T2-1
Analyzer Placement and Baseline Techniques
March 31, 2008

Tony Fortunato
Sr Network Specialist | The Technology Firm

SHARKFEST '08
Foothill College
March 31 - April 2, 2008
About your Presenter

Tony Fortunato, Sr Network Specialist, The Technology Firm

Website: www.thetechfirm.com

A Senior Network Specialist with experience in performance testing, network design, implementation, and troubleshooting LAN/WAN/Wireless networks, desktops and servers since 1989.

Tony has taught at Colleges/Universities, Networld/Interop and many onsite corporate settings to thousands of analysts.

Tony is an authorized and certified Fluke Networks and Wireshark Instructor, but trains and uses Sniffer, MRTG and many other products. Tony always demonstrates his vendor neutral approach to network design, support and implementations.

Tony has architected, installed and supported various types of Residential Wireless High Speed as well as hundreds of WIFI hotspots. Tony combines custom programs, open source and commercial software to ensure a simple support infrastructure.

Tony works on networks from 2 to 120,000 nodes and specializes in post installation performance/design review. This process involves using various tools (Protocol analyzers, traffic generators and network management) and working on multi-vendor equipment (switches, routers, servers, etc).

Tony works at customer sites within a range of capacities from project management, network design, consulting, troubleshooting, designing customized courses and assisting with installing physical equipment.
Baselining vs Troubleshooting

I personally believe that it doesn’t matter if you are troubleshooting or application baselining, the techniques and issues described in the next few slides apply to both.

I always say that, “if you do a baseline correctly, you will find something to investigate or tweak. In some cases, you may discover that the application has always had issues, but no one has ever looked into it.”
The overall methodology required to properly analyze a problem or document an application behavior falls into the following categories:

- Concisely identify the issue, purpose or goal of the exercise
- Position yourself to properly capture the data
- Configure the tool to capture efficiently
- Use various reports or techniques to investigate any issues
- Document any anomalies and recommended changes
- Test..... And Document!!!!!!!!!!!!!!
Many Windows NDIS drivers or mirror ports do **NOT** forward physical layer errors to your analyzer software.

Make sure that your analyzer and card can decode trunk specific protocols and packets since they may exceed Ethernet’s maximum packet size.
Hubbing-out refers to using a hub to see traffic from a host. The hub in this case is acting as a half-duplex tap.

Since hubs are **Half Duplex**, double check switch ports and workstation settings are set to half duplex to avoid duplex mismatch issues.
I wanted to see the effects of half/full duplex mismatch and the resulting effects with various tools. I simply performed a file transfer between both devices and intentionally set a duplex mismatch between the laptop (full duplex) and the hub (half duplex).
I wanted to confirm via my Fluke Analyzer that I was causing physical level errors and obviously I was.
Lab Setup

Even at the packet level, the hardware based analyzer did see the errors, where Wireshark (with the Windows default NDIS driver) did not.

With Wireshark, we have to learn to spot the effects of physical errors. For example, in this case we see *Duplicate ACK’s* and *Retransmissions*. 
Many monitor ports do not forward packets with physical level errors.

Try to avoid mirroring or spanning multiple ports or entire Vlans.

Span or mirror From port #3 to port #1

Please ensure that the monitored ports are the same speed as the source ports.
Port Spanning or Mirroring

Many monitor ports do not forward packets with physical level errors.

Spanned all ports to port #1

Spanning more than 1 port, may cause all sorts of unpredictable results including switch failure!! If you must, proceed with extreme caution.

Network 10.1.0.0

Network 10.2.0.0
Full Duplex Links

• Aggregating taps will combine both Rx channels to one monitoring port
• Non-aggregating taps will require 2 NICS on the monitoring tool
Multiple trace files

In this example I captured a full duplex conversation using a full duplex, non-aggregating tap. The capture is a Windows XP PC and as a Dlink and Realtek NIC.
Assembling The Data

Now that we have 2 trace files, there are 3 ways to put them together:

- GUI
- Command Line
- Drag and Drop (Windows)
mergecap - Command Line (Windows)

Syntax

Usage: mergecap [options] -w <outfile>|-> <infile> ...

Example

mergecap -v -w combined.pcap realtek.pcap dlink.pcap

Where

-v is verbose mode to get some feedback while files are being processed
-w is the output file ‘combined’

Output

.....
C:\ Record: 1050
C:\ dir combined.pcap
Volume in drive C has no label.
Directory of C:\
03/30/2008  01:54 PM       676,680 combined.pcap
Open one file (Dlink.pcap)
Drag and Drop

Drag and drop both files into the Wireshark application.
Install Wireshark on the client or server experiencing the problem.

Installing any software on a production server is more difficult to accomplish, but sometimes a development server is available for testing and may exhibit the same symptoms you are trying to troubleshoot.
Make Wireshark More Convenient (windows)

Assign a Shortcut key to Wireshark (Windows) for easier access.

This tip is applicable to all tools you use frequently

- Command prompt
- Notepad
- SNMP browser
If you select your VPN driver you can capture your decrypted data to help diagnose application or connectivity issues.
Capturing From Your Wireless Card

If you select your Wireless Card, you can capture only the data packets.

If you need to capture wireless management packets such as Beacon, Acknowledgement and Probe packets you need to use a product like AirPcap.

Captured via AirPcap driver and interface
Saving To A File

Single File

File Set

Ring Buffer

or
Capture File Configurations

File Name
Multiple Files
Stop based on
  • Data captured
  • Packets captured
  • Time
Ring buffer number
File count
The following will have the last 24 hours of packets

To or from 10.44.10.1

Every minute

Ringing 5 files, showing the last 5 minutes worth of data
Capturing from the command line allows us to start captures more quickly and consistently.

The same syntax can be used in a startup, batch file or desktop shortcut, so anyone can start a capture with very little Wireshark experience.

To capture from the command line, we will use `tshark`

The only thing to determine is the interface number of the adapter you want to use. Simply go to the Wireshark program directory and type `tshark –D`. This number will be used later.

The command to capture using the same parameters as the previous slide is;

```
tshark -i number -f "host 10.0.12.15" -w hourly.pcap -b duration:3600 -b files:24
```
I always suggest that when you are baselining or troubleshooting a device or application for the first time, try to filter on the lowest address

- MAC ADDRESS; Through Layer 2 Switches
- IP ADDRESS; Through Routers
- TCP or UDP address or Data; Through Firewalls
Accelerators (Keyboard Shortcuts)

**TAB**
- Move from Packet List, Detail and Bytes

**Ctrl+Down Arrow**
- Move to next packet in (even if packet list is not in focus)
Optimizing Wireshark

Capture
- Update List of Packets in Real Time
- Capture Dialog Window
- Name Resolution
- Buffer Size (Windows)
- Protocol Tasks
- Command-Line Capture

Display
- Number of Columns
- Split the trace file

P: 342343 D: 342343 M: 0 Drops: 9348
Name Resolution Preferences

MAC name resolution

Network name resolution
- Concurrent DNS name resolution
- Maximum concurrent requests

Transport name resolution

![Name Resolution Configuration](attachment:image.png)
Find a Packet

Find packets based on

- Display filter
- Hex Value
- String
Configuring Your Time Settings

Date/Time of Day or just Time of Day
Secs. Since Beginning of Capture
Secs. Since Previous Packets
Secs. Since Epoch

Time Precision
- Automatic
- Seconds
- Deciseconds
- Centiseconds
- Milliseconds
- Microseconds
- Nanoseconds

Use the spacebar to select multiple entries
Using the Time Reference

Used to determine the time between specific packets.

Need to have time set to “Seconds Since Beginning of Capture”. If you do not, Wireshark will ask you if it’s ok for Wireshark to change it to that format.

Can create multiple reference points

The time between the Get and response is 106 ms
Taking Frames Out Of Your Trace

After ‘cleaning’ up your trace with extensive filtering you may want to save this ‘cleaner’ version

Simply ensure you have selected the ‘Displayed’ option at the bottom of the Save Dialogue box.
Difference Between Prepare and Apply

When you right click on a report or field name, you typically have the option to ‘Apply’ or ‘Prepare’ a filter.

Apply

- Takes whatever you have selected and immediately ‘applies’ it

Prepare

- Takes whatever you have selected and simply inputs it into the display filter area
- Now you can modify it prior to invoking the display criteria
IO Graphs provide a visual representation of the traffic rate.

Consider ‘color assumptions’ when assigning filters.
In the graph below I wanted to illustrate when DNS and HTTP was active.
HTTP Flow Graphing
Creating a Packet Bookmark

Sometimes when you are capturing packets, you may need to create some kind of ‘bookmark’.

To do so, simply ping something that has nothing to do with your troubleshooting.

For example ping your router with a 500 Byte payload.

The syntax for windows; **ping ipaddress –l 500**
Boot-up Baseline

This baseline observes a device’s boot-up process and provides clues as to the configuration of that device. Most common example is to baseline your new PC build.
# Boot-up Example Findings

<table>
<thead>
<tr>
<th>Servers</th>
<th>Protocol</th>
<th>Bytes/Packets</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.10.11</td>
<td>DNS</td>
<td></td>
</tr>
<tr>
<td>10.10.10.2</td>
<td>DHCP</td>
<td></td>
</tr>
<tr>
<td>10.10.10.1</td>
<td>Default Gateway</td>
<td></td>
</tr>
<tr>
<td>10.10.22.10</td>
<td>PDC</td>
<td></td>
</tr>
<tr>
<td>10.10.10.3</td>
<td>LDAP/Kerberos</td>
<td></td>
</tr>
</tbody>
</table>
## Application Baseline Example

<table>
<thead>
<tr>
<th>Task</th>
<th>Start Frame #</th>
<th>End Frame #</th>
<th>Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Launch acme Data Entry Application</td>
<td>0</td>
<td>1,000</td>
<td>100,829</td>
</tr>
<tr>
<td>Login</td>
<td>1,002</td>
<td>3,121</td>
<td>6,232,232</td>
</tr>
<tr>
<td>Query for account 123</td>
<td>3,231</td>
<td>5,764</td>
<td>13,123,385</td>
</tr>
<tr>
<td>Change name and submit</td>
<td>6,000</td>
<td>6456</td>
<td>213,489</td>
</tr>
</tbody>
</table>
Roll your Own

With a PC and 2 network interface cards, you can easily design a remote capture tool.

Using Remote Desktop or VNC, you can connect from your PC to your Wireshark and capture from the other interface.
Pathping (windows)

Use Microsoft’s pathping command to document packet loss and response time
Netstat results (windows)

Use Netstat to see what TCP/UDP ports and IP addresses your application is using

C:\>netstat  -n -b

Active Connections

<table>
<thead>
<tr>
<th>Proto</th>
<th>Local Address</th>
<th>Foreign Address</th>
<th>State</th>
<th>PID</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP</td>
<td>10.10.10.10:2716</td>
<td>142.161.235.2:17879</td>
<td>ESTABLISHED</td>
<td>2724</td>
</tr>
</tbody>
</table>

[Skype.exe]
Specific Display Filters

Once you have captured your packets, use Statistics->Conversation->Display Filters to understand your application behavior.