

# T2-6: Trace File Analysis - The Elephant Coming From Behind: Full Window, Window Update and TCP Keep-Alive's

**Laura Chappell**

Founder | Wireshark University

**Betty DuBois**

Principal Consultant | DuBois Training & Consulting, LLC

**SHARKFEST '08**

Foothill College

March 31 - April 2, 2008

# Agenda

## Using Wireshark to isolate a TCP Window issue

- Expert Info
  - Errors
  - Warnings
  - Notes
    - Zero Window
    - Window Update
    - Keep Alive
    - Window Full
  - Chat
- Time Column
  - Delta
  - Delta Displayed

# Best Practices for Protocol Analysis

Onsite v. offsite analysis

Create a baseline when performance is acceptable

Analyze application traffic before deployment (capacity planning)

Troubleshooting Tips:

- Who complained?
- Begin as close to the user as possible
- Name captures appropriately (sue1, sue2, sue3mac, etc.)
- Move analyzer as needed or use multiple analyzers and agents
- Time-sync if using multiple analyzers
- Have taps/hubs in place for when the need arises
- Focus on the time column (delta time setting)
- Consider command-line capture (nmcap/tshark)

Security Tips:

- Baseline protocols, applications, traffic patterns
- Examine summary and protocol information for anomalies
- Look for signatures in questionable traffic
- Snort website has many signatures in the rule sets

# Expert Info – Errors

**Error (red)**: serious problem, e.g. [Malformed Packet]

- **Malformed**: malformed packet or dissector has a bug, dissection of this packet aborted

Expert Errors do NOT always mean a network or communication error, they may indicate a dissector error

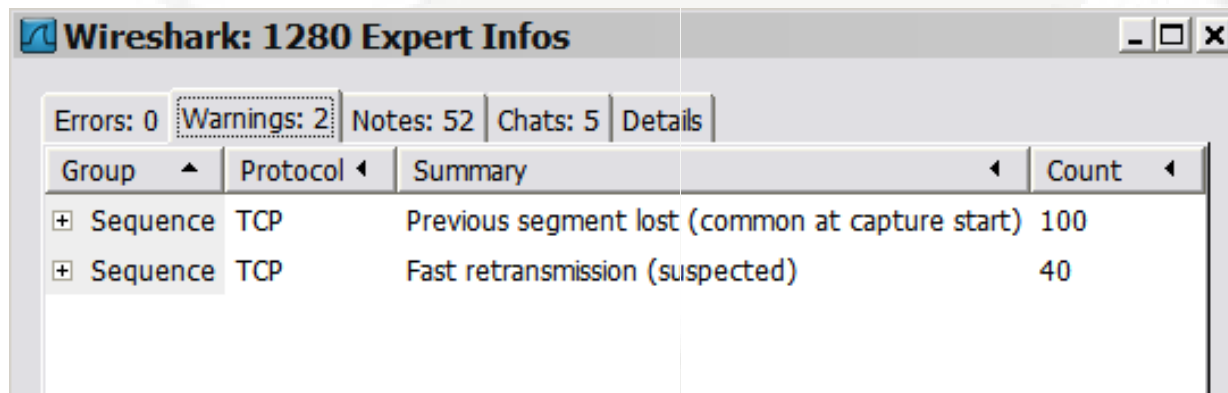
# Expert Info – Warnings

**Warn (yellow):** warning, e.g. application returned an "unusual" error code like a connection problem

Are the fast retransmissions the issue?

- Calculate percentage of retransmitted vs. normal packets.
- Over 1% error ratio merits further investigation. That is not the case here.

Attend Session T2-8 for further info on Retransmissions and Fast Retransmissions



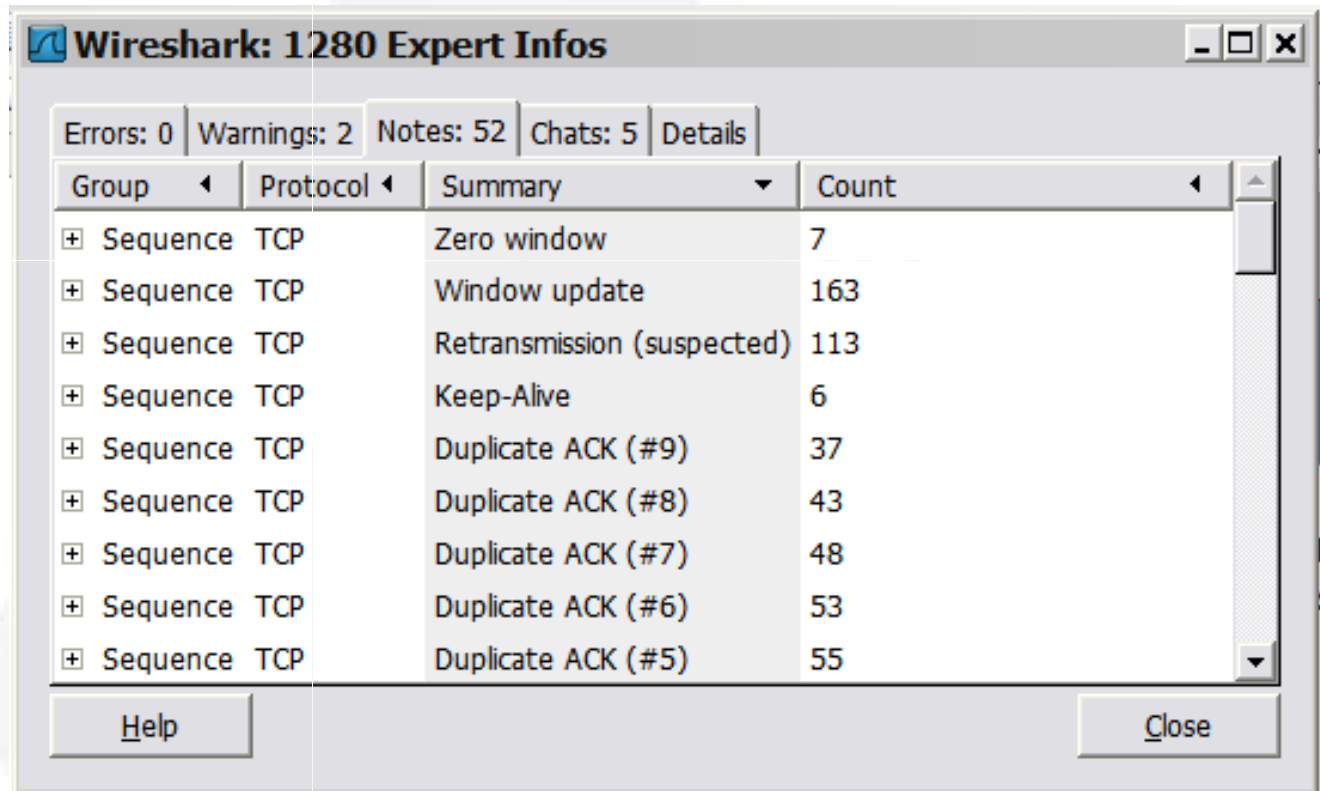
The screenshot shows the Wireshark Expert Info window with the following data:

Group	Protocol	Summary	Count	
+	Sequence	TCP	Previous segment lost (common at capture start)	100
+	Sequence	TCP	Fast retransmission (suspected)	40

# Expert Info – Notes

**Note (cyan):** notable things, e.g. an application returned an "usual" error code like HTTP 404

- Zero window
- Window update
- Keep-Alive
- Duplicate ACK (Covered in Session T2-8)



Wireshark: 1280 Expert Infos

Errors: 0 | Warnings: 2 | Notes: 52 | Chats: 5 | Details

Group	Protocol	Summary	Count
+	Sequence TCP	Zero window	7
+	Sequence TCP	Window update	163
+	Sequence TCP	Retransmission (suspected)	113
+	Sequence TCP	Keep-Alive	6
+	Sequence TCP	Duplicate ACK (#9)	37
+	Sequence TCP	Duplicate ACK (#8)	43
+	Sequence TCP	Duplicate ACK (#7)	48
+	Sequence TCP	Duplicate ACK (#6)	53
+	Sequence TCP	Duplicate ACK (#5)	55

Help Close

# Expert Info – Notes – Zero Window

The key to determining when a Zero Window is truly an issue, is how long does it stay at zero before updating.

The example below shows a host processing its data and updating the TCP window very quickly.

No. -	Delta	Source	Destination	Protocol	Bytes	Info
97	0.000034	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
98	0.000033	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
99	0.000038	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
100	0.000035	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
101	0.000043	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
102	0.000035	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
103	0.000159	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
104	0.000225	192.168.1.112	192.168.1.103	TCP	60	[TCP Zerowindow] 1939 > 2707
105	0.009293	192.168.1.112	192.168.1.103	TCP	60	[TCP window update] 1939 > 2707
106	0.000071	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110054
107	0.000060	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110068
108	0.000052	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110082

# Expert Info – Notes – Window Update

Wireshark creates an Expert Note whenever the TCP window size increases.

Warning: Wireshark does NOT create the Note when the window size is decreasing.

No. -	Delta	Source	Destination	Protocol	Bytes	Info
97	0.000034	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
98	0.000033	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
99	0.000038	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
100	0.000035	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
101	0.000043	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
102	0.000035	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
103	0.000159	192.168.1.112	192.168.1.103	TCP	60	1939 > 2707 [ACK] Seq=14227937
104	0.000235	192.168.1.112	192.168.1.103	TCP	60	[TCP Zerowindow] 1939 > 2707
105	0.009293	192.168.1.112	192.168.1.103	TCP	60	[TCP window update] 1939 > 2707
106	0.000071	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110054
107	0.000060	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110068
108	0.000052	192.168.1.103	192.168.1.112	TCP	1514	2707 > 1939 [ACK] Seq=27110082



# Expert Info – Notes – Keep-Alive

Normal Keep-Alive's should be sent at regular intervals when no data is being transferred to keep the session alive.

In this trace, what do you notice about the Delta times?

No.	Delta ^	Source	Destination	Protocol	Bytes	Info
375	16.074234	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
373	8.064197	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
371	4.128068	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
369	2.198012	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
367	1.133508	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
5966	0.681690	madheifer.pacific.net.au	10.0.52.164	HTTP	1514	[TCP Retransmission] Continuation
365	0.667083	madheifer.pacific.net.au	10.0.52.164	TCP	60	[TCP Keep-Alive] http > 2550 [ACK]
370	0.270205	madheifer.pacific.net.au	10.0.52.164	HTTP	1514	Continuation

# Timings

Using the Time columns makes isolating faults much more efficient.

- **Delta** previous packet may not be seen
- **Delta Displayed** only care about seen packets

No. -	Time	Delta	Delta Disp	Source	Destination	Info
1	0.000000	0.000000	0.000000	192.168.15.106	Broadcast	who has 192.168.15.1? Te
2	0.002606	0.002606	0.002606	Cisco-Li_25:8c:6b	192.168.15.106	192.168.15.1 is at 00:13:
9036	275.426286	0.588873	275.423680	192.168.15.106	Broadcast	who has 192.168.15.1? Te
9037	275.430991	0.004705	0.004705	Cisco-Li_25:8c:6b	192.168.15.106	192.168.15.1 is at 00:13:
9065	706.289888	17.864596	430.858897	192.168.15.106	Broadcast	who has 192.168.15.1? Te
9066	706.292966	0.003078	0.003078	Cisco-Li_25:8c:6b	192.168.15.106	192.168.15.1 is at 00:13:
9067	730.401840	24.108874	24.108874	192.168.15.106	Broadcast	who has 192.168.15.1? Te
9068	730.405763	0.003923	0.003923	Cisco-Li_25:8c:6b	192.168.15.106	192.168.15.1 is at 00:13:
10808	1005.932605	8.860180	275.526842	192.168.15.106	Broadcast	who has 192.168.15.104?
10817	1014.878299	2.517608	8.945694	192.168.15.106	Broadcast	who has 192.168.15.104?
12464	1364.444775	2.468330	349.566476	192.168.15.106	Broadcast	who has 192.168.15.104?
12476	1373.474585	0.748050	9.029810	192.168.15.106	Broadcast	who has 192.168.15.104?

# Receiver Congestion

SEQ 3000 – 1460 bytes of data



SEQ 2000 – 0 bytes of data; ACK 4460, Win=5840



SEQ 4460 – 1460 bytes of data



SEQ 5920 – 1460 bytes of data



SEQ 7380 – 1460 bytes of data



SEQ 2000 – 0 bytes of data; ACK 5920, Win=4380



SEQ 2000 – 0 bytes of data; ACK 7380, Win=2920



SEQ 8840 – 1460 bytes of data



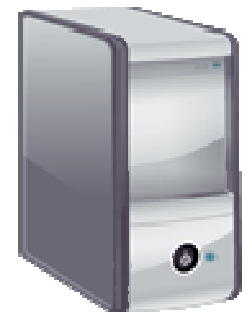
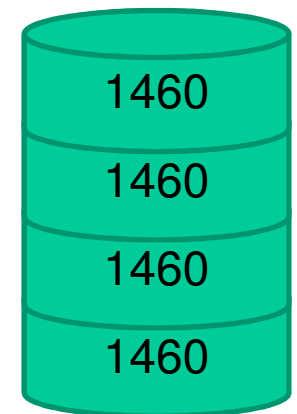
SEQ 10300 – 1460 bytes of data



SEQ 2000 – 0 bytes of data; ACK 8840, Win=1460



SEQ 2000 – 0 bytes of data; ACK 8840, Win=0



# Lab: Congestion

**File:** download-bad.cap

Select View > Time Display Format > Seconds Since Previous Displayed Packet

Sort the packets on the Time column

What is the cause of the highest delays in this trace file?

Re-sort by the number column

- Which packet filled the receive buffer?
- What was the total delay time caused by receiver congestion?
- Was packet loss and high latency the biggest problem in this trace file?

# What's Next?

Laura's Lab Kit v9

In show bags as well as...

ISO image: [www.novell.com/connectionmagazine/laurachappell.html](http://www.novell.com/connectionmagazine/laurachappell.html)

Wireshark University: [www.wiresharkU.com](http://www.wiresharkU.com)

Laura's Blog: [laurachappell.blogspot.com/](http://laurachappell.blogspot.com/)

