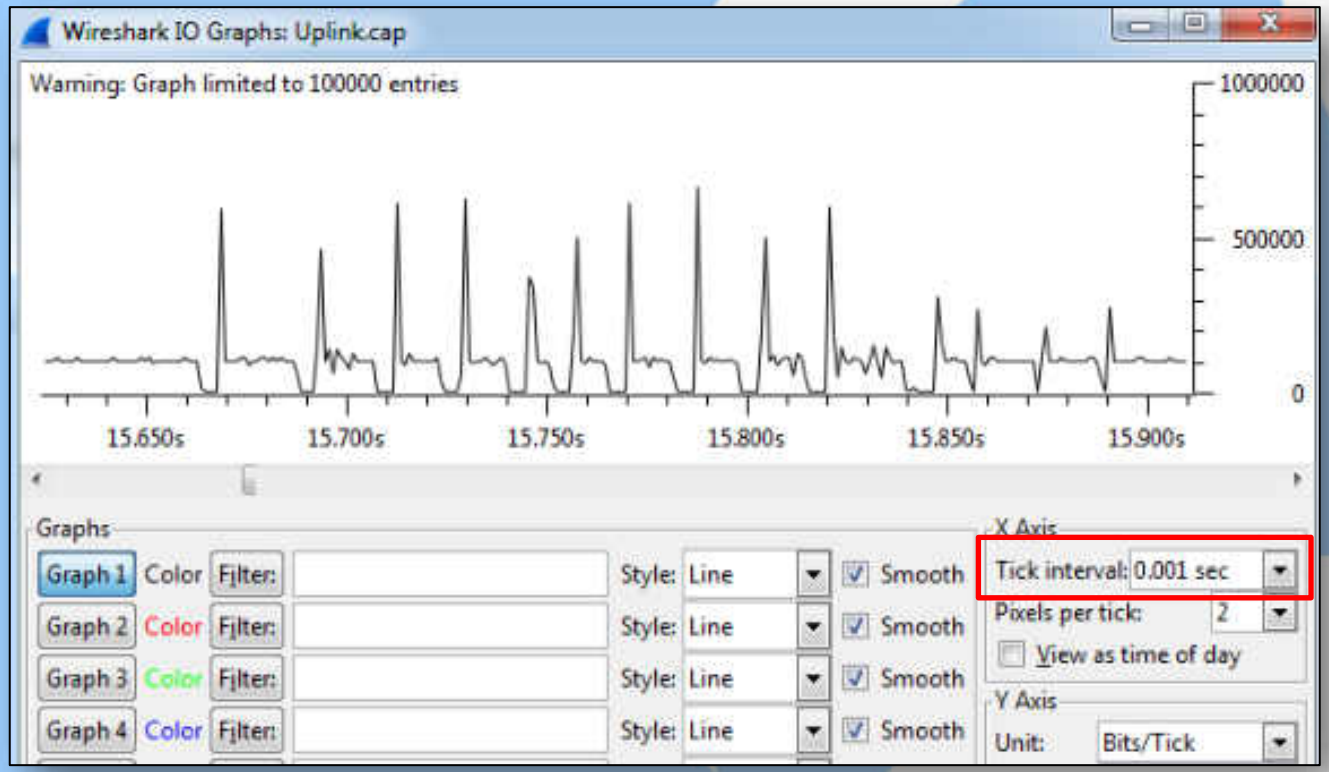
Uplink File Analysis:

- I/O shows perfect steady 100MBit/s
- 44 Retransmissions
  - → Obviously no impact on performance
  - → Really?

Let's take a deeper look at what happened here…

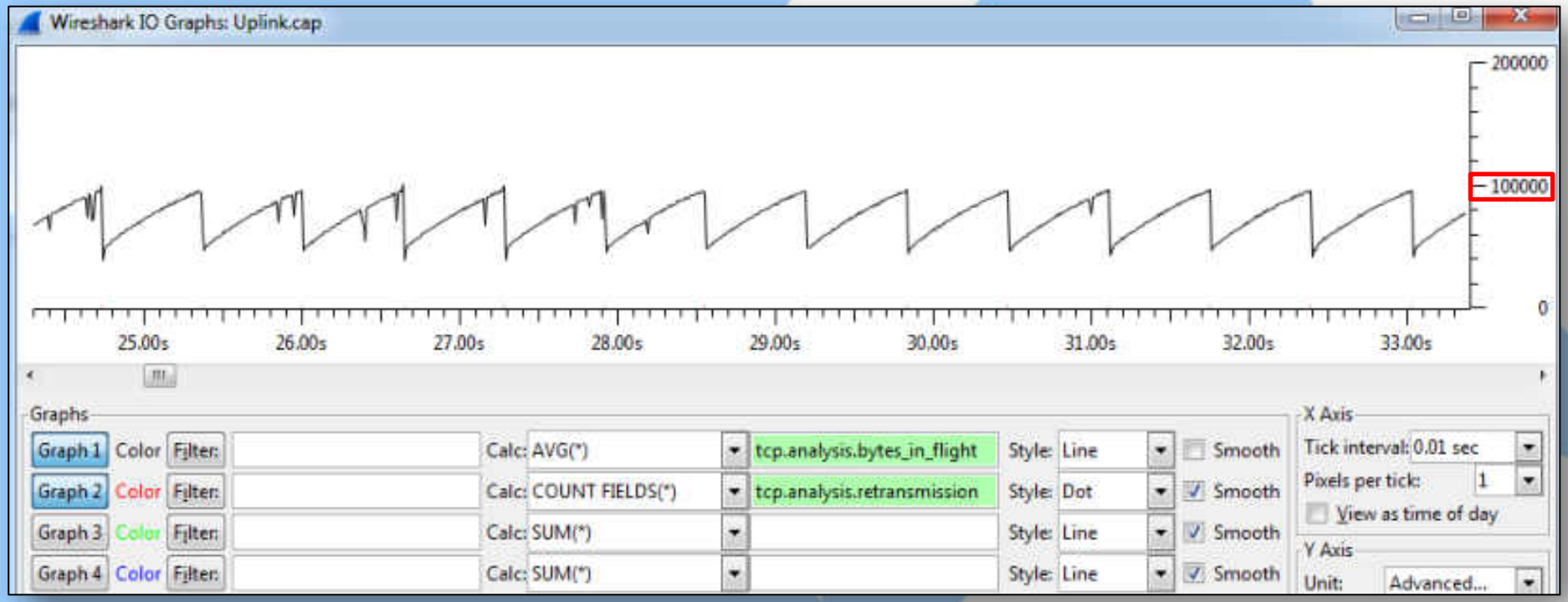
Uplink File Analysis:

- Change the time scale within I/O Graph to 1msec



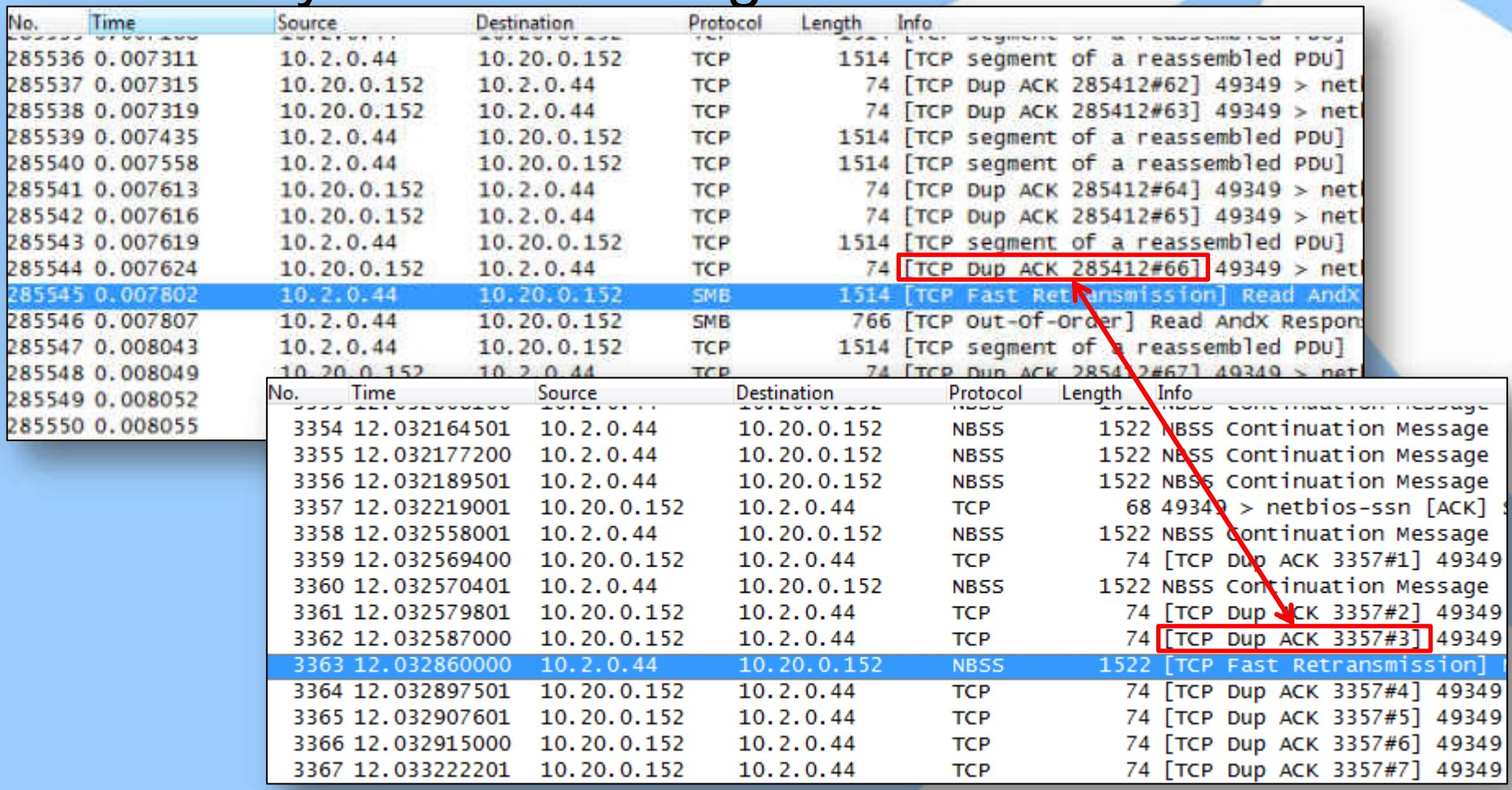
Uplink File Analysis:

- Verify with bytes in flight:
- Packet loss around 100k bytes in flight



Uplink File Analysis:

- Verify with the timings until fast retransmit

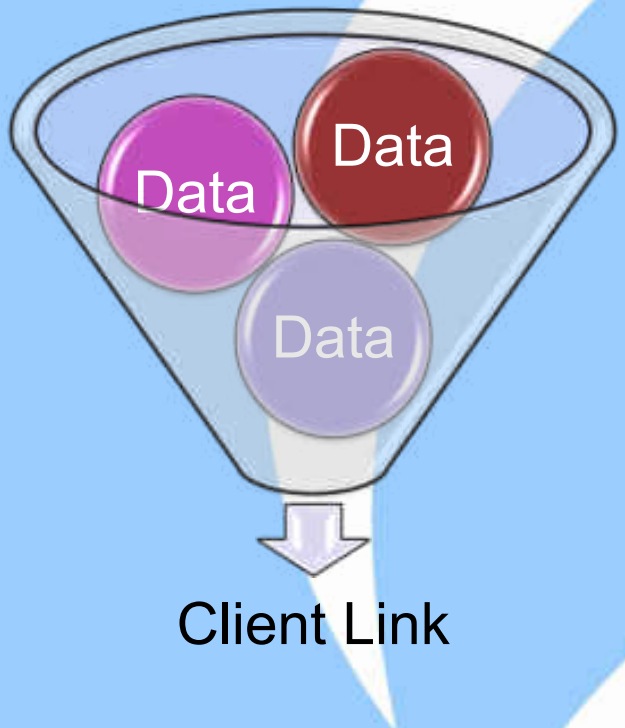


No.	Time	Source	Destination	Protocol	Length	Info
285536	0.007311	10.2.0.44	10.20.0.152	TCP	1514	[TCP segment of a reassembled PDU]
285537	0.007315	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#62] 49349 > net
285538	0.007319	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#63] 49349 > net
285539	0.007435	10.2.0.44	10.20.0.152	TCP	1514	[TCP segment of a reassembled PDU]
285540	0.007558	10.2.0.44	10.20.0.152	TCP	1514	[TCP segment of a reassembled PDU]
285541	0.007613	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#64] 49349 > net
285542	0.007616	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#65] 49349 > net
285543	0.007619	10.2.0.44	10.20.0.152	TCP	1514	[TCP segment of a reassembled PDU]
285544	0.007624	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#66] 49349 > net
285545	0.007802	10.2.0.44	10.20.0.152	SMB	1514	[TCP Fast Retransmission] Read AndX
285546	0.007807	10.2.0.44	10.20.0.152	SMB	766	[TCP Out-Of-Order] Read AndX Respon
285547	0.008043	10.2.0.44	10.20.0.152	TCP	1514	[TCP segment of a reassembled PDU]
285548	0.008049	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 285412#67] 49349 > net
285549	0.008052					
285550	0.008055					

No.	Time	Source	Destination	Protocol	Length	Info
3354	12.032164501	10.2.0.44	10.20.0.152	NBSS	1522	NBSS Continuation Message
3355	12.032177200	10.2.0.44	10.20.0.152	NBSS	1522	NBSS Continuation Message
3356	12.032189501	10.2.0.44	10.20.0.152	NBSS	1522	NBSS Continuation Message
3357	12.032219001	10.20.0.152	10.2.0.44	TCP	68	49349 > netbios-ssn [ACK]
3358	12.032558001	10.2.0.44	10.20.0.152	NBSS	1522	NBSS Continuation Message
3359	12.032569400	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#1] 49349
3360	12.032570401	10.2.0.44	10.20.0.152	NBSS	1522	NBSS Continuation Message
3361	12.032579801	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#2] 49349
3362	12.032587000	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#3] 49349
3363	12.032860000	10.2.0.44	10.20.0.152	NBSS	1522	[TCP Fast Retransmission]
3364	12.032897501	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#4] 49349
3365	12.032907601	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#5] 49349
3366	12.032915000	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#6] 49349
3367	12.033222201	10.20.0.152	10.2.0.44	TCP	74	[TCP Dup ACK 3357#7] 49349

Uplink File Analysis:

- Switch buffers filling constantly until overflow
- Data arriving at higher bandwidth leading to packet loss



Conclusion:

- No obvious “performance problem” visible
- Yet infrastructure is overloaded and high packet loss is expected when multiple clients simultaneously request data
- Too big receive window leading to buffer bloating within switching infrastructure

End of story:

- Watch for user interaction inside trace files
  - Might explain strange network behavior
  - Name your traces accordingly to remember when user interaction was involved
- Double-Check every finding either by yourself or even better with a co-worker
- Think out of the box, the most obvious findings are not necessarily the correct ones
- And of course… Keep learning!

!! Thank you for your attention !!

Q / A...