

DATA COMMUNICATIONS MANAGEMENT

WHAT'S ON MY NETWORK? A NETWORK MONITORING AND ANALYSIS TUTORIAL

Betty DuBois, SCE, CNI, CNE, CNX

INSIDE

Step 1: Network Monitoring: What types of traffic are on my network now?;
Now that the tools are in place, what is on my network?; Network Utilization; Application Distribution;
Top Talkers; Step 2: Network Analysis: What could be eliminated from our network and what could be faster?;
What can be eliminated from our network?; Excessive broadcasts; Others;
Conversation Flow Analysis: what could be faster?; Response times; Conversation Flow Errors;
Step 3: Packet Analysis

INTRODUCTION

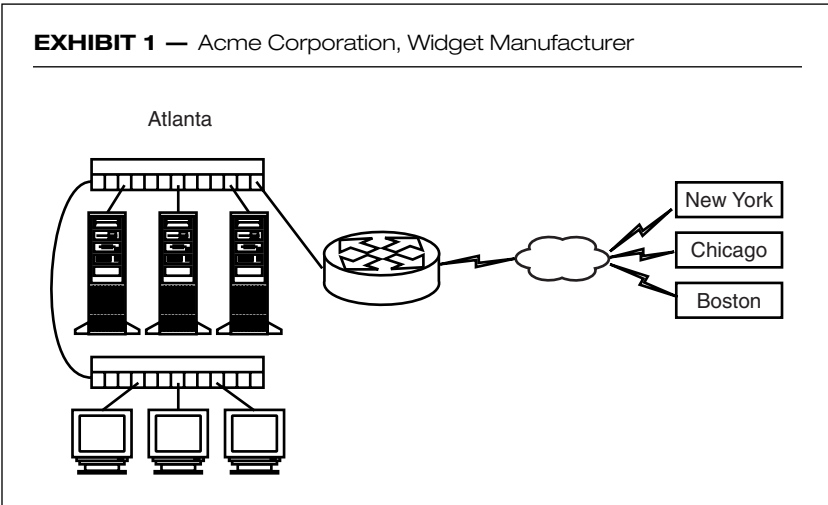
Are you planning to move to a “pure” IP infrastructure in the near future? Do you want to make sure there are no extraneous protocols lurking on your network? Perhaps you want to optimize your applications, or are just interested in how much traffic is really going out to your WAN link. If any of these issues are of concern or importance to you, then you will want to read further.

So often network administrators need to know what is *really* happening on their networks, and a network analysis tool, or *analyzer*, is the only way to obtain this information. First, you need to look at the big picture. This information-gathering process is known as *monitoring*. Before you can start to capture the traffic on your network, you must learn what is really taking place on your network. Monitoring shows you what is actually happening on your network in real-time. For example:

- Protocols being used
- The “top talkers”
- Server response times

PAYOFF IDEA

A network analysis tool, or *analyzer*, is the only way a network administrator can know what is *really* happening on a network. The first step is information gathering, or *monitoring*. Monitoring shows what is actually happening on your network in real-time. Having studied the big picture, next obtain the details, which is called *protocol analysis*, or more commonly, *sniffing*. By copying the datastream into the analyzer software, a network administrator can eliminate extraneous traffic, troubleshoot slow response times, and understand how the conversations really flow.



- Volume of local conversations versus those traveling across the WAN
- Traffic that can be eliminated without affecting the users

Once you have studied the big picture, you can work on obtaining the details, which is called *protocol analysis* or, more commonly, *sniffing*. By copying the datastream into the analyzer software, the network administrator can eliminate extraneous traffic, troubleshoot slow response times, and understand how the conversations really flow.

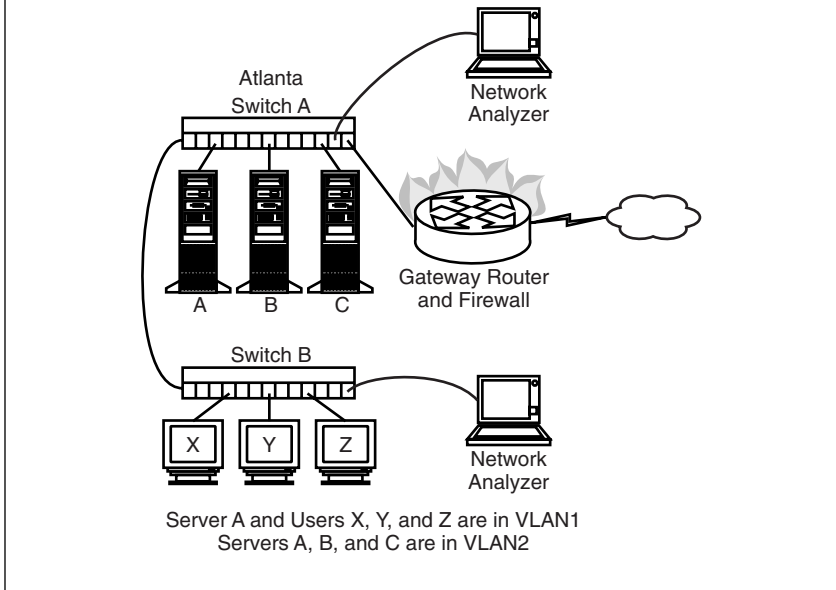
The purpose of this article is to guide you through the placement and use of a network monitor and analysis tool. Exhibit 1 shows Acme Corporation, a fictitious company, which will serve as our sample network for this tutorial. We start the analysis with a 30,000-foot overview of the network, and work our way down to the nitty-gritty details.

STEP 1: NETWORK MONITORING — WHAT TYPES OF TRAFFIC ARE ON MY NETWORK NOW?

This is where network monitoring comes in. Planning is the most important aspect of this process. Before purchasing and positioning a network monitoring and analysis tool, you need to answer a couple key questions:

1. What topologies are on your network?
2. How many segments do you intend to monitor?

Looking at Acme Corporation's network in Exhibit 1, you see there are two different topologies. Ethernet is used for the local area network (LAN), and frame relay is used for the wide area network (WAN). Because not all analyzers are designed for the WAN environment, you

EXHIBIT 2 — Acme Corporate Headquarters

have to determine if you can get all the statistics you need from your telecommunications provider, or if you yourself need to monitor the segment you are interested in. While this article focuses only on LAN analyzers, many WAN analyzer concepts are similar.

The next question to ask yourself is, “How many segments should I monitor?” The days of employing hubs are long gone, and with them the ability to see everything on the network at one time. While switches are now the standard in Ethernet, they create new network monitoring challenges. Atlanta is Acme Corporation’s corporate headquarters, so we focus our efforts there. Exhibit 2 shows multiple Ethernet segments — two switches, the uplink between the switches, and the uplink to the router (gateway segment). In this example, the virtual local area networks (VLANs) on the switches could each be considered their own segment. Good network and VLAN documentation will save a lot of time and frustration when working on this step in the process.

Now that your segments have been identified, the next question to ask is, “Where should I put the analyzer on my network?” The correct answer is always, “It depends.” If your goal is to see all the traffic that goes out over your WAN link, your analyzer should be placed on Switch A (see Exhibit 2) *mirroring* the uplink to the router. Because a switch only forwards a packet to the port where the destination address resides and the analyzer needs to also receive a copy of that packet, the switch must be configured to *mirror* or copy the packets to the analyzer’s own port. Vendors may refer to this

process as *mirroring*, *spanning*, or *port monitoring*. For consistency purposes, this article uses the term “mirroring.”

However, if your goal is to see the traffic from the client’s perspective, the uplink from Switch B should be mirrored and therefore the analyzer should be placed on Switch B. The analyzer should always be placed on the same switch that has been configured to mirror. Trying to mirror an upstream switch would place too great a load on the uplink port.

Is mirroring your only choice when you need to see these packets? No, because (1) mirroring puts a large load on the switch’s processor and buffer, and (2) not all switches support mirroring. There is also the possibility of dropping packets from your analyzer, or worse, overloading the switch and causing it to reboot. Another choice is to use a “tap” at each insertion point and to hang the analyzer off of the tap. The problem with using taps is that there needs to be one tap between each switch and there also needs to be one tap between each server and switch to see the most traffic on your network. Installing that many taps can be cost prohibitive. Also, because our Ethernet segments are running at full-duplex, either you must use a full-duplex analyzer or you need to have two analyzers hanging off each tap so that both the send and receive packets are analyzed.

However, using taps allows you to take the analyzer load off the switch and still obtain correct statistics. But the more ports that are being mirrored, the more your switch must buffer the packets to send down to the analyzer. This can cause a skew in the timings when measuring latency between commands and responses.

Now that the Tools Are in Place, What Is on My Network?

Now that the tools are in place, you can start to monitor our network. Network monitoring is simply doing a baseline with your goal to gather statistics about what is happening on your network. Depending on the analyzer used, many different types of reports are available to display the statistical information. The key information to gather and review is network utilization, protocol distribution, and top talkers. In many cases, the information learned at this level guides the direction for the rest of the network analysis. Each answer leads to the next question. For example, while looking at the utilization statistics, the focus should be on what is happening on your network during peak times. Does the protocol distribution shift, or are you just seeing *more* of the same protocols?

Network Utilization. The first segment being monitored in our sample network is the uplink to the router. Depending on your analyzer, you might only be able to see absolute and average utilization statistics. If your analyzer supports it, being able to see what your network utilization is over a period of time makes it much easier to see when the peak times

occur on your network. This information also helps to plan when the low utilization points occur on your network, so that change control processes can be scheduled. In [Exhibit 3](#) we see that network utilization spikes on the gateway segment after 3:00 pm.

Application Distribution. How long do these spikes last? If the spikes are sustained, the next question to ask yourself is, “What protocols are being used?” Often, what the network administrator *thinks* is happening on the network and what is *really* happening on the network are two very different things.

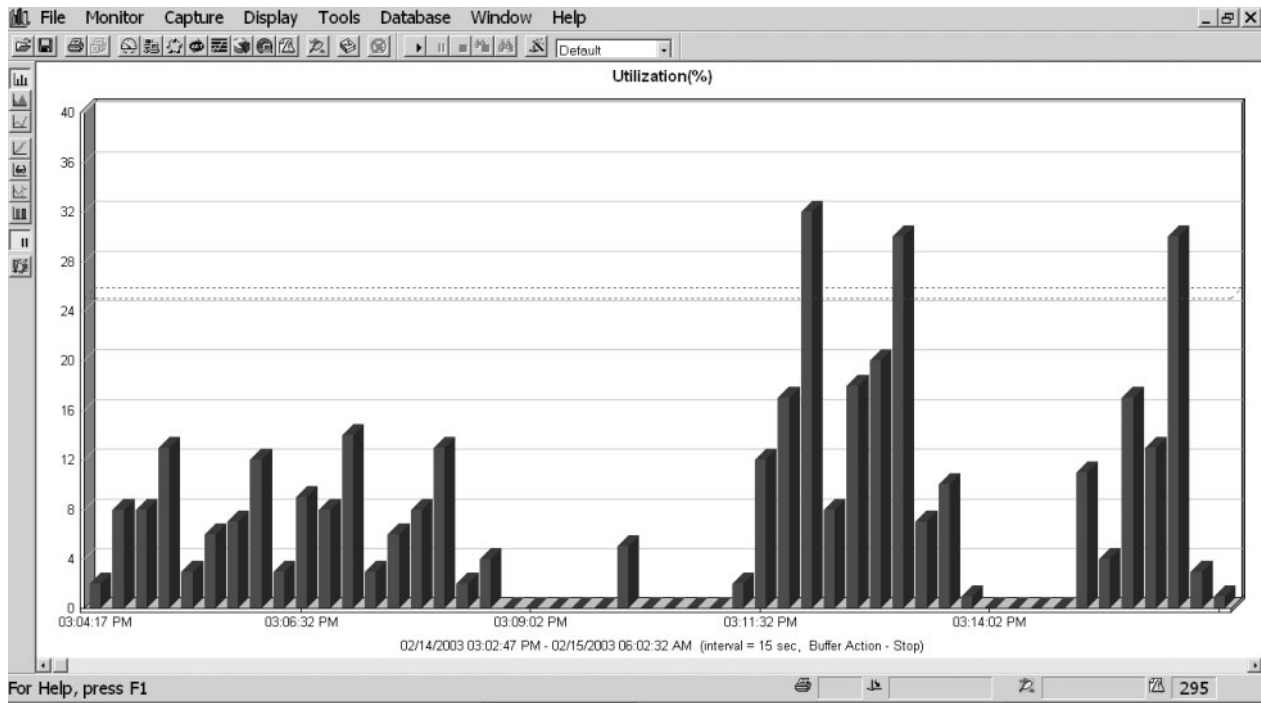
The first segment being monitored on our sample network is the uplink to the router (see [Exhibit 4](#)). We see from the analyzer that HTTP, FTP, Media Player, and something called “Others” account for most of the network traffic. We expect HTTP and FTP traffic because the users have access to the Internet; and now that we know they are listening to music via the WAN link, we can block the Media Player traffic at the firewall. But what type of traffic is “Others”? This is the kind of information you find when you start monitoring your network. There is *stuff* out on your network that you cannot account for, and this *stuff* is a prime target to evaluate when looking to optimize your network. Later, we will capture the packets to find out what these “Others” are.

Top Talkers. Now that you know what protocols are in use, you can look at who is using them the most. While it is very easy to get lost in the great abyss of information that an analyzer can provide, you need to focus your time and efforts on where the greatest impact on our sample network can be made in terms of optimization. This is one of the reasons all analyzers have a “top talkers” report (see [Exhibit 5](#)). This report tells us who is using the most bandwidth. On Acme Corporation’s network, we want to look at the top talkers for both the uplink to the router and for the server farm segments. Because our servers are all on the same VLAN, by mirroring the VLAN2 we are able to see not only all the client-to-server traffic, but also any server-to-server traffic, such as database replication. By looking at the IP address, we can also tell how much of the traffic is from the Atlanta users and how much is from users located at other sites.

Once we look at who is accessing our servers, we can focus on the gateway segment and determine:

- Who are the “top talkers” to our WAN link?
- Is the traffic predominately initiated from our other sites querying our CRM database, or is it coming from inside the company by users surfing the Internet?

EXHIBIT 3 — Sniffer Portable Utilization History Report



The answers to these questions will help you decide if you need to increase the speed of the WAN link, or if you need to load balance your database server.

Being able to see not only the “top talkers” in [Exhibit 5](#) but also the top conversations in [Exhibit 6](#) will help you make these decisions.

STEP 2: NETWORK ANALYSIS — WHAT COULD BE ELIMINATED FROM OUR NETWORK AND WHAT COULD BE FASTER?

Now that you know the network utilization, application distribution, and top talker statistics, and therefore where to concentrate your optimization efforts, you can start to do the analysis. This will involve both determining what traffic is superfluous and what can be done to improve access to the information your staff and customers need.

What Can Be Eliminated from the Network?

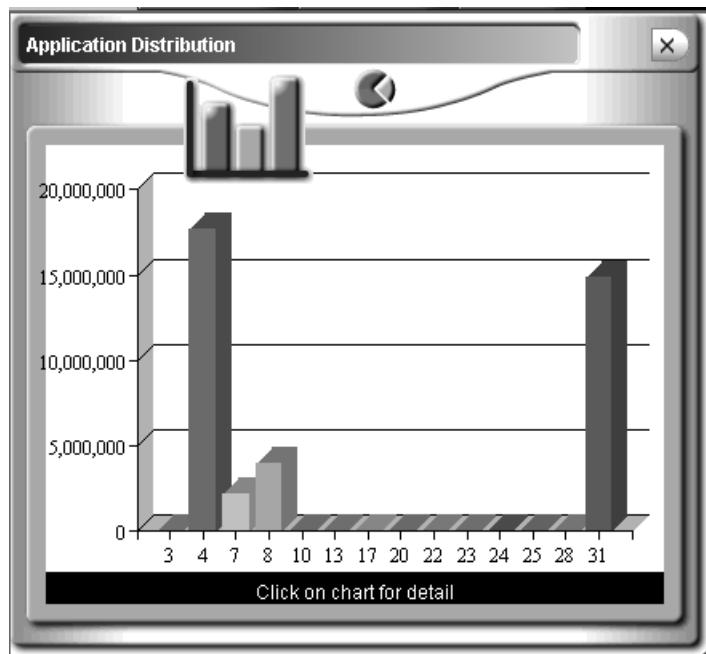
The ultimate goal of any network analysis is to make the network response time faster. With that in mind, any unnecessary traffic you locate and eliminate will move you closer toward this goal.

Excessive Broadcasts. One of the most common network issues is broadcast traffic, which is traffic sent to all devices on a segment. There are many recommendations for what should be the broadcast ratio on a network. It all depends on the types of traffic on your network. For example, in places where many screens are updated with the same information, such as a stock brokerage office, extremely high multicast traffic (i.e., traffic sent to a group of devices using a special address) is considered normal. However, for our sample client/server-based network, the recommended ratio is 10 percent broadcast and multicast traffic.

To calculate this ratio, simply divide the sum of the broadcast packets plus the multicast packets by the packets received (see [Exhibit 7](#)). The result for our sample network is 23 percent, which is much greater than our recommended ratio of 10 percent. Later, when discussing packet analysis, we will examine different types of broadcast and multicast packets to see what can be optimized. Our goal is to determine what services are being advertised on the segment that no one is using.

Others. There are a multitude of different types of conversations that can be analyzed on a network. The key is to focus on the ones that are most critical to the business. Acme Corporation’s corporate office in Atlanta also houses a Web server where the public can purchase their widgets online and download installation and maintenance information about their widgets. Therefore, although we expected to see high levels of HTTP and FTP traffic when monitoring the uplink to the router, we also saw a large percentage of “Others” traffic. Remember that “Others”

EXHIBIT 4 — AppDancer F/A Application Distribution



Statistics

	Application	Bytes	Frames		Application	Bytes	Frames		Application	Bytes	Frames		Application	Bytes	Frames
1	AIM	0	0	11	MS SQL	0	0	21	SIP	0	0	31	Others	14871329	15808
2	BOOTP	0	0	12	Napster	0	0	22	SKINNY	436	5				
3	DNS	18537	95	13	NETBIOS	8521	71	23	SMTP	20837	54				
4	FTP	17699187	19272	14	NFS	0	0	24	SNMP	1936	17				
5	Gopher	0	0	15	NNTP	0	0	25	TELNET	10541	127				
6	H.323	0	0	16	Oracle	0	0	26	TFTP	0	0				
7	HTTP	2201133	4457	17	POP	8547	81	27	X Windows	0	0				
8	Media Player	3986396	4307	18	Quicktime	0	0	28	Yahoo Mess...	13949	133				
9	MGCP	0	0	19	Real Player	0	0	29	Gnutella	0	0				
10	MSNP	6158	70	20	RIP	7470	83	30	SAP R/3	0	0				

EXHIBIT 5 — AppDancer F/A IP Hosts

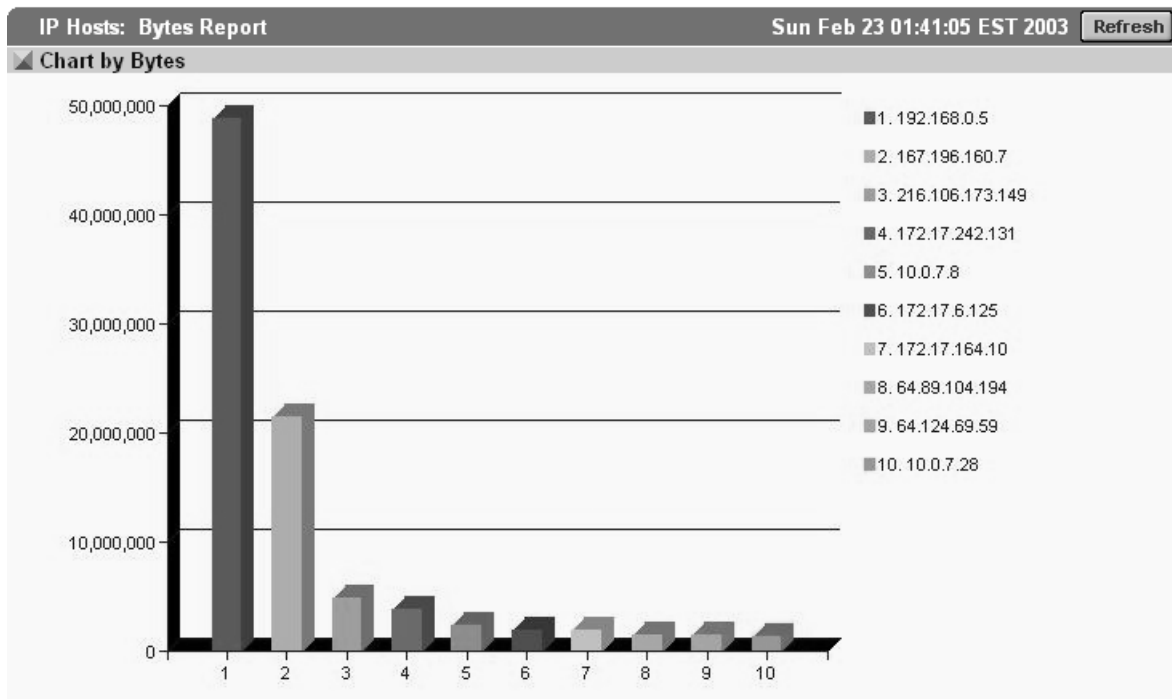


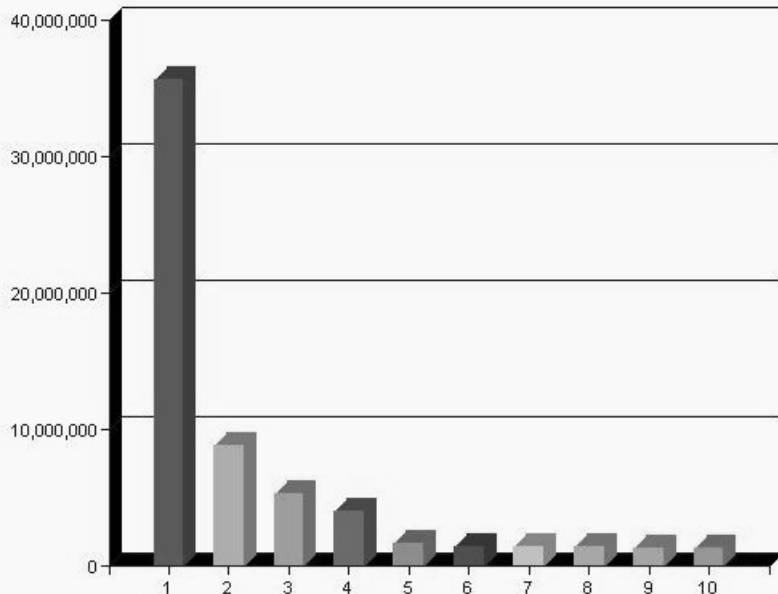
EXHIBIT 6 — AppDancer F/A IP Flows

IP Flows: Bytes Report

Sun Feb 23 01:39:14 EST 2003

[Refresh](#)

Chart by Bytes



- 1. 192.168.0.12/20 <--> 192.168.0.5/1386
- 2. 192.168.0.12/20 <--> 192.168.0.5/1389
- 3. 192.168.0.12/20 <--> 192.168.0.5/1388
- 4. 216.106.173.149/80 <--> 167.196.160.7/46911
- 5. 10.0.7.8/1034 <--> a18.CMS.CO.COM/1433
- 6. 167.196.160.7/44765 <--> 128.227.191.143/81
- 7. 172.17.6.125/2232 <--> 172.17.242.131/3108
- 8. 172.17.242.131/3112 <--> 172.17.164.10/573
- 9. 172.17.164.10/1110 <--> 172.17.242.131/311
- 10. 172.17.242.131/3108 <--> 172.17.6.125/567

EXHIBIT 7 — Etherpeek Network Statistics

Duration:	00:04:56
Packets received:	122,827
Bytes received:	19,305,015
Multicast:	655
Broadcast:	27,109
Error Type	
Packets	
Total:	0
Gauge Value	

are protocols that do not fit into the preconfigured list of defined port numbers on our analyzer. Because no analyzer is configured with *all* of the well-known port numbers, finding out what the “Others” are is usually simply a matter of capturing the data, finding the port number in the TCP or UDP header, and looking it up on <http://www.iana.org/assignments/port-numbers>. Unlike monitoring, which is looking at statistics, capturing allows us to bring an entire copy of a packet into the buffer of the analyzer for detailed analysis and decode. After capturing packets on Acme Corporation’s gateway segment long enough to see the “Others” increment in the Application Distribution in [Exhibit 4](#), we can then stop and display the buffer. Now we can filter the trace file for the “Others” (see [Exhibit 8](#)). By sorting the conversations by byte count, we can focus on those conversations with the greatest impact — in this case, the top three conversations. The first conversation used port 1494, which is the well-known port for the Citrix client. Acme Corporation’s users access an SQL database to look up customer information. The database is housed on a server in Atlanta, and all locations access it. Because staff in the other offices access the customer database using a Citrix client, this traffic is expected and normal. We now only need to add this port number to our analyzer so that it will be delineated in the future. The next conversation uses port 6346, the well-known port number for Gnutella. Gnutella is certainly not authorized use of Acme Corporation’s network resources, so we will not only add ports 6346 and 6347 (Gnutella Router) to our analyzer so they can be watched for in the future, but we will configure the firewall to block the traffic from those ports as well. The last conversation we will look at uses port 1345. Because this is not on the Internet Assigned Number Authority’s (IANA) list, we will have to look at the ASCII representation of the hexadecimal code of the packet itself to look for clues to determine what the traffic really is. In [Exhibit 9](#) we see that the words `Ghost` and `Config Server` appear in the ASCII translation. Because this packet was sent to a multicast address, we now know it is the client checking for updates from the Ghost Server. We can

EXHIBIT 8 — AppDancer F/A Classic Tab, Port 1494

Transmission Control Protocol, Src Port: 1356 (1356), Dst Port: 1494 (1494), Seq: 5685136

- Source port: 1356 (1356)
- Destination port: 1494 (1494)
- Sequence number: 568513663
- Header length: 28 bytes
- Flags: 0x0002 (SYN)
- 0... .. = Congestion Window Reduced (CWR): Not set
- .0.. = ECN-Echo: Not set

Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Text
0000:	00	02	A5	FB	49	00	00	06	5B	32	57	1B	08	00	45	00I...[2W...E.
0010:	00	30	11	19	40	00	80	06	D0	D9	0A	B5	01	74	0A	B5	.0..@.....t..
0020:	01	F8	05	4C	05	D6	21	E2	D4	7F	00	00	00	00	70	02	...L..!.....p.
0030:	40	00	28	C6	00	00	02	04	05	B4	01	01	04	02			@.(.....

EXHIBIT 9 — AppDancer F/A Classic Tab, Port 1345

🔍 📁 User Datagram Protocol, Src Port: 1028 (1028), Dst Port: 1345 (1345)

📄 Source port: 1028 (1028)

📄 Destination port: 1345 (1345)

📄 Length: 158

📄 Checksum: 0xedaf (correct)

📄 Data (150 bytes)

Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Text
0000:	01	00	5E	37	96	D0	00	B0	D0	D9	6D	D3	08	00	45	00	..^7.....m...E.
0010:	00	B2	2E	FA	00	00	0A	11	C2	A6	BF	06	83	8C	E5	37\$.....7
0020:	96	D0	04	04	05	41	00	9E	ED	AF	24	02	02	09	01	33A....\$....3
0030:	06	4C	6F	63	61	74	65	22	01	00	24	02	02	09	02	33	.Locate"..\$....3
0040:	07	50	72	6F	64	75	63	74	0C	02	24	02	02	09	03	33	.Product..\$....3
0050:	05	47	68	6F	73	74	0C	03	24	02	02	09	04	33	09	43	.Ghost..\$....3.C
0060:	6F	6D	70	6F	6E	65	6E	74	0C	04	24	02	02	09	05	33	omponent..\$....3
0070:	0D	43	6F	6E	66	69	67	5F	53	65	72	76	65	72	0C	05	.Config_Server..
0080:	24	02	02	09	06	33	04	4E	61	6D	65	0C	06	20	14	A1	\$....3.Name... ..
0090:	10	9C	BC	63	6B	94	F9	41	70	A2	0B	B9	A8	A8	D7	B0	...ck..Ap.....
00A0:	3E	12	3D	24	02	02	09	07	33	09	43	68	61	6C	6C	65	>.= \$....3.Challe
00B0:	6E	67	65	0C	07	20	08	78	F0	C6	D7	E1	7C	74	7C	02	nge.. .x.... t .

investigate perhaps changing the timings in the software to see if checking for updates can be done less frequently, but this traffic is not something we want to eliminate entirely.

These are just a few examples of the kinds of things you might find once you really start digging into your network.

Conversation Flow Analysis: What Could Be Faster?

Next we can focus on what conversation flows can be optimized, but first we have to move our analyzer to the user segment. When users complain that “the network is slow,” they mean that the access time to the applications they need is slow. By mirroring the port the user is attached to, we can get as close as we can to their actual experience and look at the response times.

Response Times. There are two types of response times (see [Exhibit 10](#)). Delta time is the time between a command and the response — literally the time between when the packets passed through the analyzer. Relative time (Rel. Time) is the cumulative sum of the delta times. It tells us how long the conversation lasted.

If you see a large gap in the delta times, you need to determine which side is causing the delay. If the large delta time occurs after a server response, it could simply imply that the user is reading what is on his screen before asking for more information. If the gap in the delta time is from the server response, you must delve deeper into the reason for the delay. Either way, you now have a baseline of response times from which to work. It is recommended that you look at the common processes on your network and get a baseline of response times for each of your different applications. Although the analyzer can also be shipped to each office to get a baseline of response times for each office accessing the servers across the WAN, a distributed network analyzer greatly simplifies the process. We can have a distributed network analyzer in each location and access each one from a single console in Atlanta. Then, when users are experiencing a slowdown, we can have one analyzer at the server and one at the user segment, and compare them to determine where the latency is occurring.

For example, if we are sniffing only on the users' segment, and we see a command travel down the wire and the delta time for the response is very large, we still do not know if the network is causing the latency or if the server or the application is causing the latency. However, by looking at the same conversation on the server segment, we see one of two scenarios: either (1) the delta time between the command and the response is small, which means the network is inducing the latency, or (2) the delta time between the command and the response is large, which means either the server or the application is at fault. Remember

EXHIBIT 10 — AppDancer F/A Application Flow

Conversation		Statistics	
192.168.0.2 <--> www.fedex.com		Delta Time	Rel. Time
●	POST /cgi-bin/tracking	0.000000	0.000000
●	200 - Ok	0.669636	0.669636
●	GET /images/ascend/shared/spacer.gif	1.427112	2.096748
●	304 - Use local copy	0.044488	2.141236
●	GET /images/ascend/shared/corp_logo.gif	0.431292	2.572528
●	304 - Use local copy	0.068455	2.640983
●	GET /images/shared/shared_bullet_triangle.gif	0.478656	3.119639
●	304 - Use local copy	0.053905	3.173544
●	GET /images/us/adobjects/us_adobjects_insightadobj.gif	0.132765	3.306309
●	304 - Use local copy	0.049381	3.355690
●	GET /images/shared/shared_dot_clear.gif	0.268849	3.624539

that delta times are based on the difference between the timestamps of the two packets *as they pass the analyzer (not when they left the originating device)*, so a small delta time at the server segment with a large delta time at the user segment means the command was delayed getting to the server and response was delayed going back to the client. Therefore, the network is causing the latency. We would then perform a Trace Route to see where within the path the latency is being induced.

If the delta time between the command and the response is large, the issue is either the server or the application. To determine if the server is causing the problem, we would look at how many conversations the server was currently involved in. If it is a large number of conversations, perhaps some tasks could be offloaded, or we could check the utilization on the server itself to see if the hardware needs to be upgraded. AnalogX's NetStat Live utility is an excellent tool for helping us with both of these tasks.

To be able to prove it is the application causing the problem, we need to look at the TCP acknowledgments. If we see a TCP ack after the user command, and then later see the server response, we can prove that the TCP stack on the server received the data but the application could not process the request fast enough. That is, it is not the network's fault. Notice the large delta time in packet 6 in [Exhibit 11](#), yet the server's TCP stack acknowledges the request packet in packet 5, then 87 seconds later, the upper layer of the server sends the response of 244 bytes.

However, before pointing fingers at the application developer, we need to check to see if the server requested data from another server. At Acme Corporation, a Citrix server makes calls to the SQL server to look up customer data before responding to the client. If we filter for the user and Citrix server conversation by the TCP port pairing, the call to the SQL server would be filtered out and we could be led to the wrong conclusion.

When doing this type of in-depth analysis, having two or more analyzers makes a tremendous difference in being able to see the conversation at each point of contact and makes it a lot easier to prove your case when participating in "whose fault is it"-type meetings.

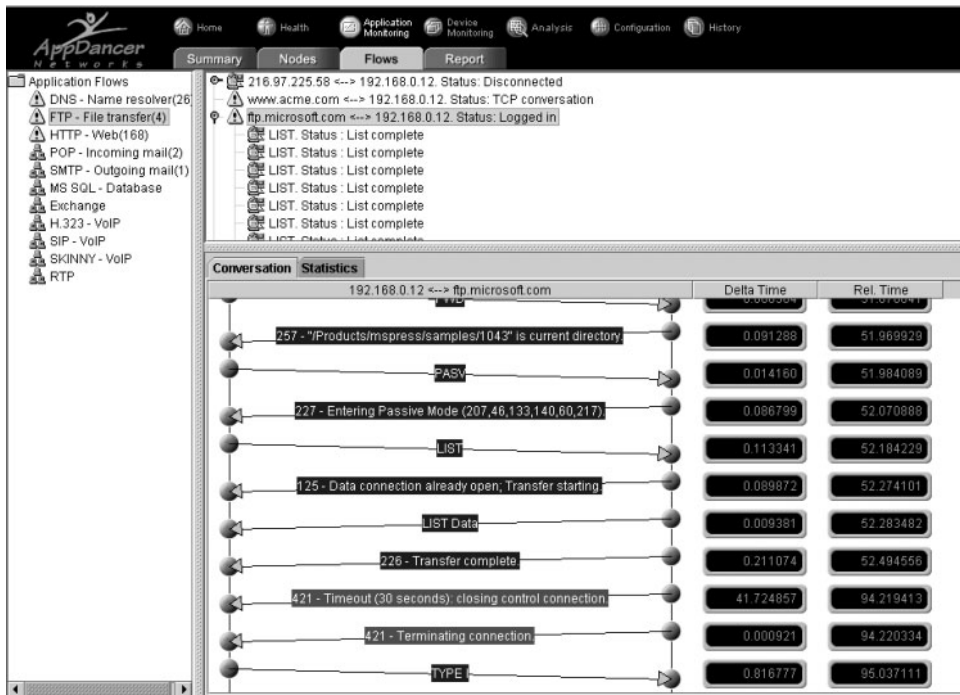
Conversation Flow Errors

Having looked at response times, now we can look at specific errors for different conversations. [Exhibit 12](#) shows an FTP download from ftp.microsoft.com where the server timed the user out after only 30 seconds (error code 421). We need to decide how long the timeout should be for Acme Corporation's FTP server. Because the site is for our customers, we want to make it as easy as possible for them to download at their leisure, yet we do not want ports open forever.

EXHIBIT 11 — AppDancer F/A Classic Summary Pane

Summary View									
Nodes View	App Flows	IP Statistics	IP Flows	Alarm Log	Filter	Classic			
No.	Src. Addr	Dst. Addr	Summary				Delta Time	Rel. Time	Len
1	Client	Server	1108 > 1535 [SYN] Seq=87414 Ack=0 Win=8192 Len=0				0.000000	0.000000	
2	Server	Client	1535 > 1108 [SYN, ACK] Seq=2094630441 Ack=87415 Win=8760 Len=0				0.000366	0.000366	
3	Client	Server	1108 > 1535 [ACK] Seq=87415 Ack=2094630442 Win=8760 Len=0				0.000110	0.000476	
4	Client	Server	1108 > 1535 [PSH, ACK] Seq=87415 Ack=2094630442 Win=8760 Len=360				0.000346	0.000822	4
5	Server	Client	1535 > 1108 [ACK] Seq=2094630442 Ack=87775 Win=8400 Len=0				0.000112	0.000934	
6	Server	Client	1535 > 1108 [PSH, ACK] Seq=2094630442 Ack=87775 Win=8760 Len=244				87.065548	87.066482	2
7	Client	Server	1108 > 1535 [PSH, ACK] Seq=87775 Ack=2094630686 Win=8516 Len=4				0.000400	87.066882	
8	Server	Client	1535 > 1108 [ACK] Seq=2094630686 Ack=87779 Win=8760 Len=0				0.000158	87.067040	
9	Server	Client	1535 > 1108 [PSH, ACK] Seq=2094630686 Ack=87779 Win=8760 Len=14				0.006146	87.073186	
10	Client	Server	1108 > 1535 [PSH, ACK] Seq=87779 Ack=2094630700 Win=8502 Len=48				0.000636	87.073822	1
11	Server	Client	1535 > 1108 [PSH, ACK] Seq=2094630700 Ack=87827 Win=8760 Len=2				0.000544	87.074366	
12	Client	Server	1108 > 1535 [PSH, ACK] Seq=87827 Ack=2094630702 Win=8500 Len=8				0.002180	87.076546	
13	Server	Client	1535 > 1108 [PSH, ACK] Seq=2094630702 Ack=87835 Win=8760 Len=60				0.000356	87.076902	1
14	Client	Server	1108 > 1535 [PSH, ACK] Seq=87835 Ack=2094630762 Win=8440 Len=18				0.000186	87.077088	
15	Server	Client	1535 > 1108 [PSH, ACK] Seq=2094630762 Ack=87853 Win=8760 Len=38				0.002582	87.078620	

EXHIBIT 12 — AppDancer F/A AppFlow FTP Error



[Exhibit 13](#) shows an error on our HTTP server. The client requested a graphic file and, after almost three (3) seconds, had not received a response and therefore asked again. After the second request, the file was downloaded so we know that the file was available. Because almost all upper-layer retransmissions are caused by a physical issue, we would first check the statistics on the switch where the Web server is connected to see if there were any physical layer errors that might cause a TCP retransmission. If there were no physical errors, then we would check the statistics on the router to see if any packets were being dropped due to our exceeding our Committed Information Rate (CIR).

[Exhibit 14](#) shows an error in an SQL conversation. The response from the server tells us that “CCSEvents” is an invalid object name and that the Statement(s) could not be prepared, which is obviously not a network issue. Armed with this information from our analyzer, we now can forward the trace file to our SQL developers and request a fix. Because the error messages are in such an easy-to-read format, we are more likely to get a cooperative response from the SQL developers than if we just told them that the SQL is broken.

STEP 3: PACKET ANALYSIS

Although our analyzers have given us a lot of useful information, sometimes you just have to look at the hexadecimal code. Remember that we had postponed looking at our excessive broadcast traffic until now. To capture broadcast and multicast traffic, we only need to plug into a switch port of the VLAN we are interested in. No mirroring is necessary because broadcast and multicast traffic is flooded throughout the VLAN. In the packet shown in [Exhibit 15](#), address 192.168.0.5 sends a WINS request for the ICS workstation because a printer share had been configured on the workstation. That printer has been out of service for three months. Yet at least 25 stations have been broadcasting a Name Query for it every 0.7 seconds!

Configuration issues like this are impossible to find without an analyzer and can easily bog down a network if not detected.

SUMMARY

This article described some of the many factors to consider when performing network monitoring and analysis. The process requires not only knowledge of the devices on your network, but also knowledge of the traffic flows and the protocols involved. We also learned that for every question answered by our analysis, two or three more questions emerge.

EXHIBIT 13 — AppDancer F/A AppFlow HTTP Error

The screenshot displays the AppDancer Network interface. The top navigation bar includes icons for Home, Health, Application Monitoring, Device Monitoring, Analysis, Configuration, and History. Below this, there are tabs for Summary, Nodes, Flows, and Report. The left sidebar shows a tree view of Application Flows, with 'HTTP - Web(166)' selected. The main area shows a list of flows, with one flow highlighted: '192.168.0.12 <-> www.acmetech.com 2 GET requests; First URL /images/mac.gif'. Below this, a 'Conversation Statistics' table provides details for the selected flow.

Application Flows:

- Application Flows
 - DNS - Name resolver(19)
 - FTP - File transfer(1)
 - HTTP - Web(166)**
 - POP - Incoming mail(1)
 - SMTP - Outgoing mail
 - MS SQL - Database
 - Exchange
 - H.323 - VoIP
 - SIP - VoIP
 - SKINNY - VoIP
 - RTSP

Flow Details:

- 192.168.0.12 <-> 203.96.214.18 8 GET requests; First URL /img/bw_hammer-bg-small.gif
- 192.168.0.12 <-> www.acmetech.com GET /asw.html
- 192.168.0.12 <-> www.acmetech.com 2 GET requests; First URL /images/prod_banner.gif
- 192.168.0.12 <-> www.acmetech.com 2 GET requests; First URL /images/quickfacts.gif
- 192.168.0.12 <-> www.acmetech.com 2 GET requests; First URL /images/mac.gif**
 - URL: /images/mac.gif
 - URL: /images/mac.gif
- 192.168.0.12 <-> www.acmetech.com GET /images/gr_bullet.gif
- 192.168.0.12 <-> www.acmetech.com GET /pricelist.html
- 192.168.0.12 <-> www.acmetech.com 2 GET requests; First URL /images/order_banner.gif
 - URL: /images/order_banner.gif
 - URL: /images/verisignsealblack.gif

Conversation Statistics:

192.168.0.12 <-> www.acmetech.com	Delta Time	Rel. Time
GET /images/mac.gif	0.000000	0.000000
GET /images/mac.gif (Retry)	2.935653	2.935653
200 - OK	0.064426	3.000079

EXHIBIT 14 — AppDancer F/A AppFlow SQL Error

AppDancer NETWORKS

Home Health Application Monitoring Device Monitoring Analysis Configuration History

Capture Post Analysis

C:\Documents and Settings\Administrator\Desktop\Atlanta.sql and exchange.cap Frames: 9192

Summary View Nodes View App Flows IP Statistics IP Flows Alarm Log Filter Classic

Application Flows

- DNS - Name resolver
- FTP - File transfer
- HTTP - Web(65)
- POP - Incoming mail
- SMTP - Outgoing mail(2)
- MS SQL - Database(62)
- Exchange(52)
- H.323 - VoIP
- SIP - VoIP
- SKINNY - VoIP
- RTP

View classic panel using selected flow as filter

192.168.8.68 <--> 192.168.9.10 Current query: select substring('NY',status/1024&1+1,1) from master..sysdatabases where name=DB_NAME()

192.168.9.51 <--> 192.168.9.10 Current query: Select * from BranchNode WHERE (type=4) ORDER BY Type DESC, NodeName ASC

192.168.9.51 <--> 192.168.9.10 Current query: RPC exec AVISP_GetUserAccessData

192.168.9.51 <--> 192.168.9.10 Current query: select substring('NY',status/1024&1+1,1) from master..sysdatabases where name=DB_NAME()

192.168.9.51 <--> 192.168.9.10 Last error: Statement(s) could not be prepared. Current query: exec sp_unprepare

select substring('NY',status/1024&1+1,1) from master..sysdatabases where name=DB_NAME()

RPC exec [ePO_PDC].DBO.AMISP_FILTERASC,1

Conversation Statistics Error Log

Frame #	192.168.9.51 <--> 192.168.9.10	Delta Time	Rel. Time	Len
7314	exec sp_unprepare	0.019353	225.372106	219
7315	Response From Server	0.000794	225.372900	328
7315	Invalid object name 'CCSEvents'	0.000000	225.372900	328
7315	Statement(s) could not be prepared	0.000000	225.372900	328
7318	RPC exec AVISP_GetSQLServerTime	2.201425	227.574325	115
7319	Response From Server	0.000372	227.574697	114
7319	Stored procedure done	0.000000	227.574697	114
7320	RPC exec AVIDALSP_GetNodeIDFilterQueries	0.000563	227.575260	148
7321	Response From Server	0.000570	227.575830	209

EXHIBIT 15 — AppDancer F/A Classic, Name Query

No.	Src. Addr	Dst. Addr	Len	Protocol	Summary
1	192.168.0.1	192.168.0.255	86	RIPv2	Response
2	192.168.0.5	192.168.0.255	92	NBNS	Name query NB ICS <00>
3	192.168.0.5	192.168.0.255	92	NBNS	Name query NB ICS <00>
4	192.168.0.5	192.168.0.255	216	BROWSER	Get Backup List Request

```

0... .. = Query
.000 0... .. = Name query
... ..0. .... = Message is not truncated
... ..1 .... = Do query recursively
... .. ..1 .... = Broadcast packet
Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 0
Queries
ICS <00>: type NB, class inet
Name: ICS <00> (Workstation/Redirector)
Type: NB
Class: inet
  
```

Address	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	Text
0000:	FF	FF	FF	FF	FF	FF	00	E0	00	97	6C	EC	08	00	45	001...E.
0010:	00	4E	11	BA	00	00	80	11	A6	90	C0	A8	00	05	C0	A8	.N.....
0020:	00	FF	00	89	00	89	00	3A	B6	F7	80	B0	01	10	00	01
0030:	00	00	00	00	00	00	20	45	4A	45	44	46	44	43	41	43EJEDFDCAE
0040:	41	43	41	43	41	43	41	43	41	43	41	43	41	43	41	43	ACACACACACACAC
0050:	41	43	41	43	41	41	41	00	00	20	00	01					ACACAAA. . .

References

- AppDancer Networks — AppDancer F/A.
- Compuware Corporation — Application Vantage.
- Sniffer Technologies — Sniffer Pro.
- Wildpackets — Etherpeek.

Betty DuBois, SCE, CNI, CNE, CNX, is the president of Cornerstone Professional Services, a Powder Springs, Georgia-based network consulting firm dedicated to getting networks running at their peak efficiency. She has more than ten years of networking, protocol analysis, and training experience. Additional company and services information is available at www.CStonePros.com. Betty can be reached via e-mail at Betty@CStonePros.com.