



10 Cool Things You Should Know How to Do with Wireshark

June 16, 2010


Laura Chappell

Founder | Chappell University/Wireshark University

SHARKFEST '10

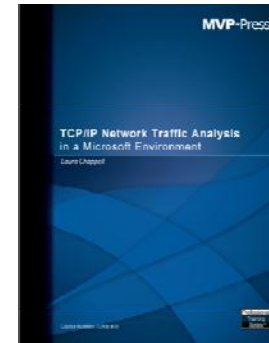
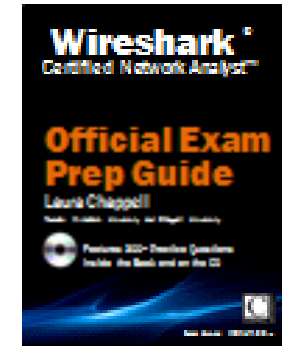
Stanford University

June 14-17, 2010



What's Up These Days?

- **Translations of Wireshark Network Analysis**
- **Wireshark Certified Network Analyst Exam Release**
- **Wireshark Certification Official Exam Prep Guide**
- **Wireshark Certification Bootcamps**
- Oh yeah... and this little **“Microsoft project”**



Skills to Master

1. Perform Local/Remote Capture Like a Pro

Locate most active interface

Test your interfaces (see video at wiresharkbook.com)

Use `rpcapd.exe` for remote capture

2. WLAN Graphing (Get a Wi-Spy Adapter now... Just do it!)

Graphing 802.11 retries (`wlan.fc.retry == 1`)

3. VoIP Playback

Look for jitter, packet loss and errors

Skills to Master

4. Create Sexy Hot Profiles

Free profiles online at wiresharkbook.com

Video on copying in profile info at wiresharkbook.com

5. Recognize Malicious Traffic Patterns

Have a baseline ready

Know scanning/discovery signs

Colorize questionable traffic

6. Analyze an Application

What is the process?

Skills to Master

6. Command-line statistical reporting

Using Tshark effectively

7. Perform QoS Comparisons

Graphs						
Graph 1	Color	Filter:		Calc:	SUM(*)	Style: Line
Graph 2	Color	Filter:	tcp.port==21	Calc:	AVG(*)	tcp.analysis.ack_rtt
Graph 3	Color	Filter:	tcp.port==80	Calc:	AVG(*)	tcp.analysis.ack_rtt
Graph 4	Color	Filter:	tcp.port==8080	Calc:	AVG(*)	tcp.analysis.ack_rtt
Graph 5	Color	Filter:		Calc:	SUM(*)	Style: Line

8. Compare subnet performance

Same as #7, but use subnet filters such as ip.addr==10.2.0.0/16

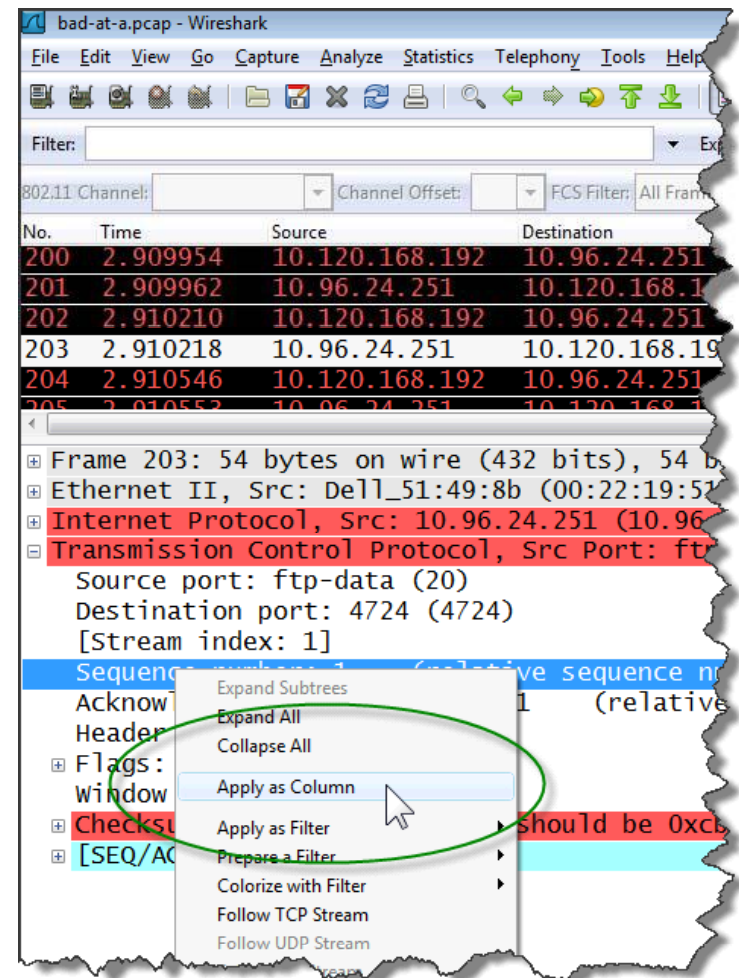
Skills to Master

9. Add Columns Fast!

Available with version. 1.4.0rc1

Right click on any field and select
Apply as Column

Right click column headings to align,
rename and more (yes – you can
left-align the No. column!)



Skills to Master

10. Build Your “Exclusion Filter of Death”

```
ip.addr==192.168.0.106 && !srvloc && !dns && !ip.addr==74.6.114.56  
&& !ip.addr==239.255.255.250 && !ip.addr==96.17.0.0/16 &&  
!ip.addr==192.168.0.102 && !smb && !nbns && !ip.addr==  
192.168.0.103 && !ip.addr==64.74.80.187 && !ip.addr==83.150.67.33  
&& !ip.addr==67.217.0.0/16 && !ip.addr==66.102.7.101 &&  
!ip.addr==216.115.0.0/16 && !ip.addr==216.219.0.0/16 &&  
!ip.addr==69.90.30.72
```

See **Analyzing TweetDeck Ttraffic** Project Report at
www.chappellseminars.com/projects.html

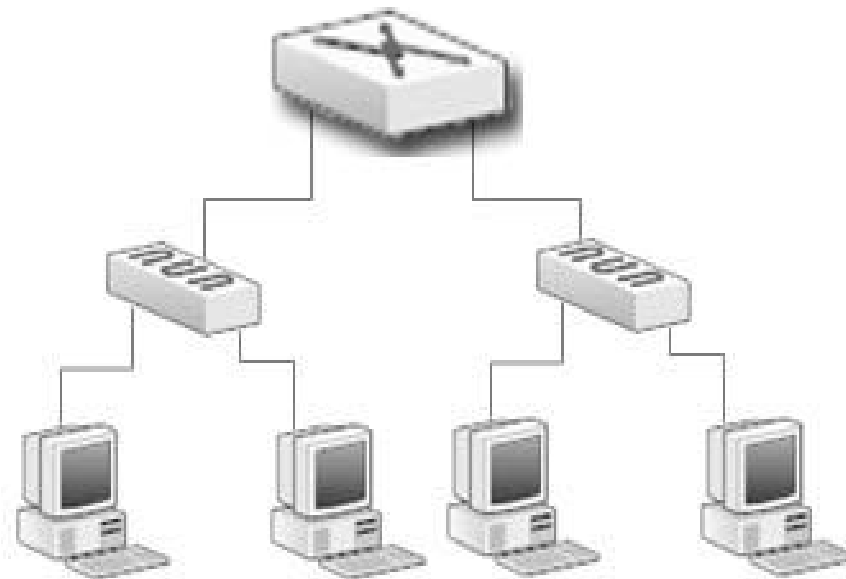
See “Google over SSL” Analysis at
www.wiresharkbook.com/coffee



Let's Go Play with Wireshark

- Profile Stuff
- Application Analysis Stuff
- Advanced IO Graphing Stuff
- Whatever else comes to mind...

Remote Capture with Rpcapd.exe

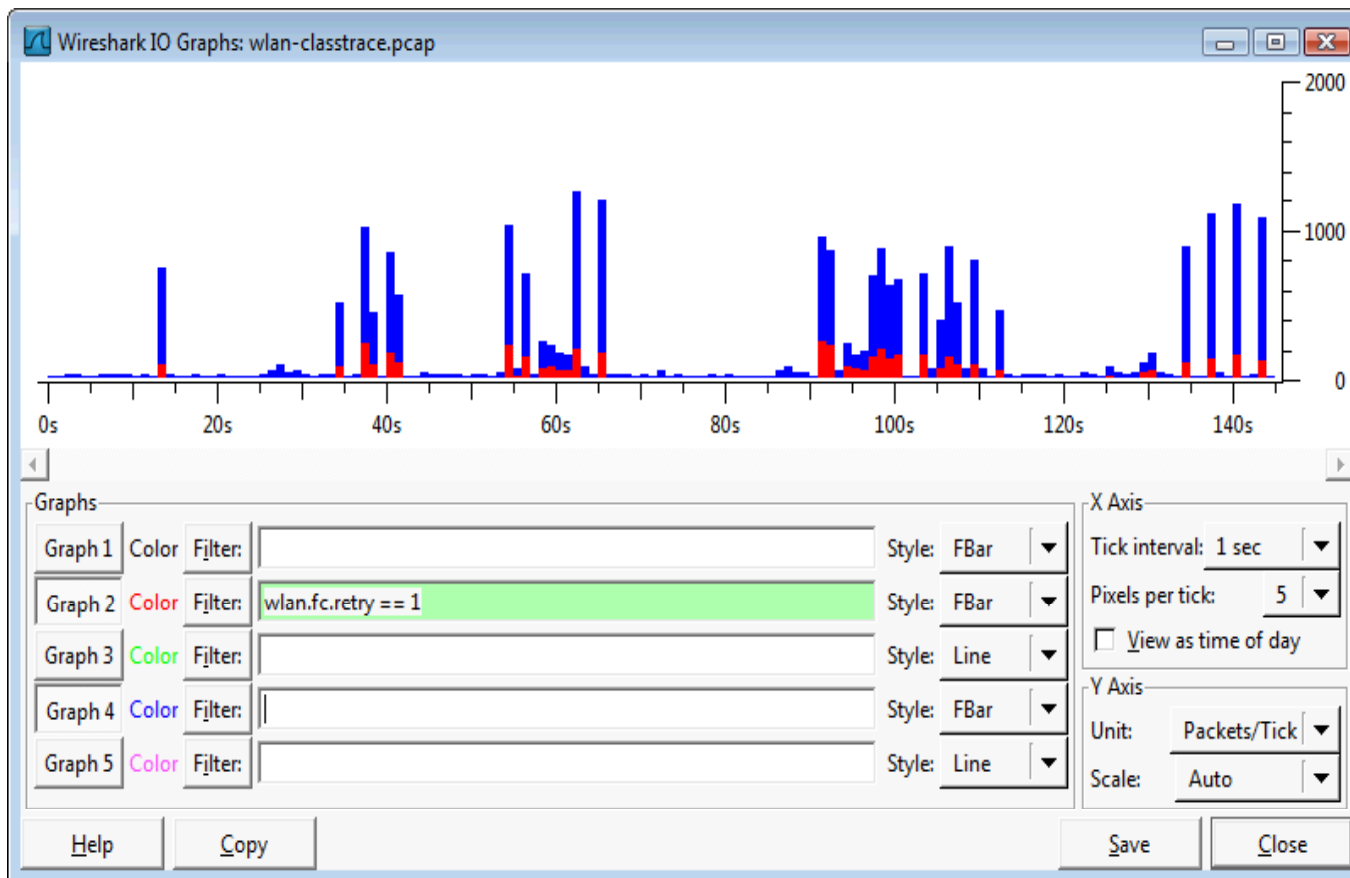


Wireshark
Capture Options
Interface: Remote
Host: 192.168.0.102
Port: 2002





```
rpcapd -n  
(default port: 2002)
```

Graphing WLAN Retries

`(wlan.fc.retry==1) && (wlan.sa==00:24:b2:1f:27:f9)`



Try Application Analysis Yourself!

- Launch First Instance of Wireshark
- Clear DNS and browsing cache (`ipconfig /flushdns`)
 - Start capture 
 - <http://sharepoint.microsoft.com/?wax=off>
 - Stop capture 
- Launch Second Instance of Wireshark
- Clear DNS and browsing cache (`ipconfig /flushdns`)
 - Start capture 
 - <http://sharepoint.microsoft.com/?wax=on>
 - Stop capture 

Capture on your local host while running Wireshark and connecting to the site.

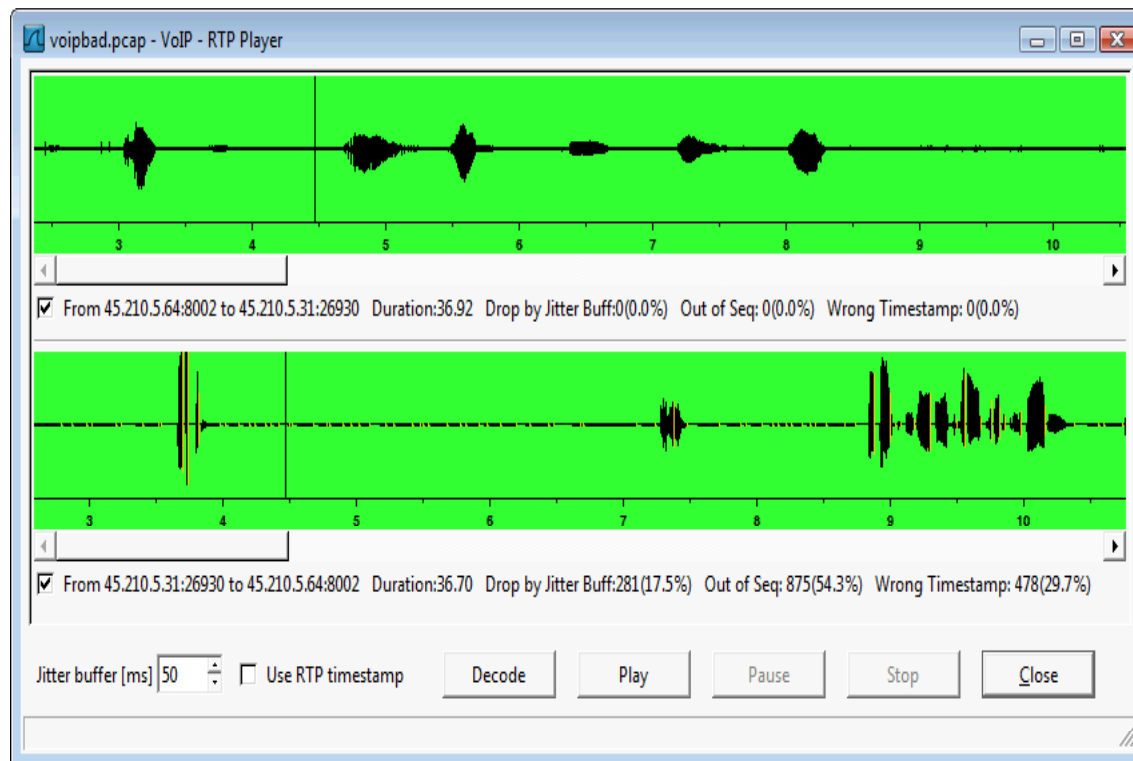


Compare Conversations (Time Values)

Statistic	Aptimize		Difference
	Off	On	
Time to Load Page Plus Links (secs)	6.91	5.33	24.30% faster launch
Packets to Load Page Plus Links	2,180	1,651	22.90% fewer packets
Bytes to Load Page Plus Links	1,779,036	1,468,861	17.44% fewer bytes
HTTP GET Requests	90	34	62.22% fewer GETs

VoIP Analysis and Playback

- Telephony | VoIP Calls | [select call] | Player | Decode [Check conversation(s)] | Play



Malicious Traffic Detection

- Baseline, baseline, baseline

The screenshot displays the Wireshark interface with a packet capture window and a coloring rules dialog box. The packet capture window shows a list of packets with columns for No., Time, Delta, and Source. Packet 3939 is highlighted in blue, indicating it matches a coloring rule. The coloring rules dialog box is open, showing a list of rules with their names and filter strings. The rule 'ICMP Response to TCP Packet (Sender Firewall?)' is highlighted in green, and its filter string is 'icmp && (tcp)'. The dialog box also includes buttons for 'New', 'Edit...', 'Enable', 'Disable', 'Delete', 'Import...', 'Export...', and 'Clear', along with 'OK', 'Apply', and 'Cancel' buttons.

No. .	Time	Delta	Source
3933	33.135265	0.000034	128.241.19
3934	33.135303	0.000038	128.241.19
3935	33.135339	0.000036	128.241.19
3936	33.135368	0.000029	128.241.19
3937	33.135403	0.000035	128.241.19
3938	33.138045	0.002642	192.168.0.
3939	33.138424	0.000379	192.168.0.
3940	33.142420	0.003996	128.241.19

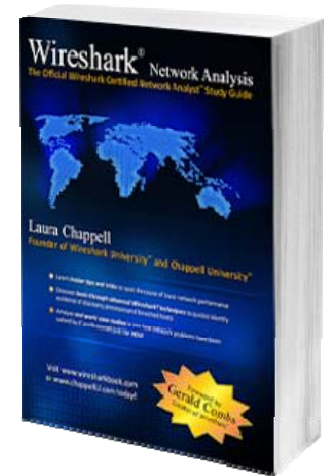
Name	String
Default IRC TCP Ports 6666-6669 (IRC Traffic - Bot Issue?)	tcp.port == 6666 tcp.port == 666
DHCP NACK (DHCP Server Does Not Like Target)	(bootp.option.type == 53) && (bo
DNS Answers > 5 (IRC Server List in this Packet?)	dns.count.answers > 5
ICMP Destination Unreachables (TCP Firewall Host?)	tcp && icmp.type==3 && (icmp.c
ICMP Protocol Unreachable (IP Scan Underway?)	icmp.type==3 && icmp.code==2
ICMP Response to TCP Packet (Sender Firewall?)	icmp && (tcp)
ICMP TTL Exceeded (Traceroute Underway?)	icmp.type==11
ICMP Type 3/Code 4 (Black Hole Detection?)	icmp.type == 3 and icmp.code ==
ICMP Types 13, 15 or 17 (OS Fingerprinting?)	icmp.type == 13 icmp.type == 15
Non-Standard ICMP Echo Request (Can You Detect the App?)	icmp.type == 8 && icmp.code==
PPI Signal < -80 (Weak Signal Strength at Antenna Location)	ppi.80211-common.dbm.antsignal
RadioTap Signal < -80 (Weak Signal Strength at Antenna Location)	radiotap.dbm.antsignal < -80

Tshark Command-Line Statistics

- From **Wireshark Network Analysis**

`-z <statistics>` Examples

<code>tshark -qz io,phs</code>	Display protocol hierarchy statistics as seen in Figure 367
<code>tshark -qz conv,eth -z conv,ip -z conv,tcp</code>	Display Ethernet, IP and TCP conversation statistics
<code>tshark -qz conv,eth -z conv,ip -z conv,tcp</code>	Display Ethernet, IP and TCP conversation statistics
<code>tshark -qz io,stat,10,ip,udp,tcp</code>	Display IO statistics for IP, UDP and TCP traffic at 10 second intervals
<code>tshark -z io,stat,5,icmp -w allpkts.pcap</code>	Displays IO statistics for ICMP traffic at 5 second intervals—all traffic is saved to a trace file called <i>allpkts.pcap</i> (Note the filter used for ICMP is not applied to the traffic captured—to apply this filter to the traffic captured, use the <code>-f</code> parameter)



Tshark Command-Line

- **tshark -i 3 -qz conv,eth -z conv,ip -z conv,tcp**

`-i 3`

Capture on the
3rd interface
listed by
tshark -D

`-qz conv,eth`

Don't show
packets (`-q`), but
capture Ethernet
conversation
statistics

`-z conv,ip`

Only use `-q`
once. Capture IP
conversation
statistics

`-z conv,tcp`

Only use `-q`
once. Capture
TCP
conversation
statistics

Keep Up with Me

- **Twitter** - www.twitter.com/laurachappell
- **Newsletter** (chappellU.com)
- **Wireshark Weekly Tips** (wiresharktraining.com)
- **Free Wireshark Webinars** (chappellU.com)
- **Microsoft Project** -
<http://facebook.com/MVPpress> - Search for post “Laura Needs Your Help” and reply with your ideas and suggestions