TCP/IP Troubleshooting Tips & Tools

Gordon Webber
William Data Systems

August 2011

Gordon.Webber@willdata.com

WILLIAM DATA SYSTEMS

AGENDA . . . . . .

SHARE Technology · Connections · Results

- **Know Your Network**
- **Action Plans / Problem Determination**
- **Tools – General Usage**
- **Understanding the Common Tools**
  (ping, traceroute, netstat, nslookup, …)
- **Problem Diagnosis Tips**

WILLIAM DATA SYSTEMS

Know Your Network! . . .

SHARE Technology · Connections · Results

- In order to manage any network successfully, you must be aware of the topology.
- Before any successful, and timely, problem resolution can be attempted, a (current !) network diagram is ***essential***.
- The diagram (and associated documentation) should indicate all nodes and all possible paths, and detail the subnets, addresses and software (especially versions) available at each node.
- *Only then is it possible to create an appropriate* ***action plan****…*

WILLIAM DATA SYSTEMS

Action Plans . . . . .

SHARE Technology · Connections · Results

- **Where to Start?** - First, ***identify the problem***. This will determine the right tools to use, and the right place to start testing from (**! "Top-down" or "Bottom-up" !**). Progressive testing may be needed to isolate the problem area.

Misinformation Anecdote

- Network problems usually fall into two or three categories:-
  - **No connection can be made.**
  - **Connections can be made, but are unstable, OR , not all functions operate.**
  - **Connections are stable but performance is poor.**

SHARE in Orlando 2011

WILLIAM DATA SYSTEMS

Action Plans . . . . .

SHARE Technology · Connections · Results

Connectivity issues can be caused by:-

Application errors
Failed network connections
Bad configuration/changes
Hardware failures
Failed bind
Power failures
Security restrictions

Performance issues can be caused by:-

Insufficient bandwidth
Bottlenecks
Priorities
Retries
Broadcasts
Congestion
Routing
Fragmentation
Application errors
Switch faults

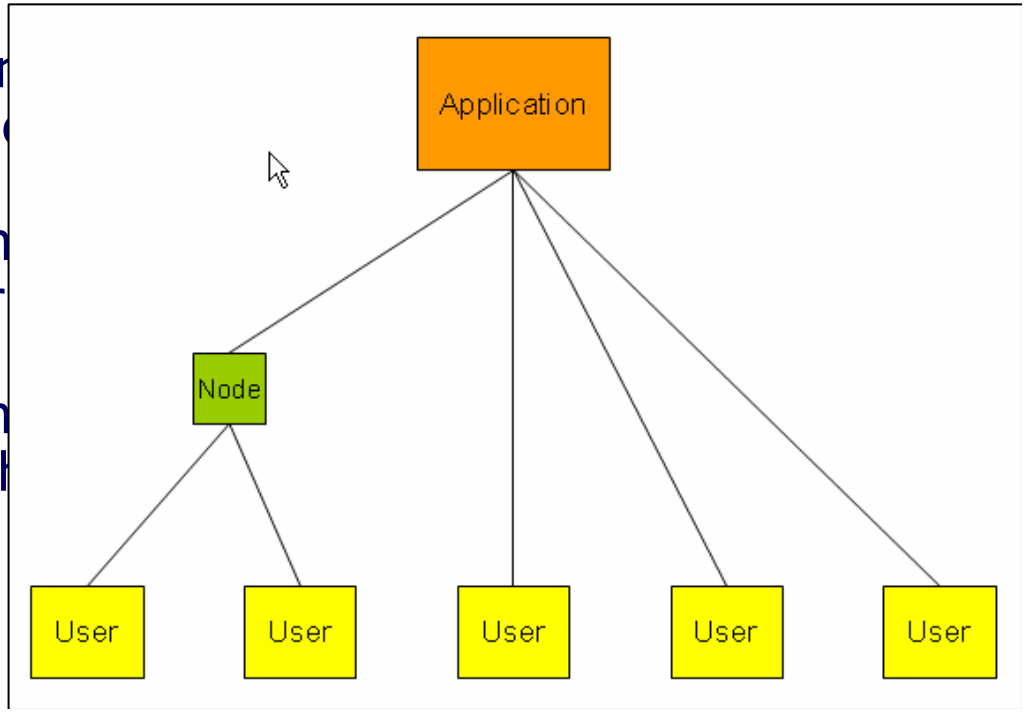
Action Plans . . . . .

1. **Investigate (*ALL*) error messages** – these may indicate the nature and location of the failure [**e.g.** "ttl" expired**,** no path available**,** packet size too large ("nofragment" is on)].

!! Syslogd !!

2. **Classify the error** – ask what works and what doesn't, and for whom . . .

- Problems affecting or
  physical (e.g. ch
- Problems affecting m
  to be the networ
- Problems affecting m
  one network path
  application.



Action Plans . . . . .

3. **Test connectivity** (***end-to-end***) – using Ping/Traceroute.
   Be careful to ensure that the packets take the same path as the problem connection (i.e. ensure correct source interface address – you may need to use an "extended" PING).

   - If PING fails, note the location and investigate there.
   - If PING succeeds (note that this is ICMP, the connection probably uses TCP, so this may *NOT* be a conclusive test), try with a TCP PING if available
   - If PING succeeds try again with larger packets, if appropriate.

WILLIAM DATA SYSTEMS

Action Plans . . . . .

SHARE Technology · Connections · Results

For Example: Problem reported as …

"end-user cannot connect to application"

- Starting at the end-user system ensure local physical connections are good, then check the next layer, such as local switch ports, vlans, routers, and even firewalls.
- Then, test each "hop" by progressive steps across the network.
- Then ensure that the system running the required application is connected at the network level ("ping" from that system outbound via the interface in question.

If all these results are good, then the issue is probably with the application and not a network problem!

WILLIAM DATA SYSTEMS

Tools . . . . .

SHARE Technology · Connections · Results

Disclaimer:

The fact that some tools are mentioned in this presentation while other tools are not, in no way implies recommendation of the tools mentioned, nor condemnation of those tools not mentioned.

The purpose of this presentation is simply to make attendees aware that such tools exist, and the attendees should make up their own mind as to the suitability of any tool used on their own system.

WILLIAM DATA SYSTEMS

"Common" Tools . . . . .

SHARE Technology · Connections · Results

"PING" - proves that connectivity exists

"TRACERTE" - discovers the network path (also "tracert")

"NETSTAT" - to locate connection information

```
ALL         - All connections to a stack
ALLConn     - TCP/IP connections
ARp         - Query ARP table or entry information
CONFIG      - Configuration data
COnn        - Active TCP/IP connections (Default)
DEvlinks    - Devices and links
Gate        - Current known gateways
HOme        - Home address list
PORTList    - Display port reservation list
ROUTe       - Display routing information
SOCKets     - Socket interface users and sockets
STATS       - TCP/IP statistics
TCP         - Displays detailed info about the stack
TELnet      - Telnet connection information
```

```
z/OS command format:
--------------------

NETSTAT < Option | Command > <
Target >
          < Output > < (Select >


E.g.:
TSO NETSTAT CONN (PORT 25
TSO NETSTAT TCP TCPIP
```

Note that "**NETSTAT .....(REPORT**" will collect the output to a dataset; for ease of reading or input to a REXX?

WILLIAM DATA SYSTEMS

Other Tools . . . . .

SHARE Technology · Connections · Results

"Nslookup" - test domain name resolution (& **"DIG"**)

"Snmp" - where SNMP is supported, there are many tools available to extract further information (MIB data), once the problem area has been located (e.g. Monitors, such as "**Implex**" for z/OS ; "**iReasoning**" elsewhere)

- - - - -

"TIVOLI" - IBM network tools (Monitor and trace facilities)

"Ctrace" - z/OS trace tool

"EXIGENCE" - WDS trace "expert" system
(now ZTS ! – "ZEN Trace & Solve")

SHARE in Orlando 2011

WILLIAM DATA SYSTEMS

Other Tools . . . . .

SHARE Technology · Connections · Results

"TPing" - ("TurboPing") "PING" using TCP packets

"Tcpdump" - (also Windump & SSLdump) is a packet sniffer found on many (most?) open platforms.

"Ethereal" - open system packet analyser (& **"Wireshark"**)

"Pchar" - is a reimplementation of Van Jacobson's ("Mr Traceroute") **pathchar** utility which analyses the individual hops of a path.

"Netcat" - Netcat is a utility which reads and writes data across network connections. It is a network debugging and exploration tool. (*+ port-scanner !*)

* New *
Ncat from Nmap

"VisualRoute" - path checker and graphical display

"NeoTrace" - (McAfee) Internet locator: enhanced traceroute

....etc

SHARE in Orlando 2011

Tools in Detail . . . . .

"Ping" - "**P**acket **IN**ternetwork **G**roper", is usually ICMP-based, which works if ICMP is allowed to pass. If not permitted, then an application-based ping can be used [e.g. "**APING**" (UDP) or "TPing" (TCP)].

Ping tests by sending out **ICMP Request** packets, and receiving **ICMP Replies**, therefore verifying up to (ISO) **layer 3** . . .

```
C:\>ping 66.249.85.99              ( www.google.co.uk ----- use IP address or URL )
Pinging 66.249.85.99 with 32 bytes of data:

Reply from 66.249.85.99: bytes=32 time=22ms TTL=244
Reply from 66.249.85.99: bytes=32 time=22ms TTL=244
Reply from 66.249.85.99: bytes=32 time=42ms TTL=244
Reply from 66.249.85.99: bytes=32 time=22ms TTL=244

Ping statistics for 66.249.85.99: Packets: Sent=4, Recvd=4, Lost=0 (0% loss),
Approx. round trip times in milliseconds: Min=22ms, Max=42ms, Ave=27ms
```

WILLIAM DATA SYSTEMS

Layers . . . . . .

SHARE Technology · Connections · Results

ISO 7-Layer Network Model

Layer 1: Physical - defines the real hardware.
Layer 2: Data Link - defines the format of data (frame/packet). (MAC)
Layer 3: Network - responsible for routing datagrams. (IP)
Layer 4: Transport - manages data between network and user. TCP/UDP)
Layer 5: Session - defines the format of the data sent.
Layer 6: Presentation - converts to/from local representation of data.
Layer 7: Application - provides network services to the end-users.

TCP/IP 4-Layer (Unix/DoD) Network Model

Layer 1: Link - defines the network hardware and device drivers.
Layer 2: Network - addressing, routing, delivery. (IP / ICMP) (ARP)
Layer 3: Transport - communication; end-to-end integrity. (TCP / UDP)
Layer 4: Application - user applications.
(DNS, arp, telnet, smtp, http, ftp, traceroute….)

ICMP Types/Codes . . . . .

ICMP Types:

0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect
6	Alternate Host Address
8	Echo
9	Router Advertisement
10	Router Solicitation
11	Time Exceeded
12	Parameter Problem
13	Timestamp
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply
30	Traceroute
31	Datagram Conversion Erro
32	Mobile Host Redirect
33	IPv6 Where-Are-You
34	IPv6 I-Am-Here
35	Mobile Registration Requ
36	Mobile Registration Repl
37	Domain Name Request
38	Domain Name Reply

ICMP Codes:

3	**Destination Unreachable**
0	Net Unreachable
1	Host Unreachable
2	Protocol Unreachable
3	Port Unreachable
4	Fragmentation Needed and DF Set
5	Source Route Failed
6	Destination Network Unknown
7	Destination Host Unknown
8	Source Host Isolated
9	Communication with Dest Network Prohibited
10	Communication with Dest Host Prohibited
11	Dest Network Unreachable for Type of Service
12	Dest Host Unreachable for Type of Service
13	Communication Administratively Prohibited
14	Host Precedence Violation
15	Precedence cutoff in effect
11	**Time Exceeded**
0	Time to Live exceeded in Transit
1	Fragment Reassembly Time Exceeded

Ref: "www.iana.org/assignments/icmp-parameters"

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

PING (Windows)

Usage: **ping** [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
[-r count] [-s count] [[-j host-list] | [-k host-list]]
[-w timeout] target_name

Options:

-t		Ping the specified host until stopped. To see statistics and continue - type Control-Break; To stop - type Control-C.
-a		Resolve addresses to hostnames.
-n	count	Number of echo requests to send.
-l	size	Send buffer size.
-f		Set Don't Fragment flag in packet.
-I	TTL	Time To Live.
-v	TOS	Type Of Service.
-r	count	Record route for count hops.
-s	count	Timestamp for count hops.
-j	host-list	Loose source route along host-list.
-k	host-list	Strict source route along host-list.
-w	timeout	Timeout in milliseconds to wait for each reply.

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

PING

```
C:\>ping 66.249.85.55     ← non-existent addresses
Pinging 66.249.85.55 with 32 bytes of data:

Request timed out.
Request timed out.          (or "Destination Unreachable ?)
Request timed out.          (if a return path is available)
Request timed out.

Ping statistics for 66.249.85.55: Packets: Sent=4, Recvd=0, Lost=4 (100% loss),
```

Drawbacks:

- Extra traffic on the network.
- "**Time To Live**" (**TTL**) set to a high value to ensure penetration.
- Network devices **may not allow** Ping/ICMP and may drop its priority.
- May not take the same path as user traffic; delay (latency) reported may ***not*** be representative for the application(s).
- Low feedback on fault and location.

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

TRACEROUTE (Windows)

Usage: **tracert [-d] [-h maximum_hops] [-j host-list]**
[-w timeout] target_name

Options:

-d	**Do not resolve addresses to hostnames.**
-h maximum_hops	**Maximum number of hops to search for target.**
-j host-list	**Loose source route along host-list.**
-w timeout	**Wait timeout milliseconds for each reply.**

- Also uses ICMP ! (although some platforms use UDP)
- Good for spotting "loops" in the routing
- "**Time To Live**" (**TTL***) is incremented for each positive response.
- Each "hop" in the path is identified (Names may be resolved!).
- "Per hop" round-trip delays can be identified.
- **Drawbacks** are similar to those of "Ping".

(* = anti-looping function of TCP/IP)

Tools in Detail . . . . .

TRACEROUTE

```
C:\>tracert 66.249.85.55 ( www.google.co.uk ----- use IP address or URL )

Tracing route to 66.249.85.55 over a maximum of 30 hops

  1    1 ms    1 ms    1 ms  81.144.212.33
  2    7 ms    6 ms    6 ms  62.7.96.41
  3    6 ms    6 ms    6 ms  core2-gig2-1.kingston.ukcore.bt.net [194.72.3.2]
  4    7 ms    7 ms    7 ms  core2-pos7-3.ealing.ukcore.bt.net [62.6.201.42]
  5    7 ms    7 ms    7 ms  core2-pos10-0.redbus.ukcore.bt.net [194.74.65.202]
  6    8 ms    7 ms    8 ms  194.74.65.38
  7    7 ms    7 ms    7 ms  72.14.238.244
  8   16 ms   16 ms   16 ms  216.239.43.91
  9   22 ms   22 ms   22 ms  72.14.232.209
 10    *       *       *     Request timed out.
 11    *       *       *     Request timed out.
 12    *     etc,etc . . .                  <----- default maximum of 30
```

TRACEROUTE should be run in BOTH directions!!

Look for unsuitable (long) routes and high latency

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

TRACEROUTE

Some platforms give status indicators…

!H - Host unreachable. (Destination Net unreachable) The router has no route to the target system.

!N - Network unreachable.

!P - Protocol unreachable.

!S - Source route failed. A router is blocking source-routed packets.

!F - Fragmentation needed. (Check the MTU configuration at the router).

!X - Communication administratively prohibited. Traceroute blocked!

TRACEROUTE can be enhanced by visualization, as is often seen in graphical traceroute tools : **such as . . .**

TraceRoute Tools . . . . . .

VisualRoute - 1

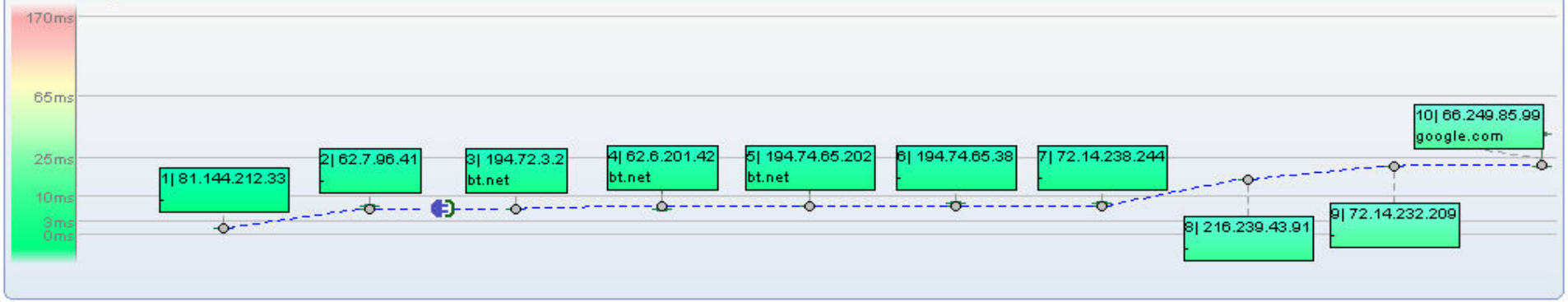
Report for www.google.co.uk [66.249.85.99]

Analysis

This trace was started on 08-Jan-2007 10:28:48. The host 'www.google.co.uk' (known as ff-in-f99.google.com) has been found, and is reachable in 10 hops. Also, it responded to HTTP requests on port 80 (it is running server GWS/2.1, which responded in 431ms). The TTL value of packets received from it is 246.
In general this route offers a good throughput, with hops responding on average within 11ms. The DNS lookup was completed almost instantaneously (less than 2ms - this may be the result of caching).

Map

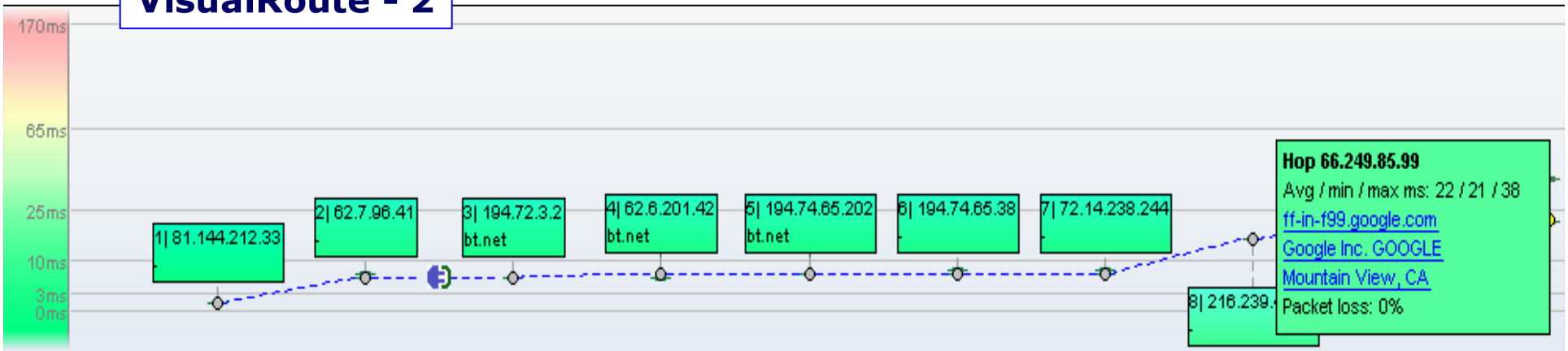
Route Graph



Route Table

TraceRoute Tools . . . . . .

VisualRoute - 2



Hop	%Loss	IP Address	Node Name	Location	Tzone	ms	Graph	Network
0		192.168.1.238	wdsgdw.wds.local	*			0 33	(private use)
1		81.144.212.33	-	(United Kingdom)	*	1		FTIP002842486 William Data Sy
2		62.7.96.41	-	(United Kingdom)	*	6		BTnet
3		194.72.3.2	core2-gig2-1.kingston.ukcc	Kingston, London, UK	*	6		PoP
4		62.6.201.42	core2-pos7-3.ealing.ukcore	Ealing, UK	*	6		Infrastructure
5		194.74.65.202	core2-pos10-0.redbus.ukc	(United Kingdom)	*	7		Private Circuit Customer Networ
6		194.74.65.38	-	(United Kingdom)	*	7		Private Circuit Customer Networ
7		72.14.238.244	-	Mountain View, CA		7		Google Inc. GOOGLE
8		216.239.43.91	-	Mountain View, CA		16		Google Inc. GOOGLE
9		72.14.232.209	-	Mountain View, CA		22		Google Inc. GOOGLE
10		**66.249.85.99**	www.google.co.uk	Mountain View, CA		21		Google Inc. GOOGLE

Roundtrip time to www.google.co.uk, average = 21ms, min = 21ms, max = 21ms -- 08-Jan-2007 10:38:43 (Collapse Table)

Learn more at:

http://www.visualroute.com

TraceRoute Tools . . . . .

PingPlotter

Target Name: **www.google.co.uk**
IP: **66.249.85.99**

0 - 200
201 - 500
501 and up

Hop	PL%	IP	DNSName	Avg	Cur	Graph
1		81.144.212.33	----------	1	1	
2		62.7.96.41	----------	6	6	
3		194.72.3.2	core2-gig2-1.kingston.ukcore.bt.net	6	6	
4		62.6.201.42	core2-pos7-3.ealing.ukcore.bt.net	7	7	
5		194.74.65.202	core2-pos10-0.redbus.ukcore.bt.net	7	7	
6		194.74.65.38	----------	7	8	
7		72.14.238.244	----------	7	7	
8		216.239.43.91	----------	16	16	
9		72.14.232.209	----------	21	21	
10		66.249.85.99	ff-in-f99.google.com	21	22	
			Round Trip:	**21**	**22**	

Data and Image generated by Ping Plotter Freeware (http://www.pingplotter.com)

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

TRACEROUTE –Alternatives

Where the target system is external to the local network, and especially where routing is not available to/from the local network, there are several sites around the World that offer the ability to run "Ping" and "Traceroute" to be instigated by remote control from their web site.

Basically, this is a "proxy" service ; the remote site issuing the test on your behalf.

This is suitable for determining the general availability of the target system (i.e. from anywhere on the Internet), but does not test specific routes.

"**www.samspade.org**" used to be an excellent example of this type of service, but is not currently available in its previous form.

Further directions to such services can be found at :-

"**www.traceroute.org**"

Tools in Detail . . . . .

NETSTAT(z/OS)

NETSTAT < Option | Command > < Target >
< Output > < (Select >

TSO NETSTAT CONN
TSO NETSTAT DEV
TSO NETSTAT TCP TCPIP
TSO NETSTAT SOCK
TSO NETSTAT ROUTE

Also "onetstat"…

Can be issued from either TSO or USS ; the results are the same.

NB. Netstat options will vary depending upon the platform!

Note the following examples from z/OS and Windows. . .

Tools in Detail . . . . .

NETSTAT(z/OS) –"DEV"

```
DevName: LCS1                 DevType: LCS        DevNum: 0E20
  DevStatus: Ready
  LnkName: ETH1                 LnkType: ETH          LnkStatus: Ready
    NetNum:  3    QueSize: 0
    IpBroadcastCapability: Yes
    MacAddress: 000255305115
    ActMtu:  1500
  BSD Routing Parameters:
    MTU Size: 00000
    DestAddr: 0.0.0.0
  Packet Trace Setting:
    Protocol: 253
    SrcPort:  *
    IpAddr:   *
  Multicast Specific:
    Multicast Capability: Y
    Group                 RefCn
    -----                 -----
    224.0.0.1             00000
 Link Statistics:
    BytesIn
    Inbound Packets
    Inbound Packets In Erro
    Inbound Packets Discard
    Inbound Packets With No
```

NETSTAT(z/OS) –"SOCK"

```
MVS TCP/IP NETSTAT CS V1R5        TCPIP Name: TCPIP
Name: APIASHB    Subtask: 007E1048
  Type: Dgram    Status:  UDP          Conn: 00001A1A
    BoundTo: 192.168.1.156..12004
    ConnTo:  *..*
  Type: Stream   Status:  Listen       Conn: 00001A19
    BoundTo: 192.168.1.156..12004
    ConnTo:  0.0.0.0..0
Name: APIASHB    Subtask: 007E12D8
  Type: Dgram    Status:  UDP          Conn: 00001A18
    BoundTo: 192.168.1.156..12000
    ConnTo:  *..*
  Type: Stream   Status:  Listen       Conn: 00001A17
    BoundTo: 192.168.1.156..12000
    ConnTo:  0.0.0.0..0
. . .
```

Tools in Detail . . . . .

NETSTAT (Windows)

Usage: **netstat [-a] [-b] [-e] [-n] [-o] [-p proto] [-r] [-s] [-v] [interval]**

-a **Displays all connections and listening ports.**
-n **Displays addresses and port numbers in numerical form.**
-r **Displays the routing table.**
. . .etc

```
C:\>netstat -a

Active Connections
  Proto  Local Address          Foreign Address        State
  TCP    wdsgdw:epmap           0.0.0.0:0              LISTENING
  TCP    wdsgdw:microsoft-ds    0.0.0.0:0              LISTENING
  TCP    wdsgdw:1028            0.0.0.0:0              LISTENING
  TCP    wdsgdw:1241            0.0.0.0:0              LISTENING
  TCP    wdsgdw:10110           0.0.0.0:0              LISTENING
  UDP    wdsgdw:microsoft-ds    *:*
  UDP    wdsgdw:isakmp          *:*
  UDP    wdsgdw:1033            *:*
  UDP    wdsgdw:4500            *:*
  UDP    wdsgdw:ntp             *:*
  UDP    wdsgdw:1900            *:*
```

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

DNS . . .

In general, it is quite common to seek an IP target using a URL (which acts rather like a PATH name**).**

This entails sending the URL to a "Domain Name Server" (or "Resolver") in z/OS terms) to have the name translated (i.e. a "table lookup") into an IP address (*this may occur locally by use of the "Hos*

The IP address returned is then used to

- -

```
** HOSTS file from Windows :-
( C:\WINDOWS\system32\drivers\etc )

127.0.0.1       localhost
192.168.1.45    lizzie
192.168.1.45    wds.local
192.168.1.45    wds
192.168.1.43    wdsnfs
```

This process may also be performed in reverse; i.e. the DNS server can translate an IP address into a URL !

The use of a URL means that remote services can be failed-over, relocated or rebuilt without the users needing to know!

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

DNS . . .

The global Domain Name System is a hierarchy of servers/services spread across the Internet. At its core is a set of servers that manage the base domains; such as "com", "edu", "gov" ...etc

When a name is "looked up" it happens from right to left - *recursively*.

Take www.google.co.uk ...

- **First the server is located that controls the "uk" domain (**there is an implied "root" service where all top-level servers are known**).**
- **This will indicate the "co.uk" server ; which in turn will indicate the "google.co.uk" server.**
- **The "google.co.uk" server will have IP addresses (*an "A" record*) for web ("www") and mail services (**note: "www" is not the only canonical form used!**)**

NAMED.CONF - lists the "zones" (eg. "google.co.uk")

ZONE FILES - hold the IP addresses

NB. Zone information changed at the bottom of a "layer" is propagated upwards by "Zone Transfer" at preset times.

Tools in Detail . . . . .

NSLOOKUP (Windows)

Usage: **nslookup** **NAME** , or , **NAME1 NAME2** ←(cf z/OS "Resolver")
or ***command***

set *option*

all	[no]debug	[no]d2	[no]defname
[no]recurse	[no]search	[no]vc	domain=NAME
srchlist=N1[/N2/.../N6]		root=NAME	retry=x
timeout=X	type=X	querytype=X	class=X
[no]msxfr	ixfrver=X		

Server NAME

Exit

"Lookup" failure will cause connectivity failure, and symptoms can be mistaken for a routing problem!

- - - -

z/OS often acts as a relay, passing the requests on to a network DNS server.

Tools in Detail . . . . .

NSLOOKUP (Windows)

```
C:\>nslookup

> set debug
> www.google.co.uk
Server:  my.router
Address:  192.168.27.1
------------ (debug information)
Got answer:
    HEADER:
        opcode = QUERY, id = 3, rcode = NOERROR
        header flags:  response, want recursion, recursion avail.
        questions = 1,  answers = 1,  authority records = 0,  additional = 0

    QUESTIONS:
        www.google.co.uk.uk.willdata.com, type = A, class = IN
    ANSWERS:
    ->  www.google.co.uk.uk.willdata.com
        internet address = 212.69.199.183
        ttl = 60 (1 min)

------------
Non-authoritative answer:              <-----( Retrieved from a cache! )
Name:    www.google.co.uk.uk.willdata.com
Address:  212.69.199.183
```

Tools in Detail . . . . .

DIG

Domain Internet Groper: A tool for system administrators; it issues DNS queries and formats/interprets the answers…. Quite popular (*allegedly!*) with hackers…

Usage: dig [@global-server] [domain] [q-type] [q-class] {q-opt}
{global-d-opt} host [@local-server] {local-d-opt}
[host [@local-server] {local-d-opt} [...]]

```
dig @lizzie www.google.co.uk any
; <<>> DiG 9.3.1 <<>> @lizzie www.google.co.uk any
; (1 server found) ; global options:  printcmd ; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16774
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;www.google.co.uk.                IN      ANY
;; ANSWER SECTION:
www.google.co.uk.       86399   IN      CNAME   www.google.com.
;; Query time: 63 msec
;; SERVER: 192.168.1.45#53(192.168.1.45)
;; WHEN: Mon Feb  5 14:11:43 2007
;; MSG SIZE  rcvd: 62                                    . . . . .>
```

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

DIG

```
>. . . . .

dig @lizzie www.google.com any

; <<>> DiG 9.3.1 <<>> @lizzie www.google.com any
; (1 server found) ; global options:  printcmd ; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60773
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 3

;; QUESTION SECTION:
;www.google.com.                        IN      ANY

;; ANSWER SECTION:
www.google.com.         86400   IN      CNAME   www.l.google.com.

;; ADDITIONAL SECTION:
www.l.google.com.       149     IN      A       66.249.93.104
www.l.google.com.       149     IN      A       66.249.93.99
www.l.google.com.       149     IN      A       66.249.93.147

;; Query time: 56 msec
;; SERVER: 192.168.1.45#53(192.168.1.45)
;; WHEN: Mon Feb  5 14:15:13 2007
;; MSG SIZE  rcvd: 100
```

Tools in Detail . . . . .

WHOIS

```
Domain name:
        google.co.uk

    Registrant:
        Google Inc

    Registrant type:
        Non-UK Corporation

    Registrant's address:
        1600 Amphitheatre Parkway
        Mountain View
        CA
        94043
        United States

    Registrant's agent:
        Markmonitor Inc. t/a Markmoni
        URL: http://www.markmonitor.c

Relevant dates:
        Registered on: 14-Feb-1999
        Renewal date:  14-Feb-2009
        Last updated:  17-Jan-2007

    Registration status:
        Renewal request being processed.

    Name servers:
        ns1.google.com
        ns2.google.com
        ns3.google.com
        ns4.google.com
```

Tools in Detail . . . . .

Pchar

Estimates bandwidth, latency and packet loss on network links.

This is a re-working of the "pathchar" utility, written by Van Jacobson and, like traceroute, is based on repeated packet transmission and TTL variation (it can use ICMP or UDP).

It is available for most "***nix**" systems : It works for IPv4 & IPv6.

Traceroute (UDP) knows when it has found its target by using a port number beyond the "normal range"… when ICMP "port unreachable" is returned it's there!

Pchar sends many packets, one hop at a time, with varying the sizes, until the target is reached or the path fails. It calculates the latency from the ICMP message response times, and the throughput per hop from the variance in response speeds. Collectively, this also gives the overall round-trip delay for the whole path.

It is not fool-proof ; it's traffic may ***not*** be allowed ; it is not a "Holy Grail" ; but it does give a good indication!

Tools in Detail . . . . .

Pchar - ./pchar www.google.co.uk

```
pchar to www.l.google.com (66.249.93.104) using UDP/IPv4
Using raw socket input
Packet size increments from 32 to 1500 by 32
46 test(s) per repetition : 32 repetition(s) per hop
Warning: target host did not respond to initial test.
 0: 192.168.1.231 (dhcp-192-168-1-231.uk.willdata.com)
    Partial loss:      0 / 1472 (0%)
    Partial char:      rtt = 0.959029 ms, (b = 0.001150 ms/B), r2 = 0.999475
                       stddev rtt = 0.003212, stddev b = 0.000004
    Partial queueing:  avg = 0.000171 ms (148 bytes)
    Hop char:          rtt = 0.959029 ms, bw = 6954.330709 Kbps
    Hop queueing:      avg = 0.000171 ms (148 bytes)
 1: 81.144.212.33 (81.144.212.33)
    Partial loss:      0 / 1472 (0%)
    Partial char:      rtt = 5.784087 ms, (b = 0.005317 ms/B), r2 = 0.999798
                       stddev rtt = 0.009218, stddev b = 0.000011
    Partial queueing:  avg = 0.002336 ms (667 bytes)
    Hop char:          rtt = 4.825058 ms, bw = 1919.855256 Kbps
    Hop queueing:      avg = 0.002165 ms (519 bytes)
 2: 62.7.96.41 (62.7.96.41)
    Partial loss:      0 / 1472 (0%)
    Partial char:      rtt = 5.824306 ms, (b = 0.005317 ms/B), r2 = 0.999847
                       stddev rtt = 0.008008, stddev b = 0.000010
    Partial queueing:  avg = 0.001486 ms (667 bytes)
    Hop char:          rtt = 0.040220 ms, bw = --.--- Kbps
    Hop queueing:      avg = -0.000850 ms (0 bytes)
 3: 194.72.3.66 (core2-gig10-1.kingston.ukcore.bt.net)
                       ???                          -          process hangs at this point!
```

This example shows a "pchar" test across a path where icmp responses are ***not*** allowed.

Tools in Detail . . . . .

Pchar - ./pchar 192.168.1.8 (a local address)

```
pchar to 192.168.1.8 (192.168.1.8) using UDP/IPv4
Using raw socket input
Packet size increments from 32 to 1500 by 32
46 test(s) per repetition : 32 repetition(s) per hop

 0: 192.168.1.231 (dhcp-192-168-1-231.uk.willdata.com)

    Partial loss:      0 / 1472 (0%)
    Partial char:      rtt = 10.792415 ms, (b = 0.003369 ms/B), r2 = 0.157013
                       stddev rtt = 0.950840, stddev b = 0.001177
    Partial queueing:  avg = 0.015037 ms (4463 bytes)
    Hop char:          rtt = 10.792415 ms, bw = 2374.706954 Kbps
    Hop queueing:      avg = 0.015037 ms (4463 bytes)
 1: 192.168.1.8 (zplex.uk.willdata.com)

    Path length:       1 hops
    Path char:         rtt = 10.792415 ms r2 = 0.157013
    Path bottleneck:   2374.706954 Kbps
    Path pipe:         3203 bytes
    Path queueing:     average = 0.0150
    Start time:        Thu Feb  1 09:07
    End time:          Thu Feb  1 09:14
```

```
Partial loss      = number of pkts / percentage pkts lost
Partial  char     = RTT, delay Byte, min delay pkt
Partial queueing  = ave. queue of data incl. of this hop
Hop char          = RTT and b/width for the current hop
Hop queueing      = average queue of data this hop

Path bottleneck   = "bottleneck" (achieved) bandwidth
Path pipe         = Bandwidth-Delay Product = traffic
                    "on the wire" (cf RWIN buffer)
```

Tools in Detail . . . . .

Pchar

Remember:

ICMP may be restricted over the test path

Not all platforms have the same controls or defaults

Think of the impact on the network of using these kind of tools!!

The figures produced are estimates (ref. pchar "man pages" of pchar and, as already mentioned for some previous tools, the results will probably not reflect the exact behaviour of the applications using the same path.

Learn more at:

http://www.kitchenlab.org/www/bmah/Software/pchar/

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

Netcat

Netcat - a read/write utility for networks (TCP or UDP).
It can be used on its own or be driven by user code.
It is also a very powerful network debugging and exploration tool, which can create almost any kind of connection:-

- Outbound or inbound, TCP or UDP, to or from any ports
- Full DNS forward/reverse checking, with appropriate warnings
- Ability to use any local source port
- Ability to use any locally-configured network source address
- Built-in port-scanning capabilities, with randomizer
- Can read command line arguments from standard input
- Slow-send mode, one line every N seconds
- Hex dump of transmitted and received data
- Ability to let another program service established connections
- Telnet-options responder

Good for testing applications and application paths, but does not "test" or measure the network itself.
Beware of misuse!

Tools in Detail . . . . .

Netcat

```
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [options] [hostname] [port]
options:
        -d              detach from console, background mode

        -e prog         inbound program to exec [dangerous!!]
        -g gateway      source-routing hop point[s], up to 8
        -G num          source-routing pointer: 4, 8, 12, ...
        -h              this help
        -i secs         delay interval for lines sent, ports scanned
        -l              listen mode, for inbound connects
        -L              listen harder, re-listen on socket close
        -n              numeric-only IP addresses, no DNS
        -o file         hex dump of traffic
        -p port         local port number
        -r              randomize local and remote ports
        -s addr         local source address
        -t              answer TELNET negotiation
        -u              UDP mode
        -v              verbose [use twice to be more verbose]
        -w secs         timeout for
        -z              zero-I/O mod
port numbers can be individual or ra
```

Learn more at:

http://netcat.sourceforge.net/

http://nmap.org/ncat/

Tools in Detail . . . . .

Netcat - Retrieve page from web server

```
C:\>nc -v www.google.co.uk 80
www.l.google.com [216.239.59.103] 80 (http) open
GET / HTTP/1.0

HTTP/1.0 302 Found
Location: http://www.google.co.uk/
Cache-Control: private
Set-Cookie:
PREF=ID=bebf53d3e8c044c6:TM=1170500572:LM=1170500572:S=DBxO29wrWXh5ex5E;
expires=Sun, 17-Jan-2038 19:14:07 G
MT; path=/; domain=.google.com
Content-Type: text/html
Server: GWS/2.1
Content-Length: 221
Date: Sat, 03 Feb 2007 11:02:52 GMT
Connection: Keep-Alive

<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>302 Moved</TITLE></HEAD><BODY>
<H1>302 Moved</H1>
The document has moved
<A HREF="http://www.google.co.uk/">here</A>.
</BODY></HTML>
```

Tools in Detail . . . . .

Netcat - "NC" to "NC" connection

192.168.27.10

```
c:\>nc -l -p 23 -t -e cmd.exe
```

192.168.27.10

```
C:\Documents and Settings\gdw>netstat -a

Active Connections
  Proto  Local Addr
  TCP    wds-gdw:ft
  TCP    wds-gdw:te
  TCP    wds-gdw:ep
  TCP    wds-gdw:mi
  TCP    wds-gdw:10
  TCP    wds-gdw:53
  TCP    wds-gdw:10
. . . .
```

192.168.27.50

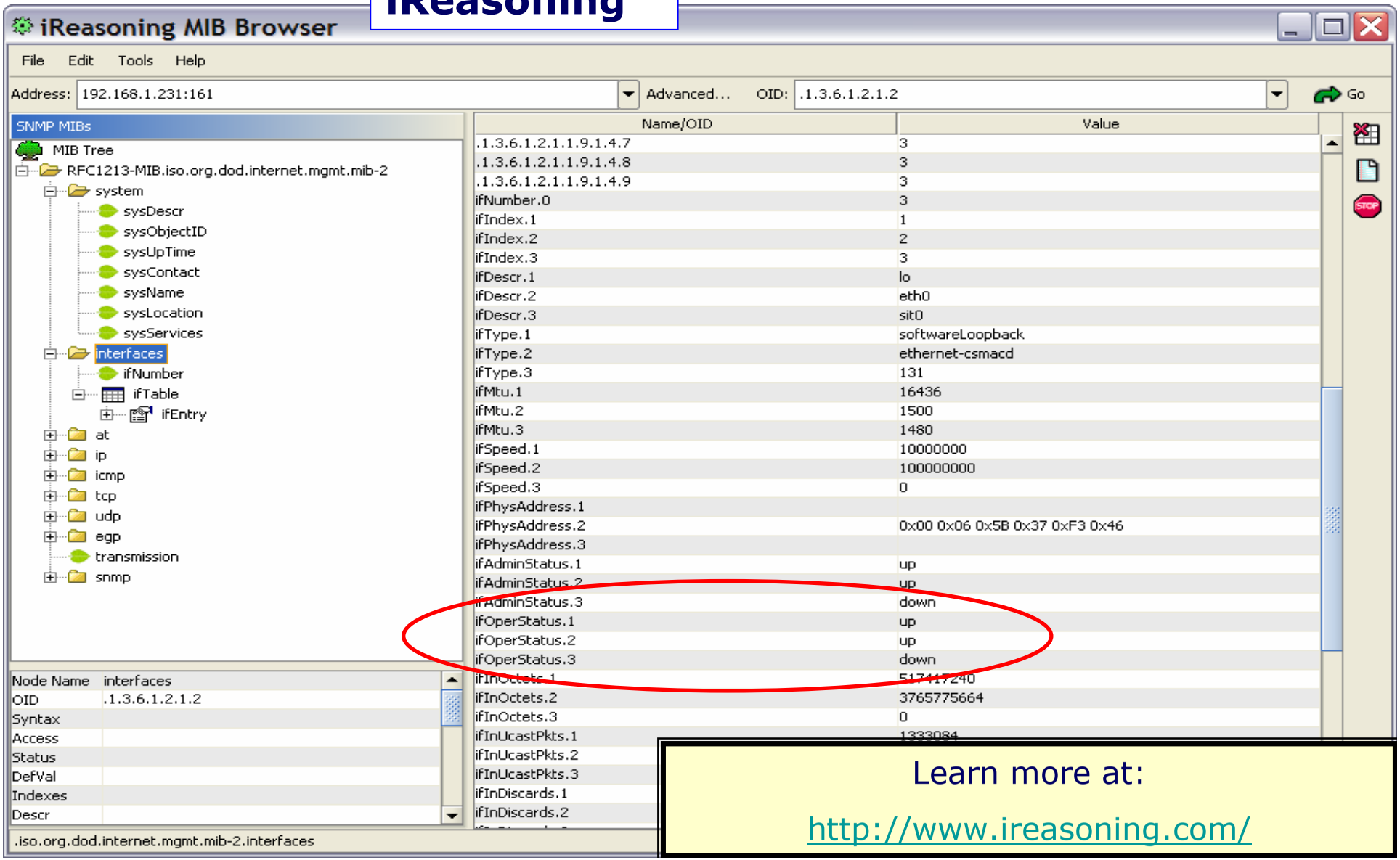
```
C:\>nc 192.168.27.10 23
Microsoft Windows XP [Version 5.1.2600] . . .

C:\>ipconfig
ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.27.10
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.27.1

C:\>^C
C:\>ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.27.50
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.27.1
```

SNMP - MIBs . . . . . .

iReasoning



Learn more at:

http://www.ireasoning.com/

SNMP - MIBs . . . . . .

IMPLEX

```
SNMP MIB Browser                          ADCDPL   P390 TCPIP     14:48:16

Host Name 192.168.1.231
Community public          MaxRequest 128

   Object                                          Value
_  system
_  interfaces
_   ifNumber                                      3
_   ifTable
_    ifEntry
_     ifIndex
_       .1                                        1
_       .2                                        2
_       .3                                        3
_     ifDescr                                (I)
_     ifType                                 (I)
_     ifMtu                                  (I)
_     ifSpeed                                (I)
_     ifPhysAddress                          (I)
_     ifAdminStatus                          (I)
_     ifOperStatus                           (I)
_     ifLastChange                           (I)
_     ifInOctets                             (I)
_     ifInUcastPkts                          (I)
_     ifInNUcastPkts                         (I)
_     ifInDiscards                           (I)
_     ifInErrors                             (I)
_     ifInUnknownProtos                      (I)
_     ifOutOctets                            (I)
_     ifOutUcastPkts                         (I)
_     ifOutNUcastPkts                        (I)
_     ifOutDiscards                          (I)
_     ifOutErrors                            (I)
_     ifOutQLen                              (I)
_     ifSpecific                             (I)
_  at
_  ip
_  icmp
_  tcp
                                          Objects 265       7671

F1 Help F2 Reset F3 End F4 Prompt F7 Up F8 Down F9 AltView
```

Tools in Detail . . . . .

Packet Analysers – "Sniffers"

- "Original" capture routine - **TCPDUMP**
  + **LIBPCAP** (the Promiscuous Capture Libary) or **WinPcap**.
  Available on most "open" platforms.

- **SSLDUMP** is TCPDUMP with SSL decryption capability.

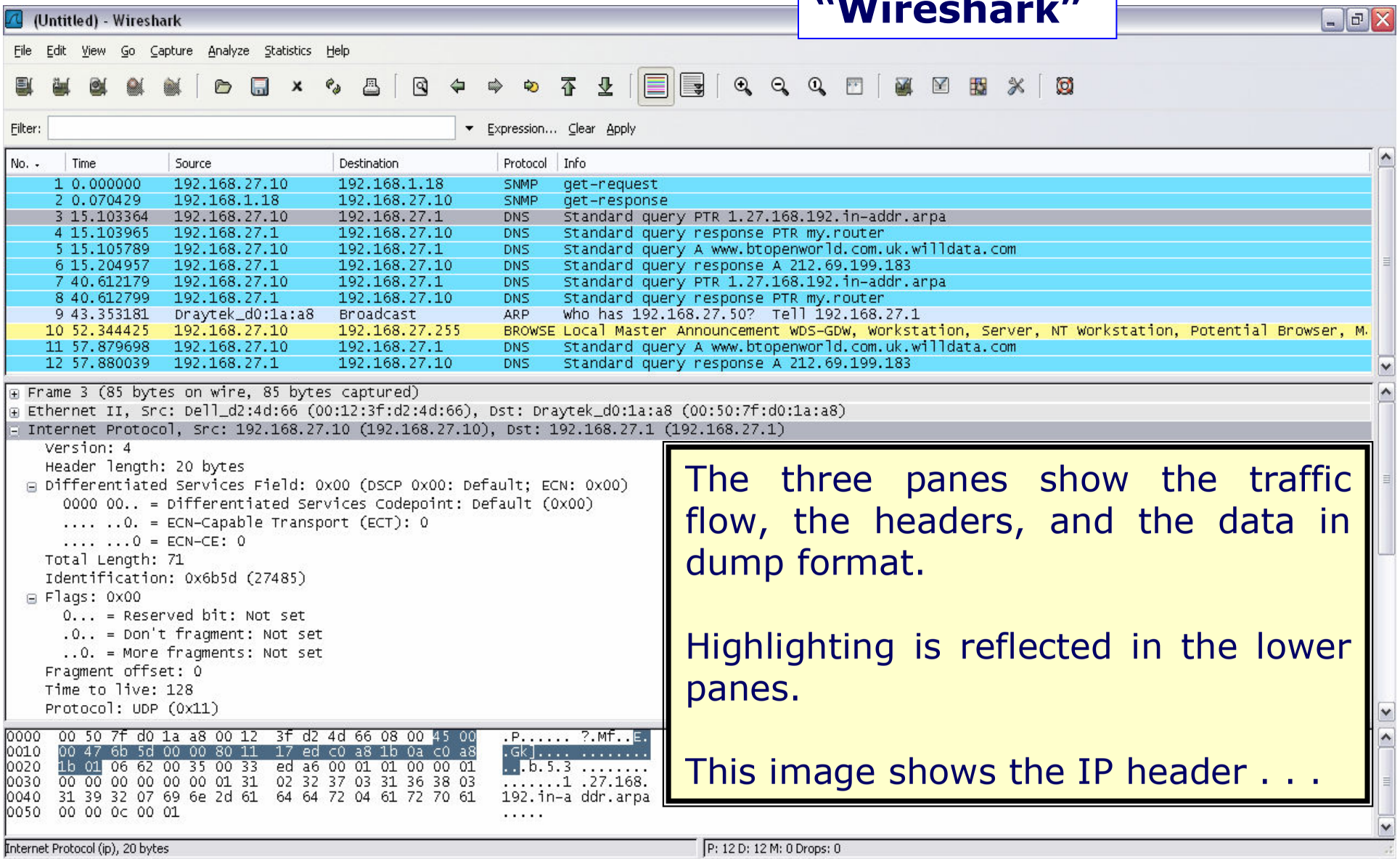
- **ETHEREAL** is a packet analyzer based on TCPDUMP.

- **WIRESHARK** is the latest incarnation of ETHERAL
  Shows actual packets on the network with "breakdown".
  Good for true analysis of the network ***and*** for establishing "common use" baselines.

- **EXIGENCE** provides similar functionality for z/OS.

Tools in Detail . . . . .

"Wireshark"



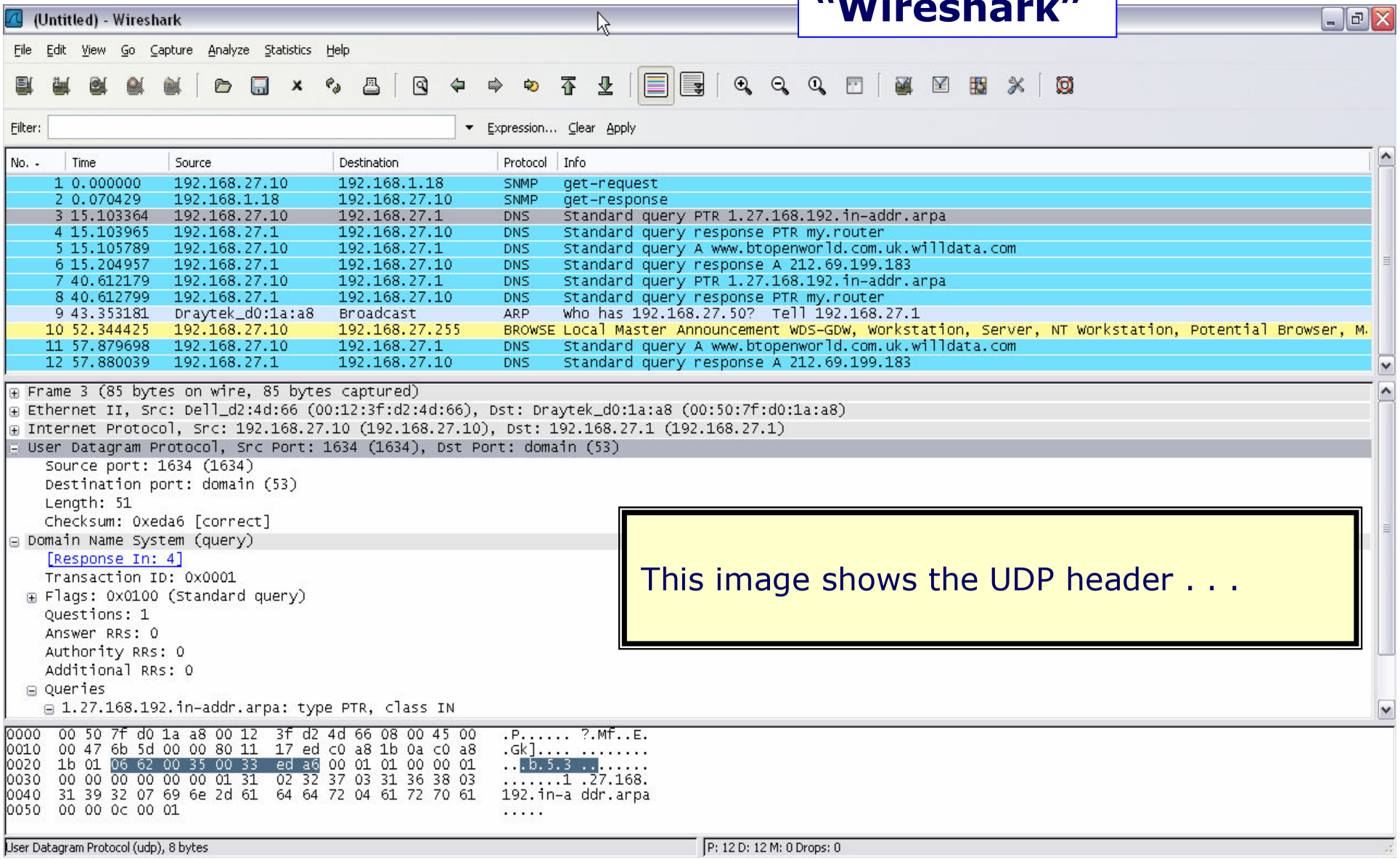
The three panes show the traffic flow, the headers, and the data in dump format.

Highlighting is reflected in the lower panes.

This image shows the IP header . . .

Tools in Detail . . . . .

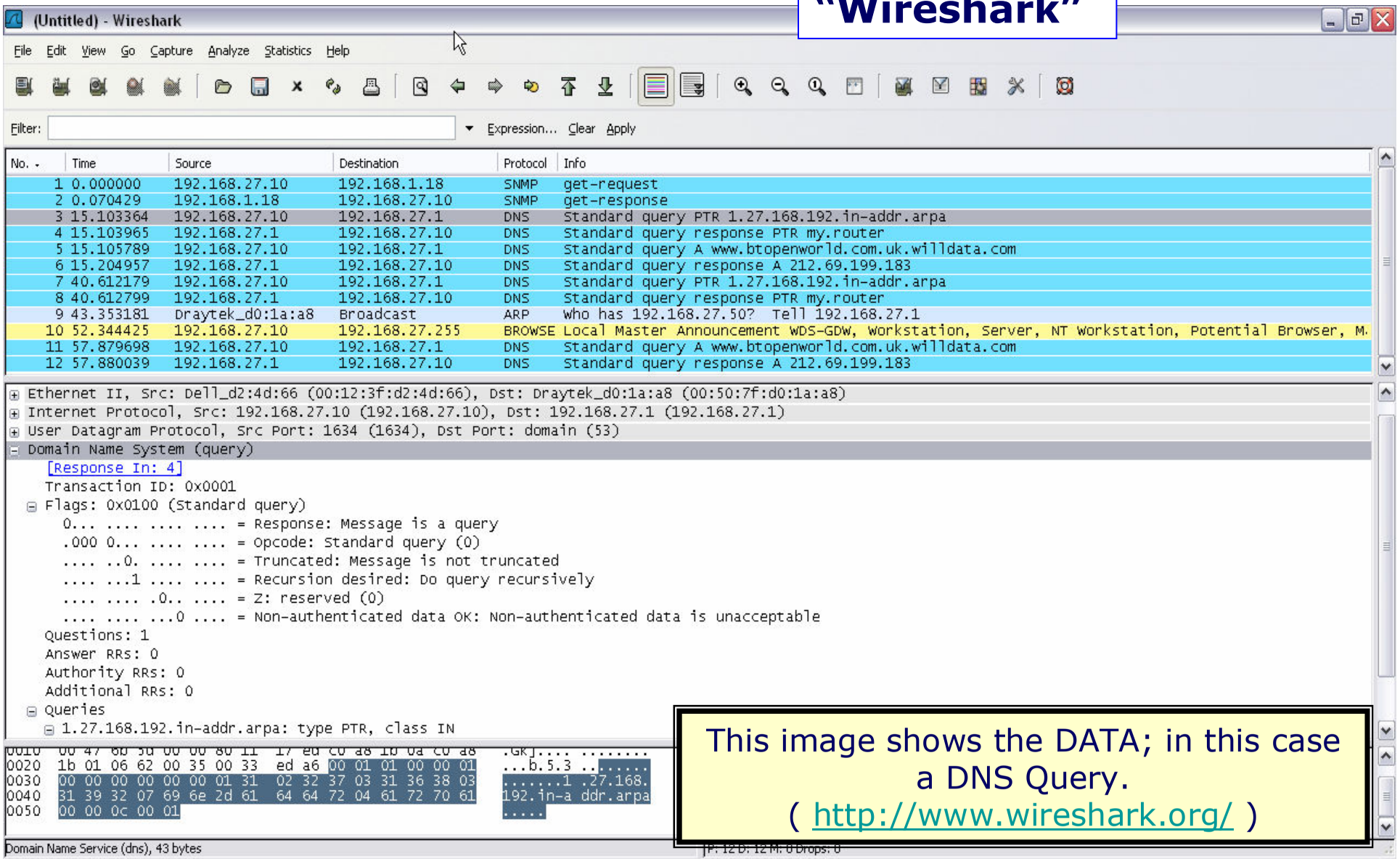
"Wireshark"



This image shows the UDP header . . .

Tools in Detail . . . . .

"Wireshark"

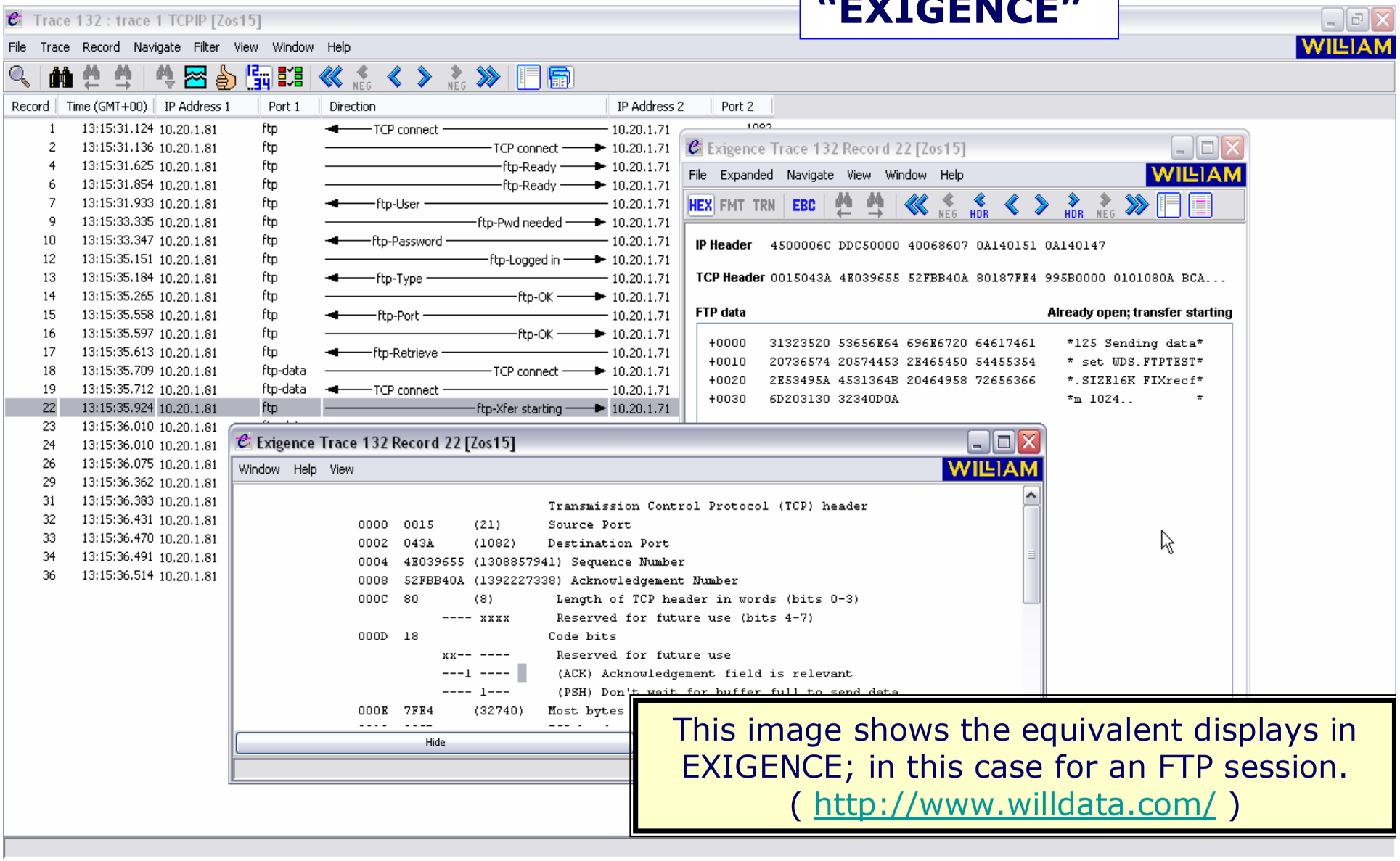


This image shows the DATA; in this case a DNS Query.

(http://www.wireshark.org/)

Tools in Detail . . . . .

"EXIGENCE"



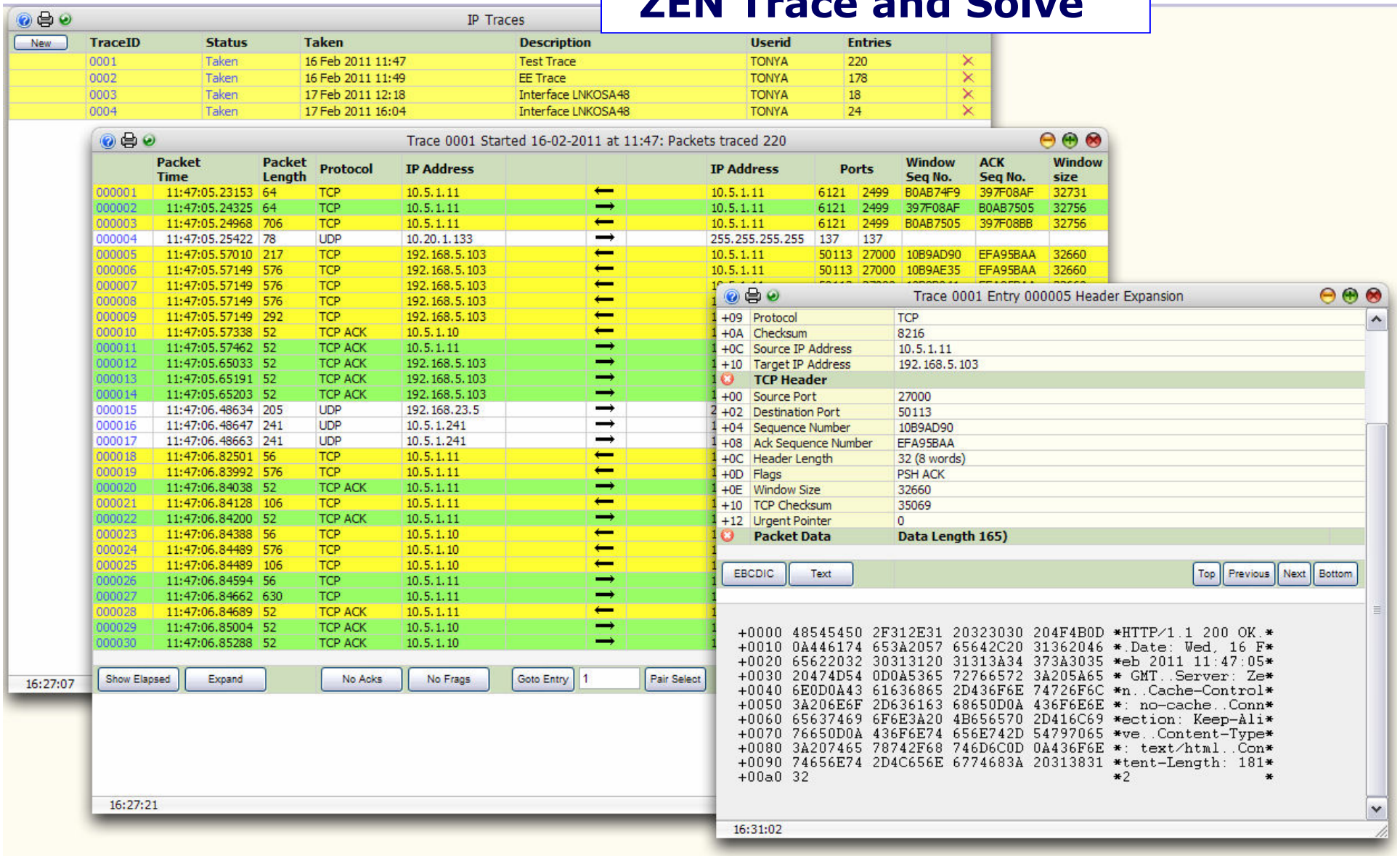
This image shows the equivalent displays in EXIGENCE; in this case for an FTP session.
(http://www.willdata.com/)

WILLIAM DATA SYSTEMS

Tools in Detail . . . . .

SHARE Technology · Connections · Results

"ZEN Trace and Solve"



IP Traces

New	TraceID	Status	Taken	Description	Userid	Entries	
	0001	Taken	16 Feb 2011 11:47	Test Trace	TONYA	220	✕
	0002	Taken	16 Feb 2011 11:49	EE Trace	TONYA	178	✕
	0003	Taken	17 Feb 2011 12:18	Interface LNKOSA48	TONYA	18	✕
	0004	Taken	17 Feb 2011 16:04	Interface LNKOSA48	TONYA	24	✕

16:27:07

Trace 0001 Started 16-02-2011 at 11:47: Packets traced 220

	Packet Time	Packet Length	Protocol	IP Address			IP Address	Ports		Window Seq No.	ACK Seq No.	Window size
000001	11:47:05.23153	64	TCP	10.5.1.11			10.5.1.11	6121	2499	B0AB74F9	397F08AF	32731
000002	11:47:05.24325	64	TCP	10.5.1.11			10.5.1.11	6121	2499	397F08AF	B0AB7505	32756
000003	11:47:05.24968	706	TCP	10.5.1.11			10.5.1.11	6121	2499	B0AB7505	397F08BB	32756
000004	11:47:05.25422	78	UDP	10.20.1.133			255.255.255.255	137	137			
000005	11:47:05.57010	217	TCP	192.168.5.103			10.5.1.11	50113	27000	10B9AD90	EFA95BAA	32660
000006	11:47:05.57149	576	TCP	192.168.5.103			10.5.1.11	50113	27000	10B9AE35	EFA95BAA	32660
000007	11:47:05.57149	576	TCP	192.168.5.103								
000008	11:47:05.57149	576	TCP	192.168.5.103								
000009	11:47:05.57149	292	TCP	192.168.5.103								
000010	11:47:05.57338	52	TCP ACK	10.5.1.10								
000011	11:47:05.57462	52	TCP ACK	10.5.1.11								
000012	11:47:05.65033	52	TCP ACK	192.168.5.103								
000013	11:47:05.65191	52	TCP ACK	192.168.5.103								
000014	11:47:05.65203	52	TCP ACK	192.168.5.103								
000015	11:47:06.48634	205	UDP	192.168.23.5								
000016	11:47:06.48647	241	UDP	10.5.1.241								
000017	11:47:06.48663	241	UDP	10.5.1.241								
000018	11:47:06.82501	56	TCP	10.5.1.11								
000019	11:47:06.83992	576	TCP	10.5.1.11								
000020	11:47:06.84038	52	TCP ACK	10.5.1.11								
000021	11:47:06.84128	106	TCP	10.5.1.11								
000022	11:47:06.84200	52	TCP ACK	10.5.1.11								
000023	11:47:06.84388	56	TCP	10.5.1.10								
000024	11:47:06.84489	576	TCP	10.5.1.10								
000025	11:47:06.84489	106	TCP	10.5.1.10								
000026	11:47:06.84594	56	TCP	10.5.1.11								
000027	11:47:06.84662	630	TCP	10.5.1.11								
000028	11:47:06.84689	52	TCP ACK	10.5.1.11								
000029	11:47:06.85004	52	TCP ACK	10.5.1.10								
000030	11:47:06.85288	52	TCP ACK	10.5.1.10								

Show Elapsed | Expand | No Acks | No Frags | Goto Entry 1 | Pair Select

16:27:21

Trace 0001 Entry 000005 Header Expansion

+09	Protocol	TCP
+0A	Checksum	8216
+0C	Source IP Address	10.5.1.11
+10	Target IP Address	192.168.5.103
	TCP Header	
+00	Source Port	27000
+02	Destination Port	50113
+04	Sequence Number	10B9AD90
+08	Ack Sequence Number	EFA95BAA
+0C	Header Length	32 (8 words)
+0D	Flags	PSH ACK
+0E	Window Size	32660
+10	TCP Checksum	35069
+12	Urgent Pointer	0
	Packet Data	**Data Length 165)**

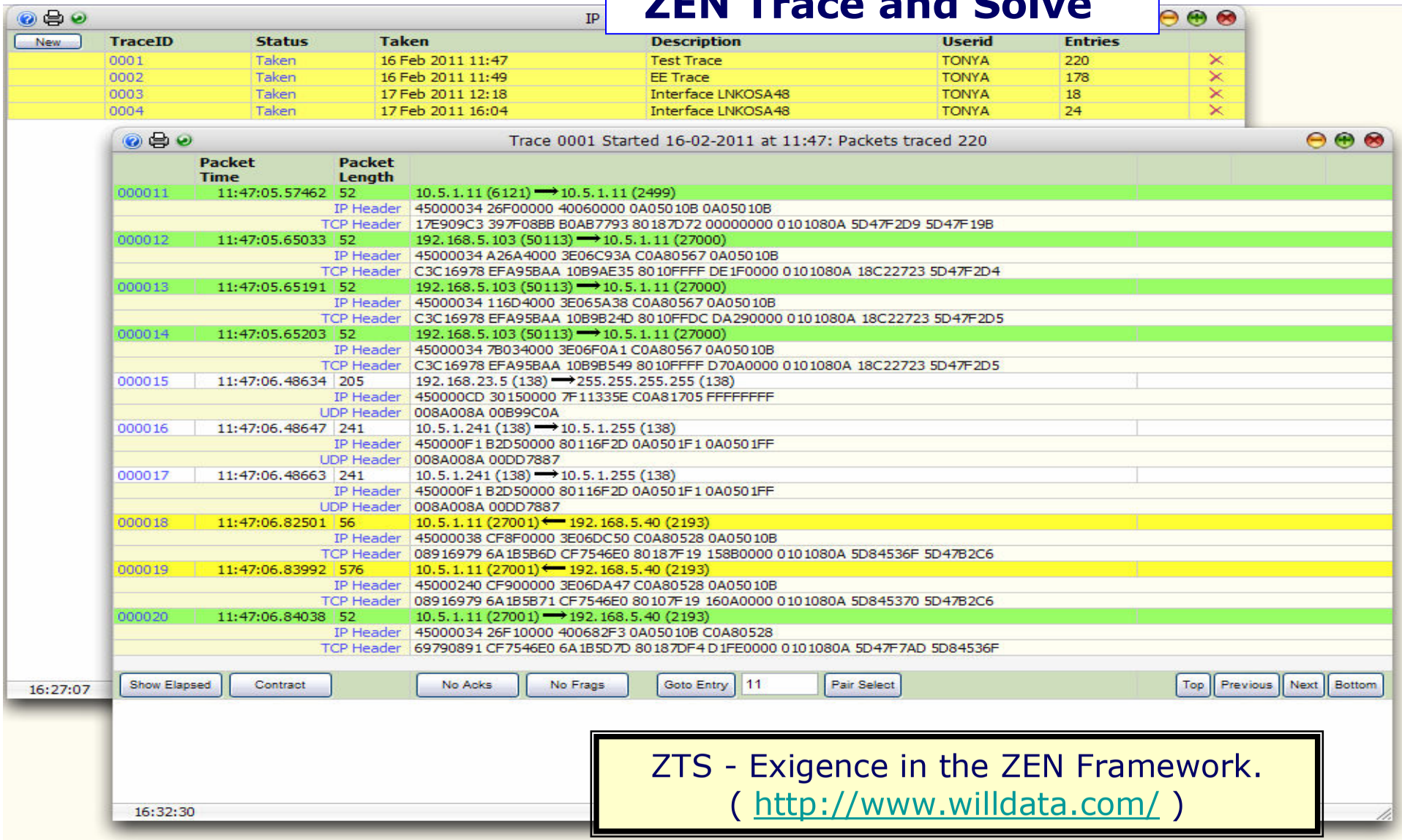
EBCDIC | Text | Top | Previous | Next | Bottom

```
+0000 48545450 2F312E31 20323030 204F4B0D *HTTP/1.1 200 OK.*
+0010 0A446174 653A2057 65642C20 31362046 *.Date: Wed, 16 F*
+0020 65622032 30313120 31313A34 373A3035 *eb 2011 11:47:05*
+0030 20474D54 0D0A5365 72766572 3A205A65 * GMT..Server: Ze*
+0040 6E0D0A43 61636865 2D436F6E 74726F6C *n..Cache-Control*
+0050 3A206E6F 2D636163 68650D0A 436F6E6E *: no-cache..Conn*
+0060 65637469 6F6E3A20 4B656570 2D416C69 *ection: Keep-Ali*
+0070 76650D0A 436F6E74 656E742D 54797065 *ve..Content-Type*
+0080 3A207465 78742F68 746D6C0D 0A436F6E *: text/html..Con*
+0090 74656E74 2D4C656E 6774683A 20313831 *tent-Length: 181*
+00a0 32                                  *2              *
```

16:31:02

Tools in Detail . . . . .

"ZEN Trace and Solve"



IP

New	TraceID	Status	Taken	Description	Userid	Entries
	0001	Taken	16 Feb 2011 11:47	Test Trace	TONYA	220
	0002	Taken	16 Feb 2011 11:49	EE Trace	TONYA	178
	0003	Taken	17 Feb 2011 12:18	Interface LNKOSA48	TONYA	18
	0004	Taken	17 Feb 2011 16:04	Interface LNKOSA48	TONYA	24

Trace 0001 Started 16-02-2011 at 11:47: Packets traced 220

	Packet Time	Packet Length	
000011	11:47:05.57462	52	10.5.1.11 (6121) ⟶ 10.5.1.11 (2499)
		IP Header	45000034 26F00000 40060000 0A05010B 0A05010B
		TCP Header	17E909C3 397F08BB B0AB7793 80187D72 00000000 0101080A 5D47F2D9 5D47F19B
000012	11:47:05.65033	52	192.168.5.103 (50113) ⟶ 10.5.1.11 (27000)
		IP Header	45000034 A26A4000 3E06C93A C0A80567 0A05010B
		TCP Header	C3C16978 EFA95BAA 10B9AE35 8010FFFF DE1F0000 0101080A 18C22723 5D47F2D4
000013	11:47:05.65191	52	192.168.5.103 (50113) ⟶ 10.5.1.11 (27000)
		IP Header	45000034 116D4000 3E065A38 C0A80567 0A05010B
		TCP Header	C3C16978 EFA95BAA 10B9B24D 8010FFDC DA290000 0101080A 18C22723 5D47F2D5
000014	11:47:05.65203	52	192.168.5.103 (50113) ⟶ 10.5.1.11 (27000)
		IP Header	45000034 7B034000 3E06F0A1 C0A80567 0A05010B
		TCP Header	C3C16978 EFA95BAA 10B9B549 8010FFFF D70A0000 0101080A 18C22723 5D47F2D5
000015	11:47:06.48634	205	192.168.23.5 (138) ⟶ 255.255.255.255 (138)
		IP Header	450000CD 30150000 7F11335E C0A81705 FFFFFFFF
		UDP Header	008A008A 00B99C0A
000016	11:47:06.48647	241	10.5.1.241 (138) ⟶ 10.5.1.255 (138)
		IP Header	450000F1 B2D50000 80116F2D 0A0501F1 0A0501FF
		UDP Header	008A008A 00DD7887
000017	11:47:06.48663	241	10.5.1.241 (138) ⟶ 10.5.1.255 (138)
		IP Header	450000F1 B2D50000 80116F2D 0A0501F1 0A0501FF
		UDP Header	008A008A 00DD7887
000018	11:47:06.82501	56	10.5.1.11 (27001) ⟵ 192.168.5.40 (2193)
		IP Header	45000038 CF8F0000 3E06DC50 C0A80528 0A05010B
		TCP Header	08916979 6A1B5B6D CF7546E0 80187F19 158B0000 0101080A 5D84536F 5D47B2C6
000019	11:47:06.83992	576	10.5.1.11 (27001) ⟵ 192.168.5.40 (2193)
		IP Header	45000240 CF900000 3E06DA47 C0A80528 0A05010B
		TCP Header	08916979 6A1B5B71 CF7546E0 80107F19 160A0000 0101080A 5D845370 5D47B2C6
000020	11:47:06.84038	52	10.5.1.11 (27001) ⟶ 192.168.5.40 (2193)
		IP Header	45000034 26F10000 400682F3 0A05010B C0A80528
		TCP Header	69790891 CF7546E0 6A1B5D7D 80187DF4 D1FE0000 0101080A 5D47F7AD 5D84536F

16:27:07

Show Elapsed | Contract | No Acks | No Frags | Goto Entry 11 | Pair Select | Top | Previous | Next | Bottom

16:32:30

ZTS - Exigence in the ZEN Framework.
(http://www.willdata.com/)

WILLIAM DATA SYSTEMS

And, In Passing . . . . .

SHARE Technology · Connections · Results

Network & Security testers

"Nessus" **-** ("**The Tenable Newt**") a security vulnerablility scanner.
(www.nessus.org)

"Nmap" - a network and security scanner
(insecure.org & nmap.org)

Use responsibly – Use with care !

SHARE in Orlando 2011

Tools in Detail . . . . .

Nmap (edited)

```
>nmap -v -A 192.168.27.50
Starting Nmap 4.20 ( http://insecure.org ) at 2007-02-03 11:40 GMT Standard Time
Initiating ARP Ping Scan at 11:40
Scanning 192.168.27.50 [1 port]
Completed ARP Ping Scan at 11:40, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 11:40
Completed Parallel DNS resolution of 1 host. at 11:40, 0.03s elapsed
Initiating SYN Stealth Scan at 11:40 : Scanning 192.168.27.50 [1697 ports]
Discovered open port 135/tcp on 192.168.27.50
Completed SYN Stealth Scan at 11:40, 39.05s elapsed (1697 total ports)
Initiating Service scan at 11:40 : Scanning 1 service on 192.168.27.50
Completed Service scan at 11:41, 11.63s elapsed (1 service on 1 host)
Warning:  OS detection for 192.168.27.50 will be MUCH less reliable because we did not
find at least 1 open and 1 closed TCP port
. . .
Host 192.168.27.50 appears to be up ... good.
Interesting ports on 192.168.27.50:
Not shown: 1696 filtered ports
PORT    STATE SERVICE VERSION
135/tcp open  msrpc   Microsoft Windows RPC
MAC Address: xx:xx:xx:xx:xx:xx (Dell ESG Pcba Test)
Running (JUST GUESSING) : Microsoft Windows 2000|XP (98%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop : TCP Sequence Prediction: Difficulty=0 (Trivial joke)
. . .
OS and Service detection performed. Nmap finished: 1 IP address (1 host up) scanned in
67.000 seconds
                Raw packets sent: 3517 (162.066KB) | Rcvd: 86 (4770B)
```

(NB. This sample has been edited to fit !)

WILLIAM DATA SYSTEMS

Problem Diagnosis . . .

SHARE Technology · Connections · Results

Outline Steps:

- Check the stack – "**ping**" local loopback
- "**ping**" the remote host/server name
- "**ping**" with IPaddress – the DNS may be down
- If "ping" fails "**traceroute**" - find where it stops
- Use "**netstat**" to check the interface
- Check routing (is it as expected?)
- If ping works, try "**telnet**" (standard port 23)
- If "**telnet**" works try **telnet to the application port**
- If that works try the application
- Use "**netstat**" to check the connection exists
- Check your syslogs (remember USS ! "syslogd" !)
- Do you ***still*** have a failure? … **trace it!**

WILLIAM DATA SYSTEMS

Summary . . . . .

SHARE Technology · Connections · Results

- **Know Your Network !**
- **Keep Up-to-Date Documentations & Diagrams !**
- **Know the Tools** **(most tools can be used for practice at any time)**
- **Plan Your Approach to Any Problem**
- **Stop , Look , and LISTEN !!**

WILLIAM DATA SYSTEMS

SHARE
Technology · Connections · Results

Thank you !

SHARE in Orlando 2011