AU-8 Finding the Latency

17 June 2009

Ray Tompkins

Founder & CEO |

SHARKFEST **'09**

Stanford University

June 15-18, 2009

Finding Latency

- The total transaction time is the sum of all of the delta times between each of the packets
- If the overall transaction time is long, one or both of these is present
  - A few packets with long delays between them
  - Many packets with small delays between them

Myth

Upgrading the bandwidth will improve response time.

The Truth

- In most cases upgrading the bandwidth will not significantly improve application response time.
- You will pay more every month for the additional bandwidth you are not using.

What kind of application do we want?

- We want a bandwidth dependent application
  - We can always buy more bandwidth.
- We don't want a latency dependent application
  - The only way to reduce latency is to put the client closer to the server.

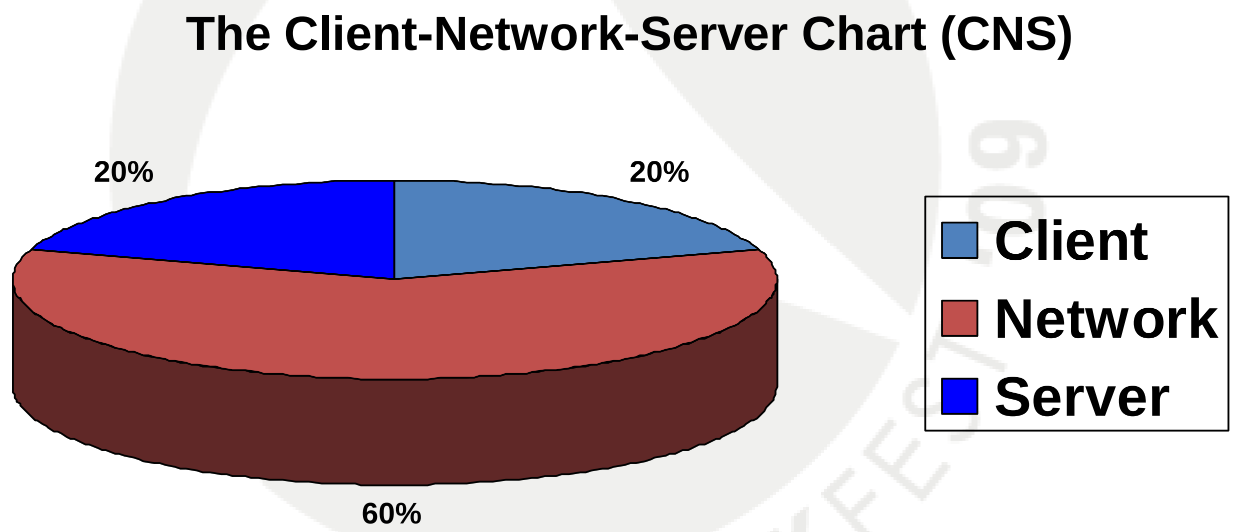
The components of latency

- Insertion Latency
  - This is the amount of time it takes to insert a frame onto a particular media type.
- Distance Latency
  - This latency is a factor of the distance that the frame must travel.
- Queue Latency
  - This latency is the result of frames sitting in router and switch queues.

The components of latency

- Request Latency
  - This is the time between requests sent by the client device. This is typically a factor of the processing power of the device.
- Response Latency
  - The amount of time it takes the receiving device to respond to the request. As with the Request Latency, this is a factor of the processing power.

Putting them all together



Insertion Latency

- 100% based on the speed of the link.
- Can be calculated by dividing the frame size by the speed of the link.
- Example:
  - 1514 Byte frame transmitted on a T-1 circuit
  - 1514 Bytes / 192,000 Bytes / Second
  - **Insertion time = 7.885 milliseconds**

Insertion Delay Example

File Edit Capture Display Tools

No.	Time	Size	Source	Destinat
1	0.000000	60	207.141.76.86	207.15
2	0.000083	60	207.159.146.32	207.14
3	0.213099	60	207.141.76.86	207.15
4	0.052647	1514	207.141.76.86	207.15
5	0.048022	1514	207.141.76.86	207.15
6	0.000119	60	207.159.146.32	207.14
7	0.387642	1514	207.141.76.86	207.15
8	0.046880	1514	207.141.76.86	207.15
9	0.000124	60	207.159.146.32	207.14

Looking at the Interframe gap between these two frames can help us determine the bandwidth between the client and server

Some sample insertion delays

		100	512	1024	1514
Link Speed	64,000	0.012500	0.064000	0.128000	0.189250
	128,000	0.006250	0.032000	0.064000	0.094625
	256,000	0.003125	0.016000	0.032000	0.047313
	512,000	0.001563	0.008000	0.016000	0.023656
	1,536,000	0.000521	0.002667	0.005333	0.007885
	10,000,000	0.000080	0.000410	0.000819	0.001211
	45,000,000	0.000018	0.000091	0.000182	0.000269
	100,000,000	0.000008	0.000041	0.000082	0.000121

The delay on the previous slide shows 48 milliseconds between two 1514 byte frames. Using this chart, we can see that the slowest link between the client and server is 256kbps

Distance Latency

- Distance latency is based on the speed at which a signal can travel through the transport media.
- Until the change the laws of physics, we are stuck with this one.
- A good estimate of this value is 1 millisecond per 100 miles traversed by the frame.

Queue Latency

- This value depends on the ability for routers and switches to forward frames as the are received.
- Congested WAN links and slow processors on these devices can increase the queue latency.
- Variations in queue latency result in Jitter.
  - Jitter has little effect on data transfer applications that utilize protocols such as TCP.
  - Jitter will adversely impact time dependent applications such as Voice over IP.

Request Latency

- This time is measured from the time that the requested receives the last byte of the previous request, to when it sends the first byte of the next request.
- If the requesting device has a hard time processing the data that it has received, this value will be large.

Request Latency Example

- In the example below, the server is responding quickly to the requests sent by the client
- The client however is taking a long time after receiving the response to send the next request

```
5122 0.197377  SMB    NT Create AndX Request, Path: \PNTTEMPL\LOANPROG\C-A51.lpr
5123 0.000535  SMB    NT Create AndX Response, FID: 0x0015
5134 0.153126  SMB    NT Create AndX Request, Path: \PNTTEMPL\LOANPROG\C-A51.lpr
5135 0.000319  SMB    NT Create AndX Response, FID: 0x4014
5140 0.064298  SMB    NT Create AndX Request, Path: \PNTTEMPL\LOANPROG\C-A51.lpr
5141 0.000475  SMB    NT Create AndX Response, FID: 0x001e
```

Response Latency

- This is the time between when a server receives a request and when it responses to the request.
- Processor, memory, system architecture, and operating systems all play a part in how quickly the server can respond to the request.

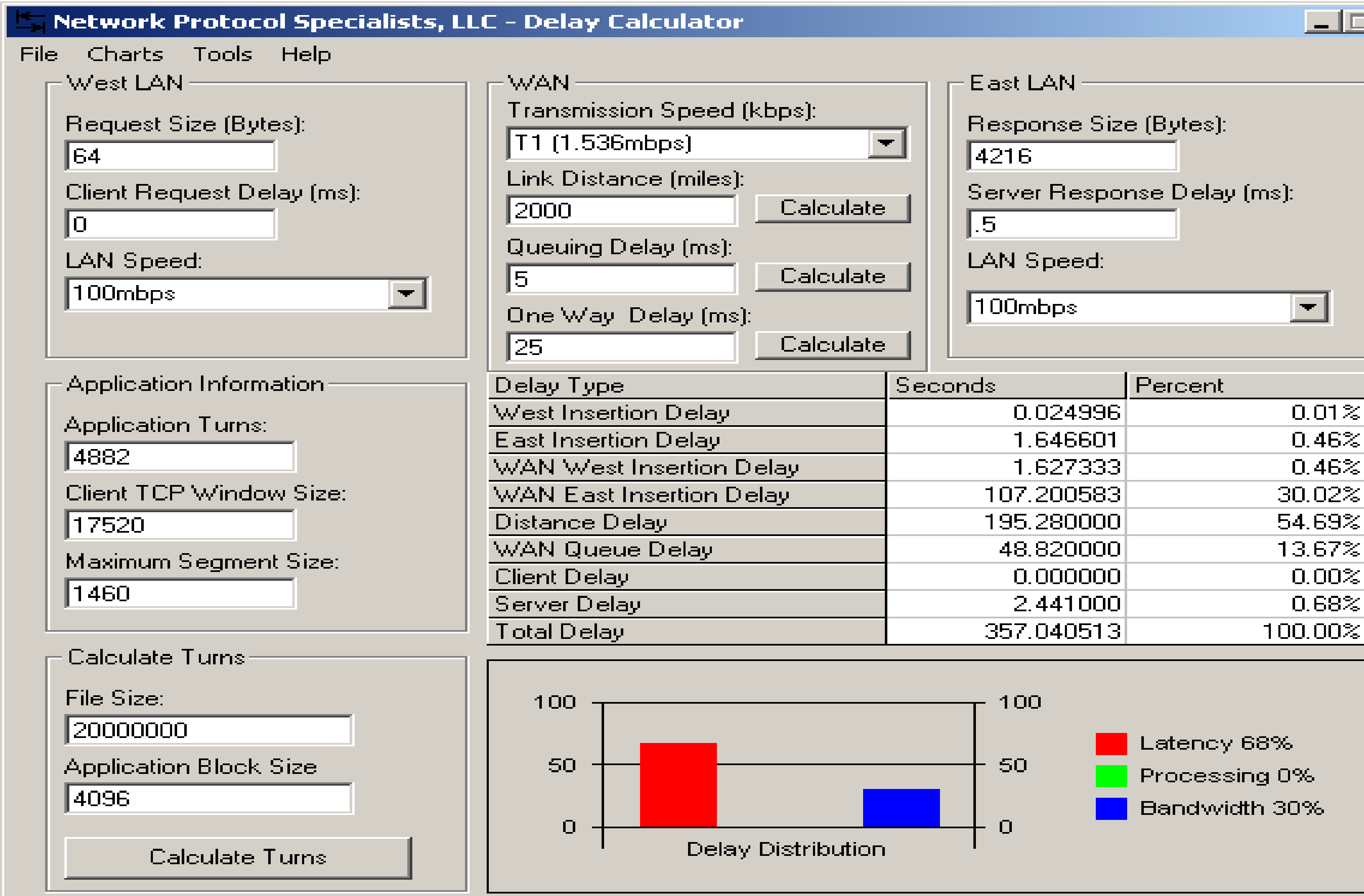
Response Latency Example

Frame 9 – We send a HTTP Get request to the Web server
Frame 10 – We get a TCP Ack back after 125 milliseconds
Frame 11 – After almost 5 seconds, we finally get the web page!

File Edit Capture Display Tools

No.	Time	Size	Source	Destination	Protocol	Info
9	0.000495	406	192.168.0.3	167.187.3.153	HTTP	GET / HTTP/1.1
10	0.125025	64	167.187.3.153	192.168.0.3	TCP	http > 1728 [ACK]
11	4.851946	249	167.187.3.153	192.168.0.3	HTTP	HTTP/1.1 200 OK

NPS Delay Calculator

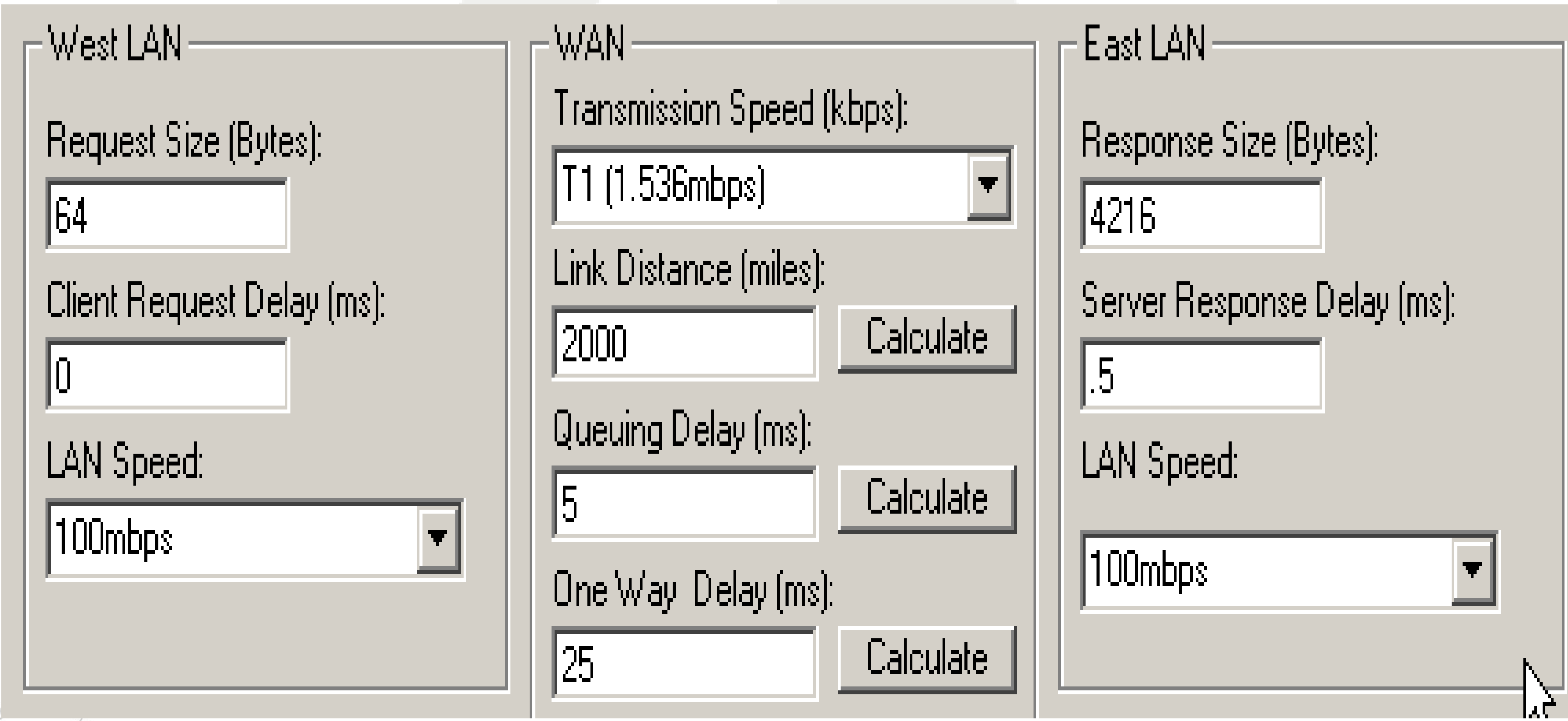


Latency Example

- In this example we
  - Transferring a 20 megabyte file from the East to the West
  - The read request size is 4096 bytes at a time
  - The roundtrip delay is 50 milliseconds
  - It takes the server .5 milliseconds to respond to the request
  - The LAN circuits on each side are 100 megabits per second

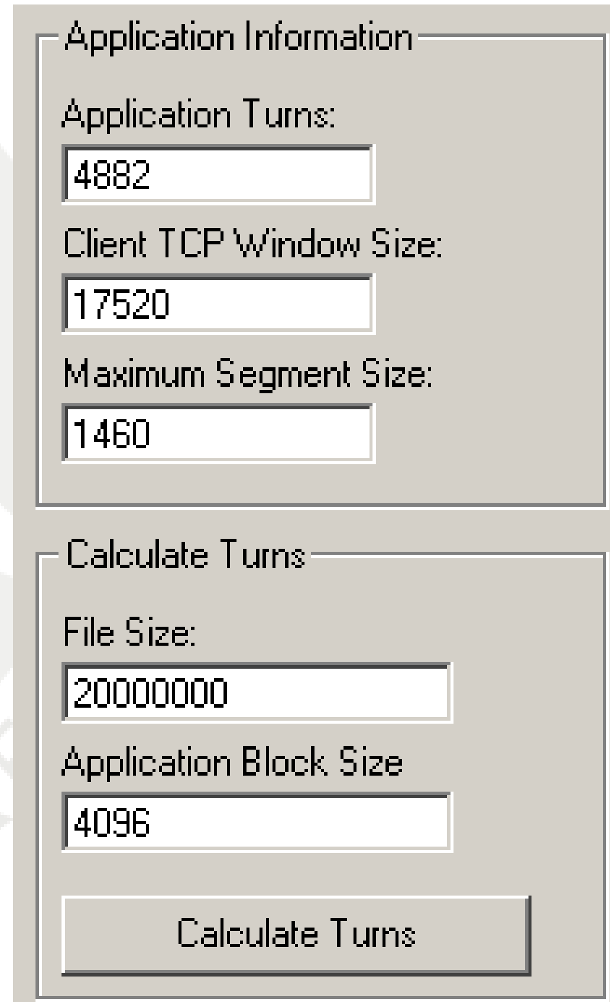
Latency Example

- Here is the network configuration



Latency Example

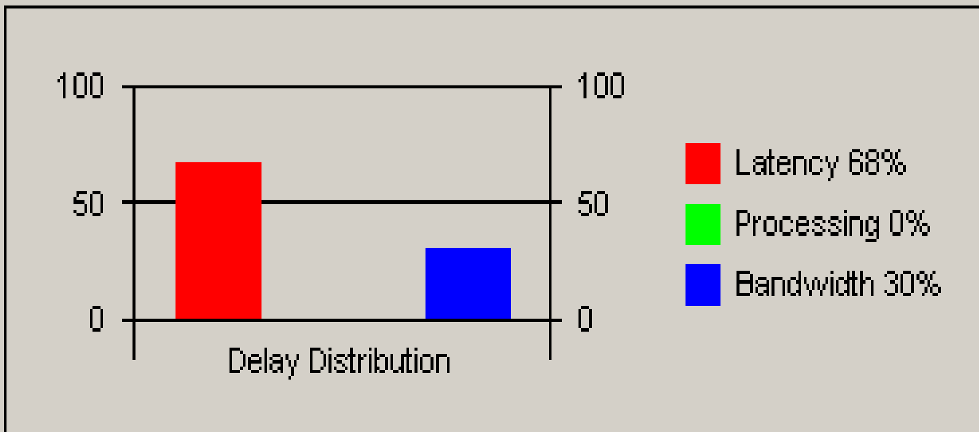
- Here is the application information
- We are sending 4882 requests and getting 4882 responses (turns)
- We are reading the 20 meg file 4096 bytes at a time



Latency Example

- Here are the results
- 68% of the time was spent waiting
- 30% is caused by bandwidth

Delay Type	Seconds	Percent
West Insertion Delay	0.024996	0.01%
East Insertion Delay	1.646601	0.46%
WAN West Insertion Delay	1.627333	0.46%
WAN East Insertion Delay	107.200583	30.02%
Distance Delay	195.280000	54.69%
WAN Queue Delay	48.820000	13.67%
Client Delay	0.000000	0.00%
Server Delay	2.441000	0.68%
Total Delay	357.040513	100.00%



Latency Example

- The insertion of packets on the LANs takes very little time
- Most of the time is spent inserting the traffic on the WAN and waiting for the roundtrip delay of the circuit

Delay Type	Seconds	Percent
West Insertion Delay	0.024996	0.01%
East Insertion Delay	1.646601	0.46%
WAN West Insertion Delay	1.627333	0.46%
WAN East Insertion Delay	107.200583	30.02%
Distance Delay	195.280000	54.69%
WAN Queue Delay	48.820000	13.67%
Client Delay	0.000000	0.00%
Server Delay	2.441000	0.68%
Total Delay	357.040513	100.00%

Wireshark Tools

- The following are features of Wireshark that can be used to determine why a transaction is taking longer than it should
  - Delta Time – The time between two frames
  - Set Time Reference – Resets the time column and shows elapsed time since the time reference
  - Follow TCP Stream – Creates a filter on IP addresses and TCP port numbers to display only the frames that are part of the TCP conversation

Delta Time

- In the example below we can see that there is a 89.377 millisecond gap between these two packets
- I prefer to set the Time column by selecting
  - View – Time Display Format – Seconds Since Previous Displayed Frame

No.	Time
35	3.551639
79	0.089377

Set Time Reference

- Here we have reset the time at frame 35
- We can see that between frame 35 and 263, 2.8091 seconds elapsed
- The time reference can be set at the beginning of a transaction and measured at the end of the transaction

No.	Time
35	*REF*
79	0.089377
80	0.089432
81	0.090187
163	0.280231
164	0.280853
166	0.288264
167	0.288320
257	2.768633
263	2.809181

Follow TCP Stream

- In the example below we selected one of the frames in the conversation that interested us
- We right clicked on any one of the frames in this conversation and selected Follow TCP Stream

	Time	Source	Destination	Protocol	Info
35	*REF*	192.168.10.20	198.238.212.10	TCP	4299 > http [SYN] Seq=0 Win=
79	0.089377	198.238.212.10	192.168.10.20	TCP	http > 4299 [SYN, ACK] Seq=(
80	0.089432	192.168.10.20	198.238.212.10	TCP	4299 > http [ACK] Seq=1 Ack=
81	0.090187	192.168.10.20	198.238.212.10	HTTP	GET /images/home/top_welcome
163	0.280231	198.238.212.10	192.168.10.20	HTTP	HTTP/1.1 304 Use local copy
164	0.280853	192.168.10.20	198.238.212.10	TCP	4299 > http [FIN, ACK] Seq=4
166	0.288264	198.238.212.10	192.168.10.20	TCP	http > 4299 [FIN, ACK] Seq=1
167	0.288320	192.168.10.20	198.238.212.10	TCP	4299 > http [ACK] Seq=404 A(
257	2.768633	192.168.10.20	198.238.212.10	TCP	4299 > http [FIN, ACK] Seq=4
263	2.809181	198.238.212.10	192.168.10.20	TCP	http > 4299 [ACK] Seq=126 A(

Putting Them Together

- We can combine these tools to zero in on a transaction that is taking a long time, measure exactly how long it is taking, and determine why it is taking so long
  - First Follow the TCP Stream
  - Set Time Reference on first frame of stream, go to bottom and see how long it took
  - If it took too long, switch back to Delta Time and look for long deltas

Documenting the Trace

- When capturing traffic that is to be analyzed later, it helps to document the trace file as you are capturing
- This will make it easier to find specific transactions later when you are looking at the trace file
- If you don't document the trace, you can spend many hours trying to find the transactions

Documenting the Trace

- Before capturing any packets, a script should be developed that will outline the transactions that will be run
- If possible, include the information that will be entered into each screen
- Create columns for the starting frame number and the ending frame number

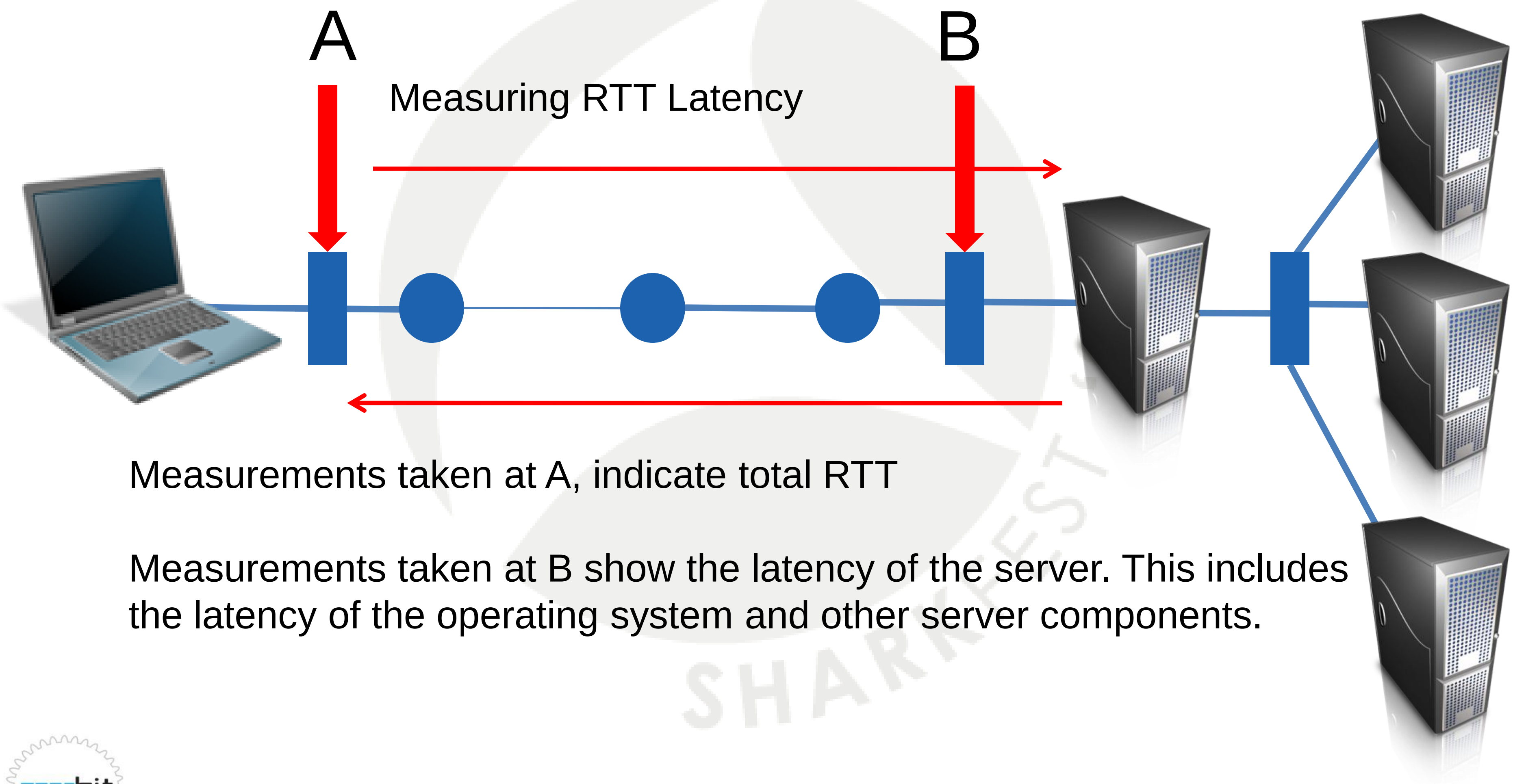
Documenting the Trace

Task	Start Frame	Stop Frame
Login into application	1	250
Select customer	251	1056
View customer detail	1057	13741
Update customer	13742	20680
Logout	20681	20732

Documenting the Trace

- After capturing and saving the trace file, you can use the spreadsheet you created to locate the specific transactions in the trace file
- DNS requests and HTTP requests can make it easy to locate the exact location of the beginning of the transaction, once you are in the right area

Where to measure latency

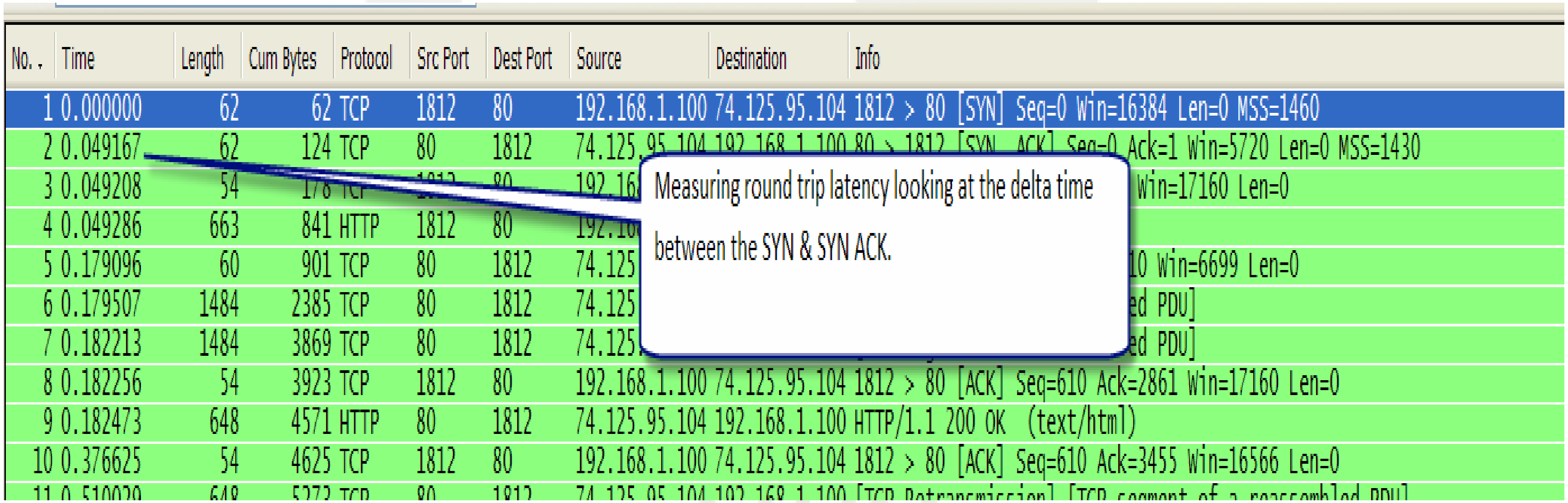


Measurements taken at A, indicate total RTT

Measurements taken at B show the latency of the server. This includes the latency of the operating system and other server components.

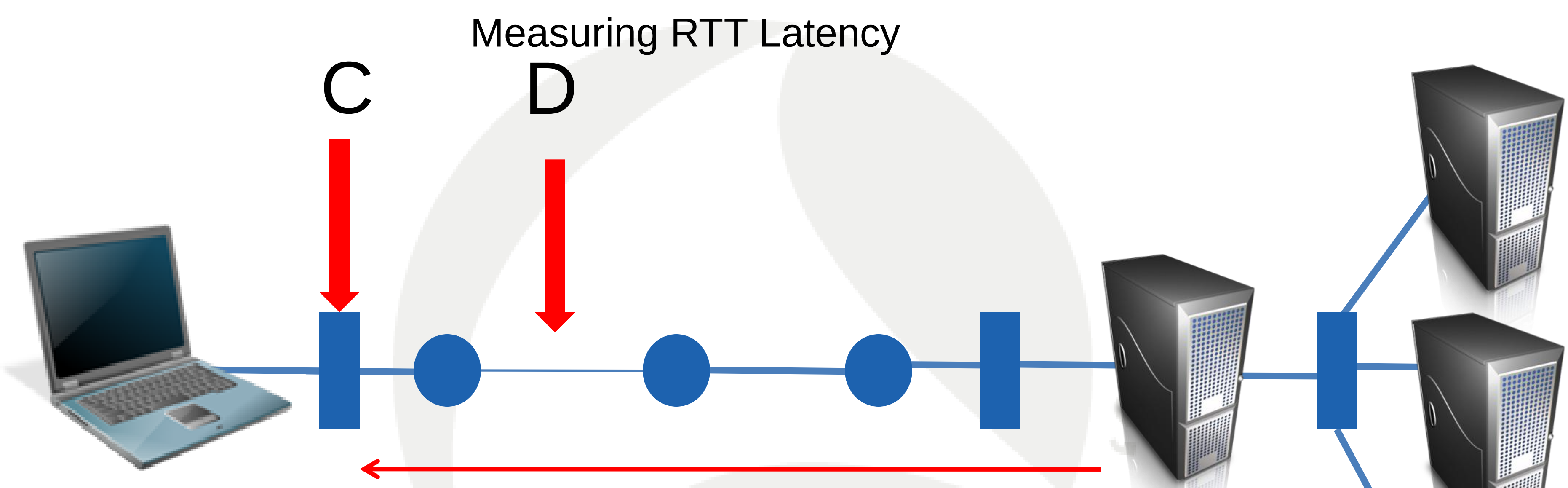
Measuring Round Trip Latency

Measuring RTT by looking at the delta time between the SYN and SYN, ACK (if capture trace is taken close to the client)
Measuring RTT can also be done by looking at the delta time between the SYN, ACK and ACK. (if capture trace is taken close to the server)



No.	Time	Length	Cum Bytes	Protocol	Src Port	Dest Port	Source	Destination	Info
1	0.000000	62	62	TCP	1812	80	192.168.1.100	74.125.95.104	1812 > 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460
2	0.049167	62	124	TCP	80	1812	74.125.95.104	192.168.1.100	80 > 1812 [SYN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430
3	0.049208	54	178	TCP	1812	80	192.16[illegible]	[illegible]	[illegible] Win=17160 Len=0
4	0.049286	663	841	HTTP	1812	80	192.16[illegible]	[illegible]	[illegible]
5	0.179096	60	901	TCP	80	1812	74.125[illegible]	[illegible]	[illegible]10 Win=6699 Len=0
6	0.179507	1484	2385	TCP	80	1812	74.125[illegible]	[illegible]	[illegible]ed PDU]
7	0.182213	1484	3869	TCP	80	1812	74.125[illegible]	[illegible]	[illegible]ed PDU]
8	0.182256	54	3923	TCP	1812	80	192.168.1.100	74.125.95.104	1812 > 80 [ACK] Seq=610 Ack=2861 Win=17160 Len=0
9	0.182473	648	4571	HTTP	80	1812	74.125.95.104	192.168.1.100	HTTP/1.1 200 OK (text/html)
10	0.376625	54	4625	TCP	1812	80	192.168.1.100	74.125.95.104	1812 > 80 [ACK] Seq=610 Ack=3455 Win=16566 Len=0
11	0.510029	648	5273	TCP	80	1812	74.125.95.104	192.168.1.100	[TCP Retransmission] [TCP segment of a reassembled PDU]

Where to measure latency

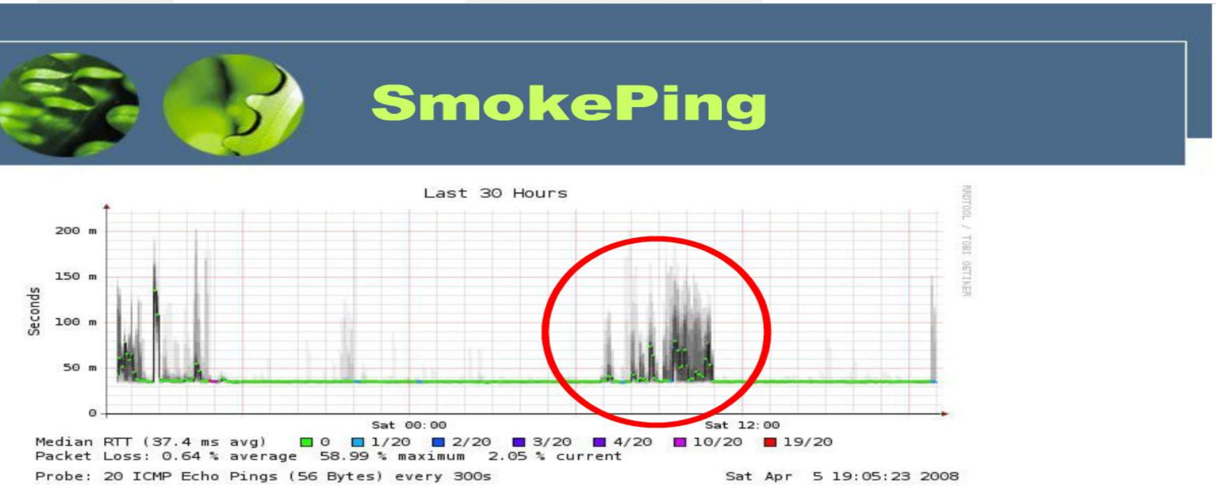


C - Taking capture trace before the router
D – At the same time capturing after the router

This allows us to review the delta time packet to packet, comparing C capture trace to D capture trace. The end result will can measure the latency caused by the router.

Other Tools Used to Measure Latency

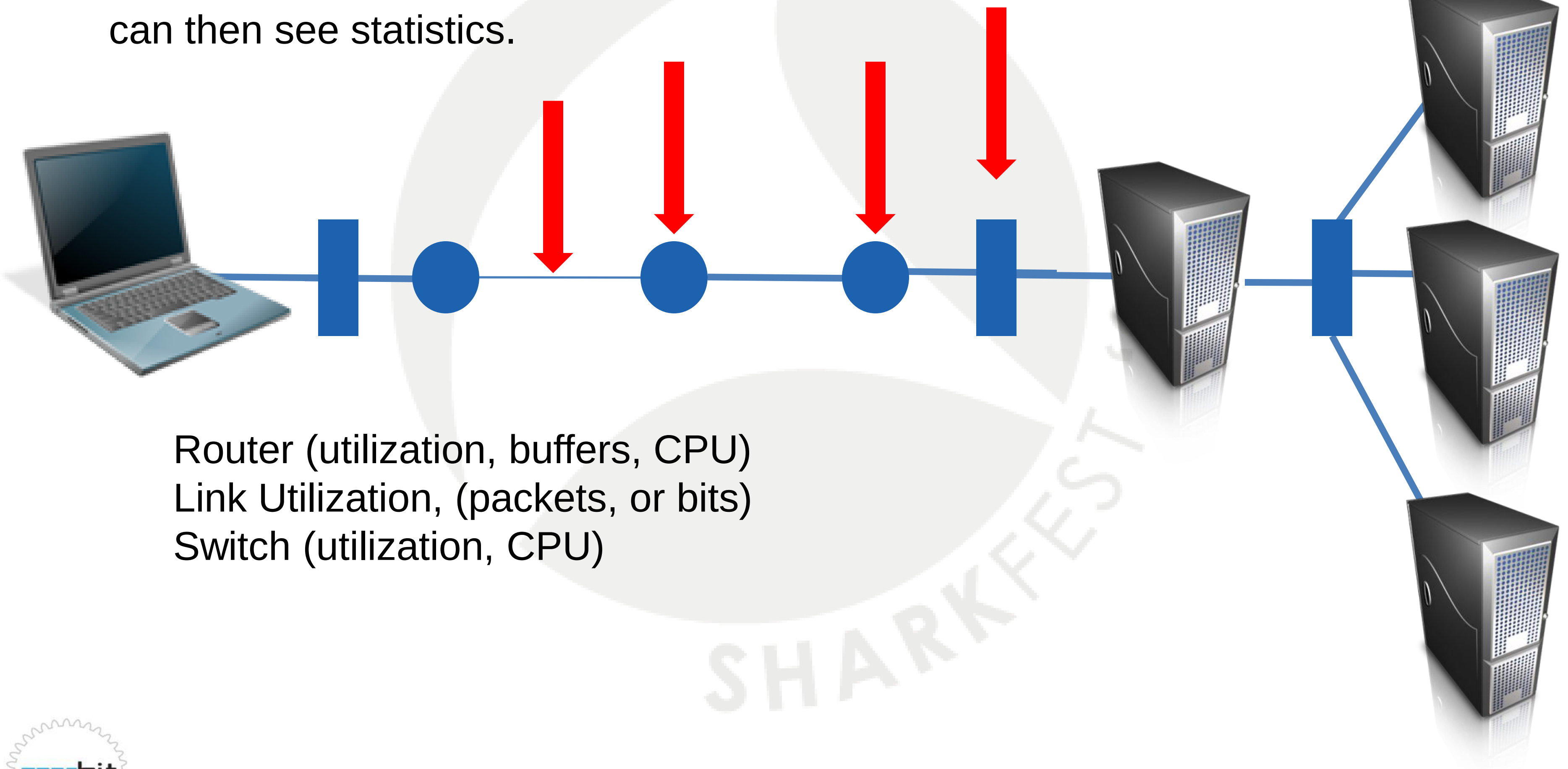
- SmokePing provides a graph of ping times, giving a historical view of latency



- Looking at the 30 hour view, we see a spike in Ping response times
- Is this an Internet problem, local problem, remote problem?

Other tools to measure latency

Other tools such as Cacti all us to look into the switch or router. We can then see statistics.



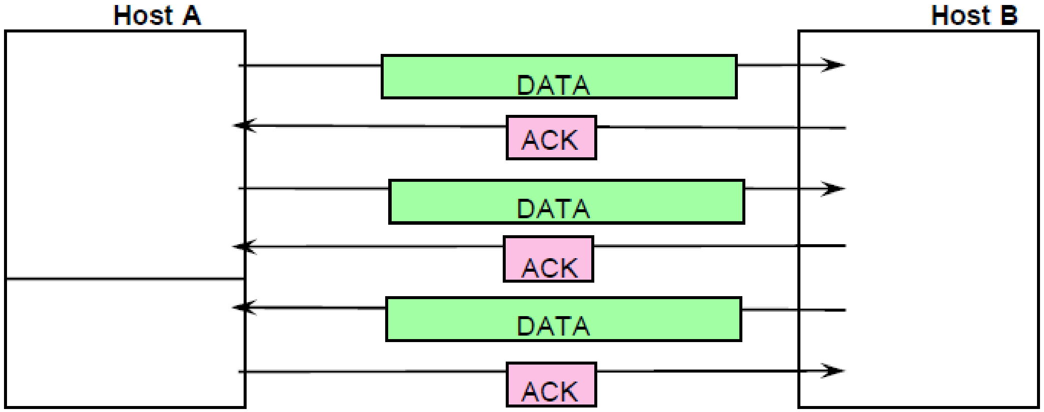
Router (utilization, buffers, CPU)
Link Utilization, (packets, or bits)
Switch (utilization, CPU)

What are the effects of Window Size

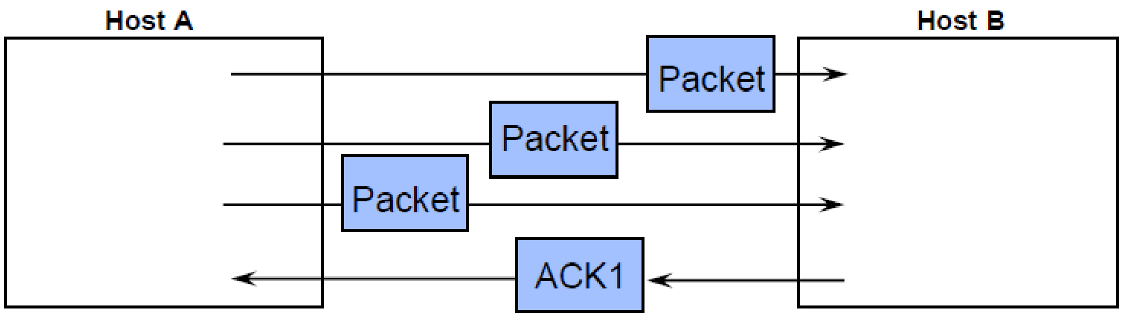
- One reason for poor performance or through put can be seen by observing the TCPWindowSize.
- When the Window size of the receiving station reaches Zero, the sending station will wait until the receiving station advertises a Window Size greater than Zero.
- Reasons for Zero Window
- LegacyApplicationsnotrecompiledfor16/32bitoperatingsystems
- Poorly designed application
- Overloaded station or Server
- To eliminate an over loaded server, try other file transfer utilities [i.e.FTP] or observe if other application ports are having Window symptoms.

TCP Transmission Types

Positive Acknowledgment with Retransmission (PAR)

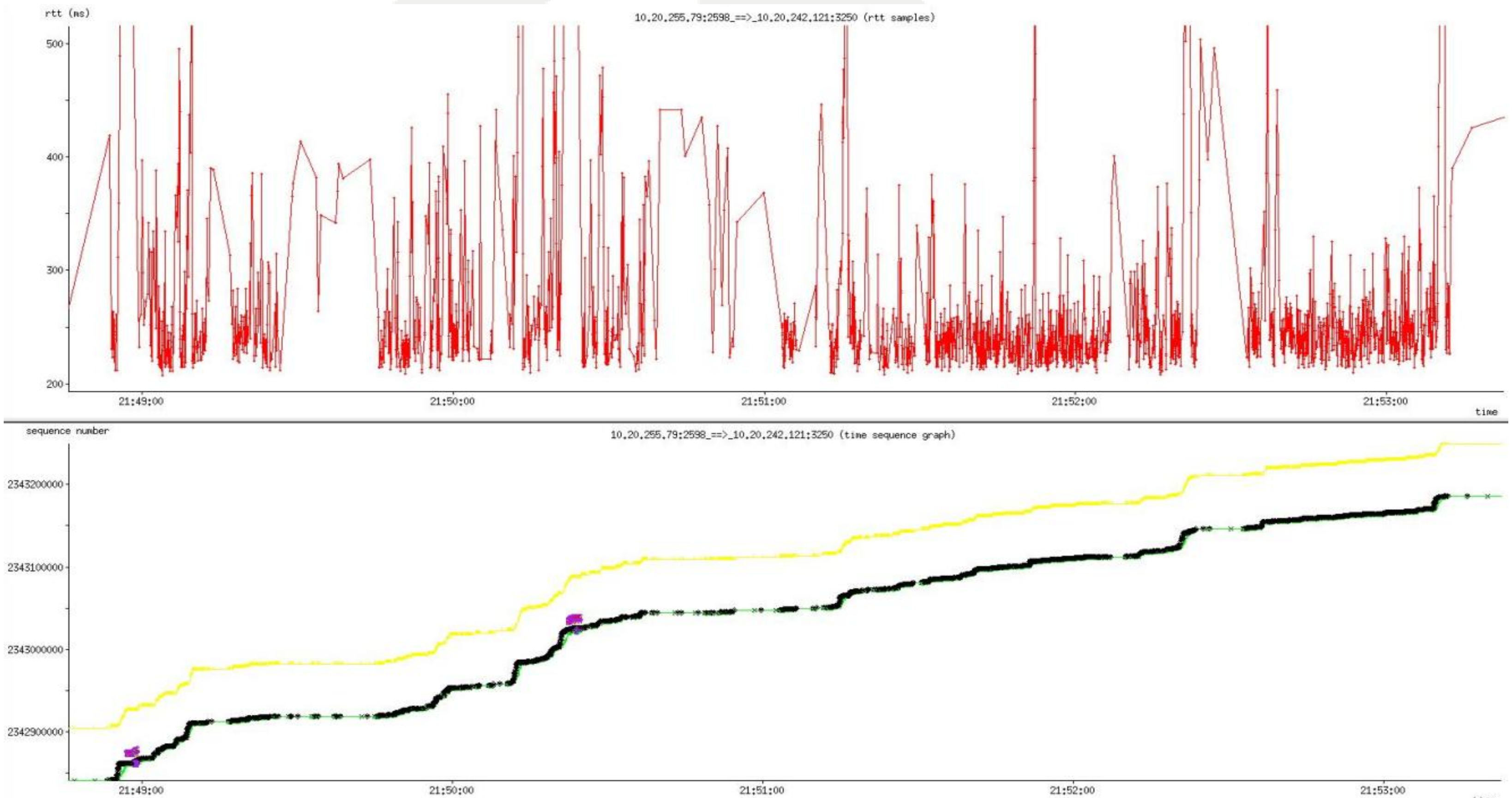


- Sliding Window



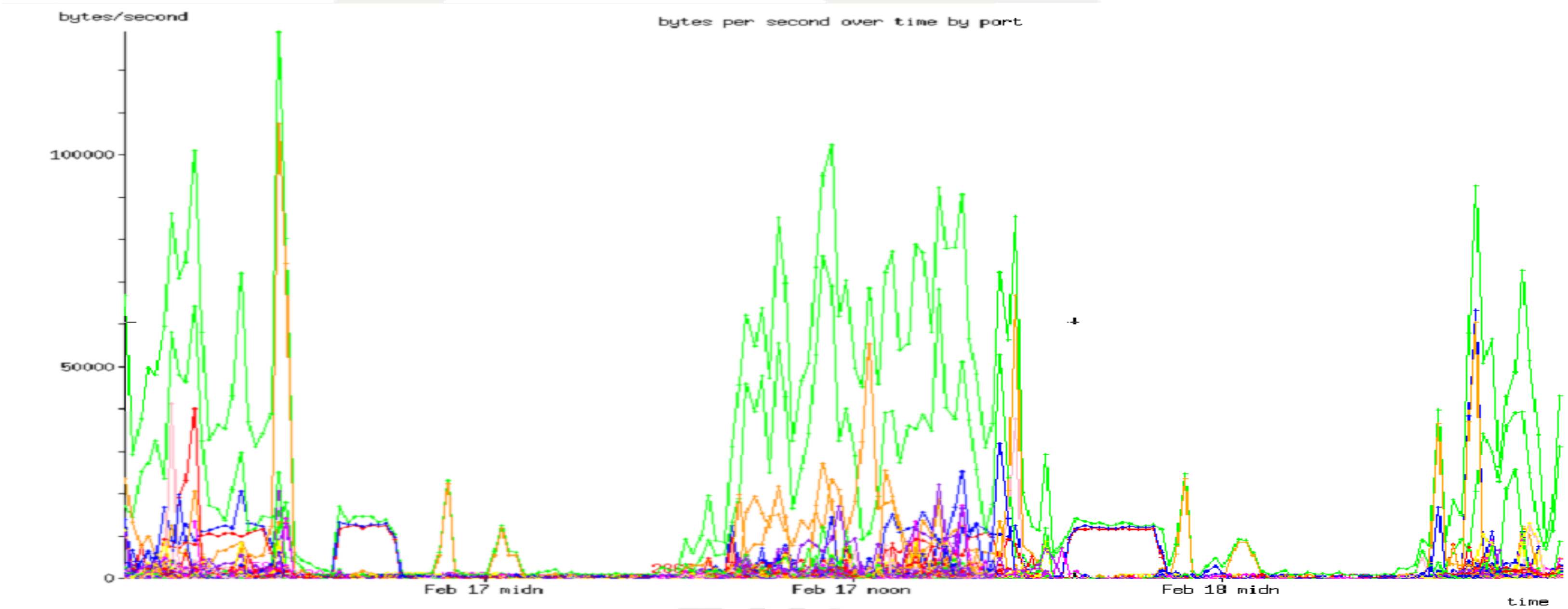
Case Study #1:Round Trip Latency

Transaction through-put is being affected by the round trip time (RTT). The delay is measured in milli-seconds and can be seen in the line graph

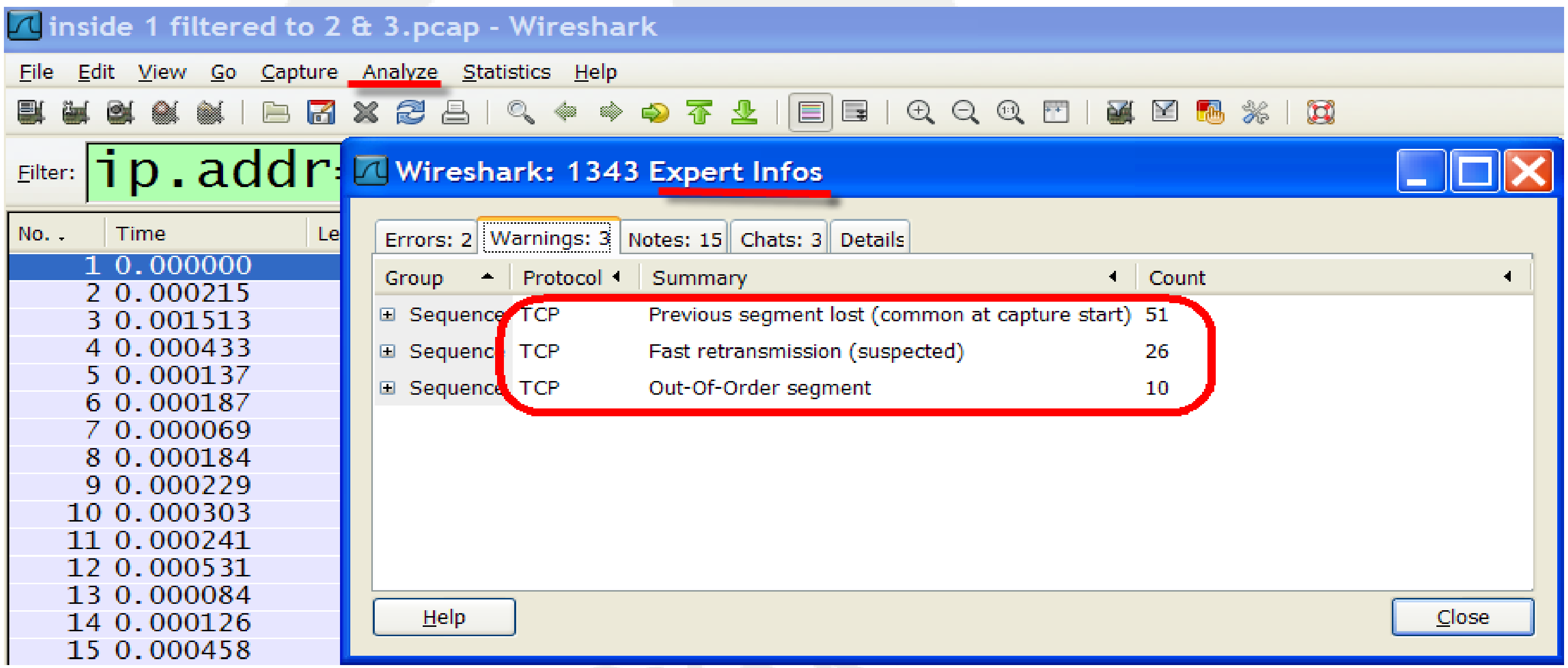


Case Study #1: What's Using the Network?

Green-	Total traffic
Light Green-	Citrix 2598
Brown-	HTTP port 80
Blue-	Microsoft SMB port 445



Case Study #2: Lost Packets Cause Latency



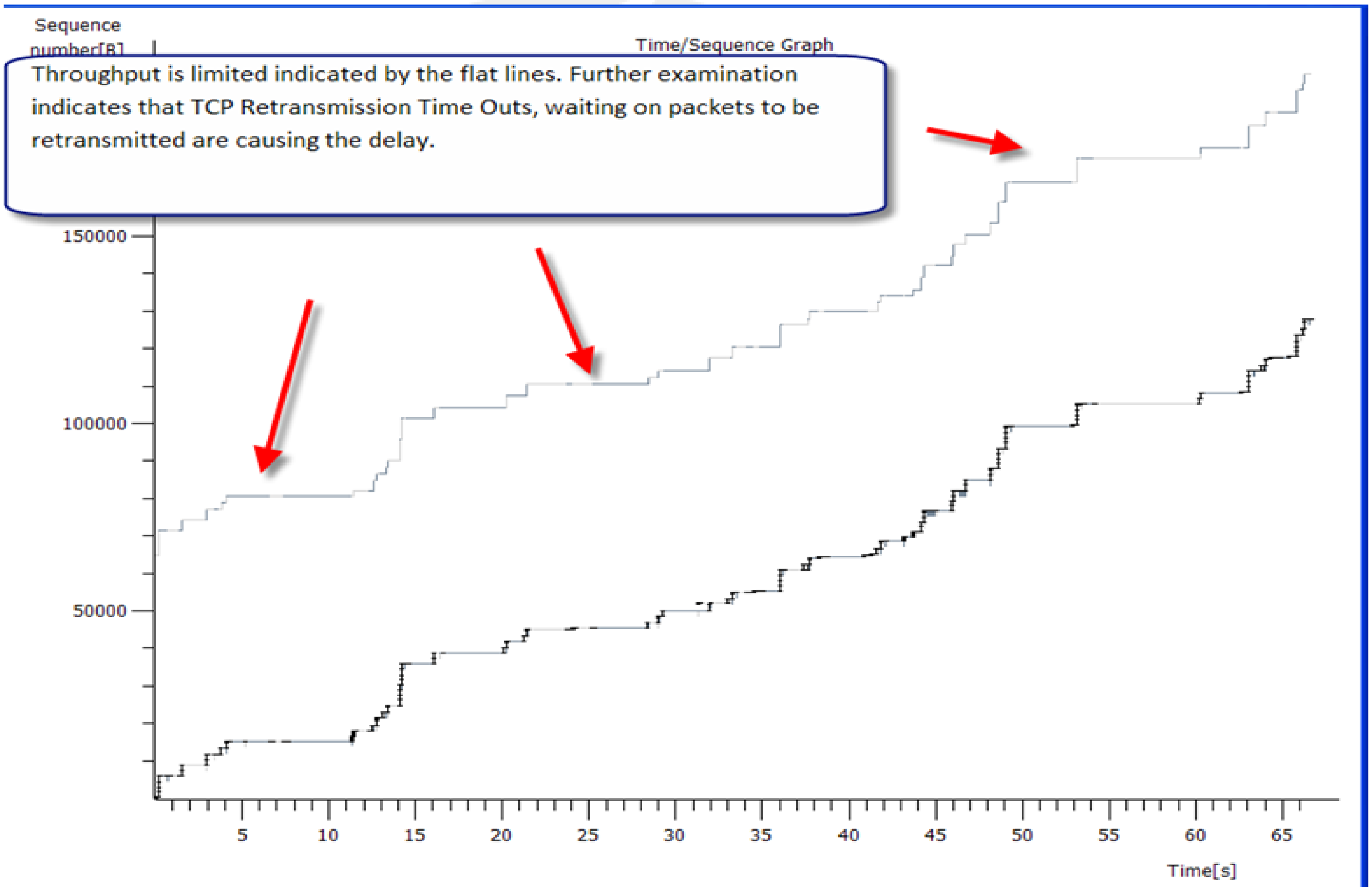
gearbit
Get it in Gear

Retransmission Cause Hugh Delays

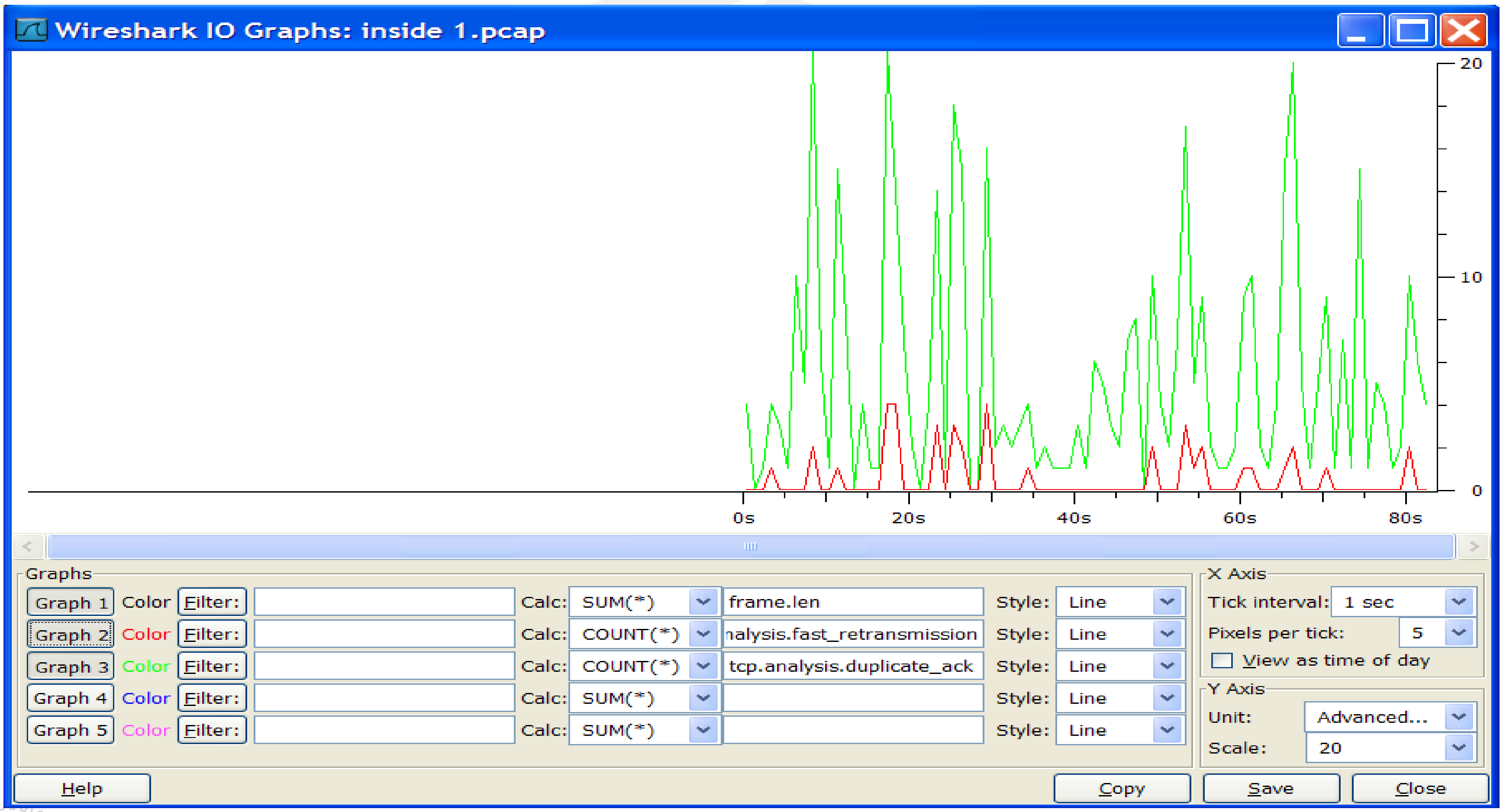
- A retransmissions cause delays:
  - delay times waiting on ACKs
  - packets need to be retransmitted

  - if packets and lost again further delays with both waiting on ACKs and retransmitted packets

Case Study #2 Lost Packets Graph

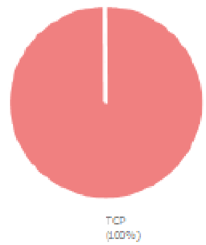
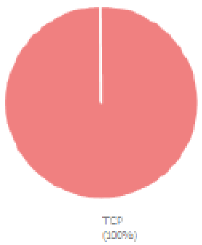
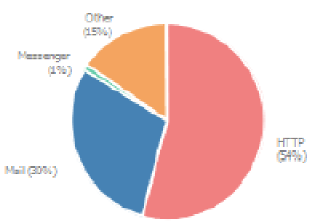
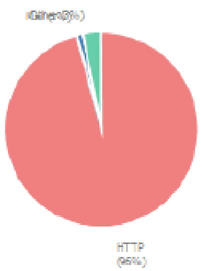


Case Study #2 Retransmissions Graph



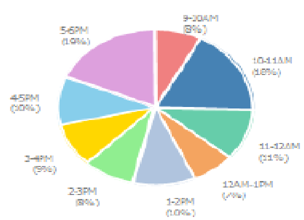
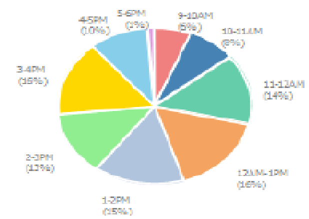
What's Using Bandwidth?

Protocol Distribution

Protocol	Data Sent		Data Rcvd	
TCP	604.5 MBytes		3.3 GBytes	
UDP	2.5 MBytes		4.2 MBytes	
ICMP	171.6 KBytes		7.5 KBytes	
(R)ARP	0.1 KBytes		0.0 KBytes	
Protocol Distribution				
IP Distribution				

What’s Using Bandwidth?

Host Traffic Stats

Time	Tot. Traffic Sent	% Traffic Sent	Tot. Traffic Rcvd	% Traffic Rcvd
5 PM	113.1 MBytes	18.6 %	35.7 MBytes	1.0 %
4 PM	61.4 MBytes	10.1 %	341.0 MBytes	9.9 %
3 PM	55.7 MBytes	9.2 %	540.0 MBytes	15.8 %
2 PM	51.5 MBytes	8.5 %	450.7 MBytes	13.1 %
1 PM	62.2 MBytes	10.2 %	515.0 MBytes	15.0 %
12 PM	43.8 MBytes	7.2 %	561.9 MBytes	16.4 %
11 AM	64.0 MBytes	10.5 %	494.2 MBytes	14.4 %
10 AM	107.4 MBytes	17.7 %	274.1 MBytes	8.0 %
9 AM	47.9 MBytes	7.9 %	215.1 MBytes	6.3 %
8 AM	0	0.0 %	0	0.0 %
7 AM	0	0.0 %	0	0.0 %
6 AM	0	0.0 %	0	0.0 %
5 AM	0	0.0 %	0	0.0 %
4 AM	0	0.0 %	0	0.0 %
3 AM	0	0.0 %	0	0.0 %
2 AM	0	0.0 %	0	0.0 %
1 AM	0	0.0 %	0	0.0 %
12 AM	0	0.0 %	0	0.0 %
11 PM	0	0.0 %	0	0.0 %
10 PM	0	0.0 %	0	0.0 %
9 PM	0	0.0 %	0	0.0 %
8 PM	0	0.0 %	0	0.0 %
7 PM	0	0.0 %	0	0.0 %
6 PM	0	0.0 %	0	0.0 %
Total				

gearbit
Get it in Gear

What’s Using Bandwidth?

Network Traffic [TCP/IP]: All Hosts - Data Received

Hosts: [All] [Local Only] [Remote Only]

Data: [All] [Sent Only] [Received Only]

Host	Domain	Data		FTP	HTTP	DNS	Telnet	NBios-IP	Mail	DHCP-BOOTP	SNMP	NNTP	NFS/AFS	VoIP	X11	SSH	Gnutella	Kazaa	WinMX	DC++	eDonkey	BitTorrent	Messenger	Other IP
liveupdate.symantecliveupdate.com		1.0 KBytes	0.0 %	0	1.0 KBytes	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b.rad.msn.com		1.0 KBytes	0.0 %	0	1.0 KBytes	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
64.18.3.78		1001	0.0 %	0	0	0	0	0	1001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c.msn.com		987	0.0 %	0	987	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c.msn.com		927	0.0 %	0	927	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
m.2mdn.net		908	0.0 %	0	908	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

gearbit
Get it in Gear

How to contact us at gearbit

Ray Tompkins

info09@gearbit.com
www.gearbit.com