Network Measurement and Monitoring Tools

Lecture 3-4

Network Monitoring and Optimisation tools

Introduction: Why do we measure?

- Network measurements are essential to keep a network running efficiently
- For detecting bottlenecks/system faults and for forecasting future network requirements.
- Dealing with performance problems or faults (hardware and software) is relatively easy providing the network's usual performance is well documented.

How to monitor and measure

- You can monitor server and network performance with a variety of OS utility programmes or
- Use specialized test instruments such as field testers for diagnosing cabling problems and sophisticated monitoring and management tools for network performance analysis.

Network Measurement Tools

- Cable testers
- OS diagnostic tools
- TCP/IP utilities
- RMON (Remote Monitoring)
- SNMP (Simple Network Management Protocol)
- Network Analysers
- Wireless Monitoring and Measurement
- Simulators/Emulators

Cable Testing Tools

- Multimeters
- Basic cable testers
- Verification cable testers
- Certification cable testers

Cable Tester

MicroScanner™ Pro

OMNIScanner®2

Verification Testing

- Visual Inspection (for damage, bend radii)
- Continuity to remote end
- Shorts between conductors
- Crossed, reversed, split pairs etc.
- Other mis-wiring

Verification Testing

Correct

Reversed

Crossed

Split

Certification Testing CAT 5e

- Wire map (inc. continuity)
- Length
- Insertion Loss (attenuation)
- NEXT loss (pair to pair) (Near End Croxx Talk)
- Propagation delay
- Delay skew
- Needs advanced field tester like Fluke Omniscanner

Time Domain Reflectometer

- Time Domain Reflectometer (TDR) is used to measure length of cables - Copper or Fibre
- Must be calibrated for NVP
- *NVP = nominal velocity of propagation for the cable type quoted as fraction of speed of light*

  e.g. NVP for CAT5 UTP is approx. 2/3 x10^8 m/s

- Length = NVP x T_P
- T_P ***= propagation time***

e.g. 100m max for 100/1000BaseT channel or 90m max between fixed sockets to allow 10m for patch cables

TDR Example

- Determine the correct NVP setting as a % age of the speed of light (c) for a TDR cable tester if a calibration test on a 50m length of cable gave a round trip delay reading of 0.5 µs

- Length = NVP x T_P
- NVP = Length / T_P
  = $50/0.25 \times 10^{-6} = 2 \times 10^8$ m/s
- therefore NVP is …… % of c

Windows OS Monitoring Tools

- Event Viewer
- Performance Console
- Task Manager
- Network Monitor

Event Viewer

- The Event Viewer allows you to view the events in the computer log to help locate problems.
- The event logs in Event Viewer allow you to gather information about hardware, software, and system problems.
- You can also monitor Windows security events such as logins.

Event Viewer Logs

Application log

The application log contains events logged by applications or programs. For example, a database program might record a file error in the application log. The program developer decides which events to record.

System log

The system log contains events logged by the Windows system components. For example, the failure of a driver or other system component to load during startup is recorded in the system log. The event types logged by system components are predetermined by Windows.

Security log

The security log can record security events such as valid and invalid logon attempts as well as events related to resource use such as creating, opening, or deleting files. An administrator can specify what events are recorded in the security log. For example, if you have enabled logon auditing, attempts to log on to the system are recorded in the security log.

Performance Console

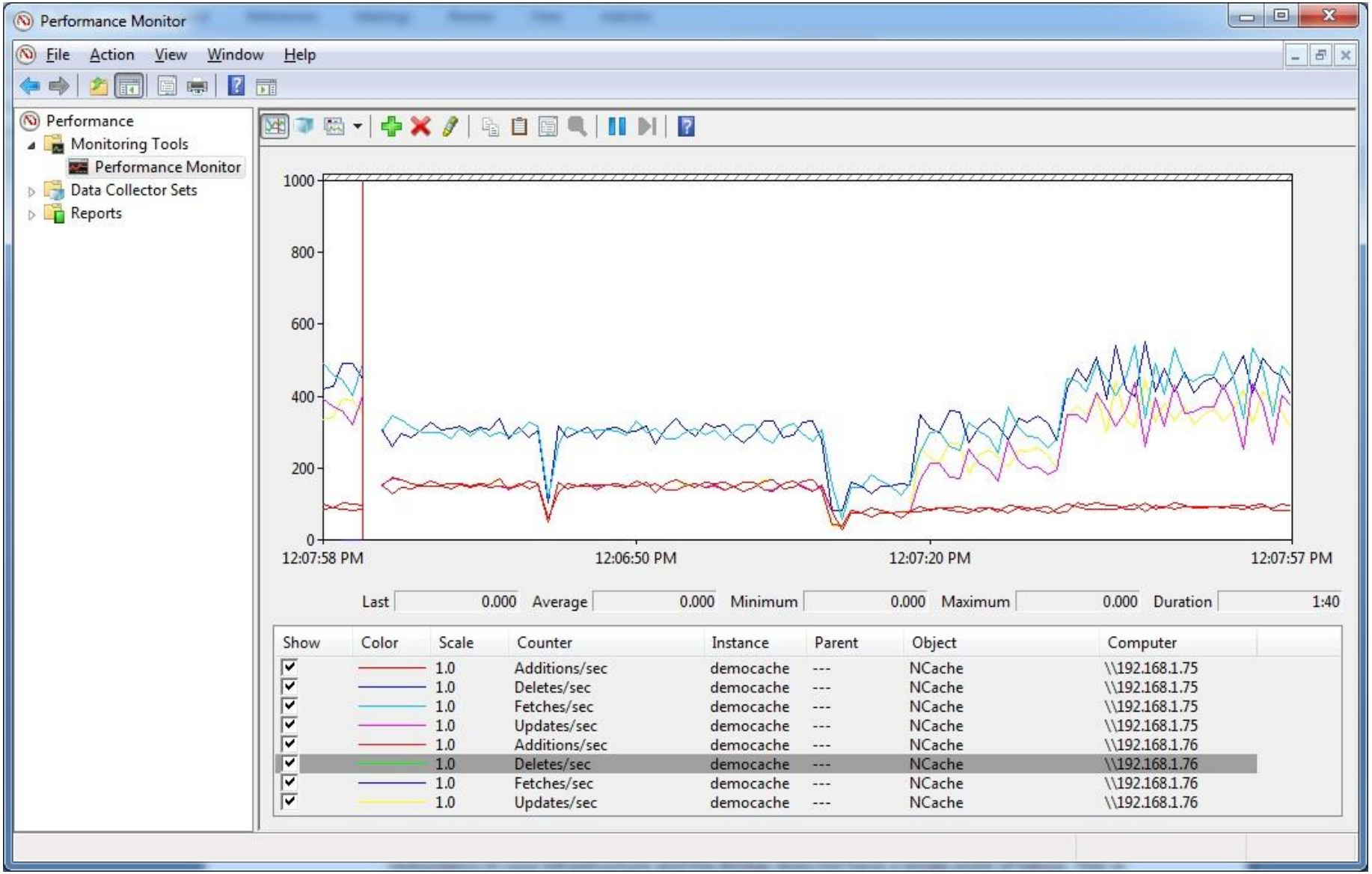
Performance Console :

System Monitor

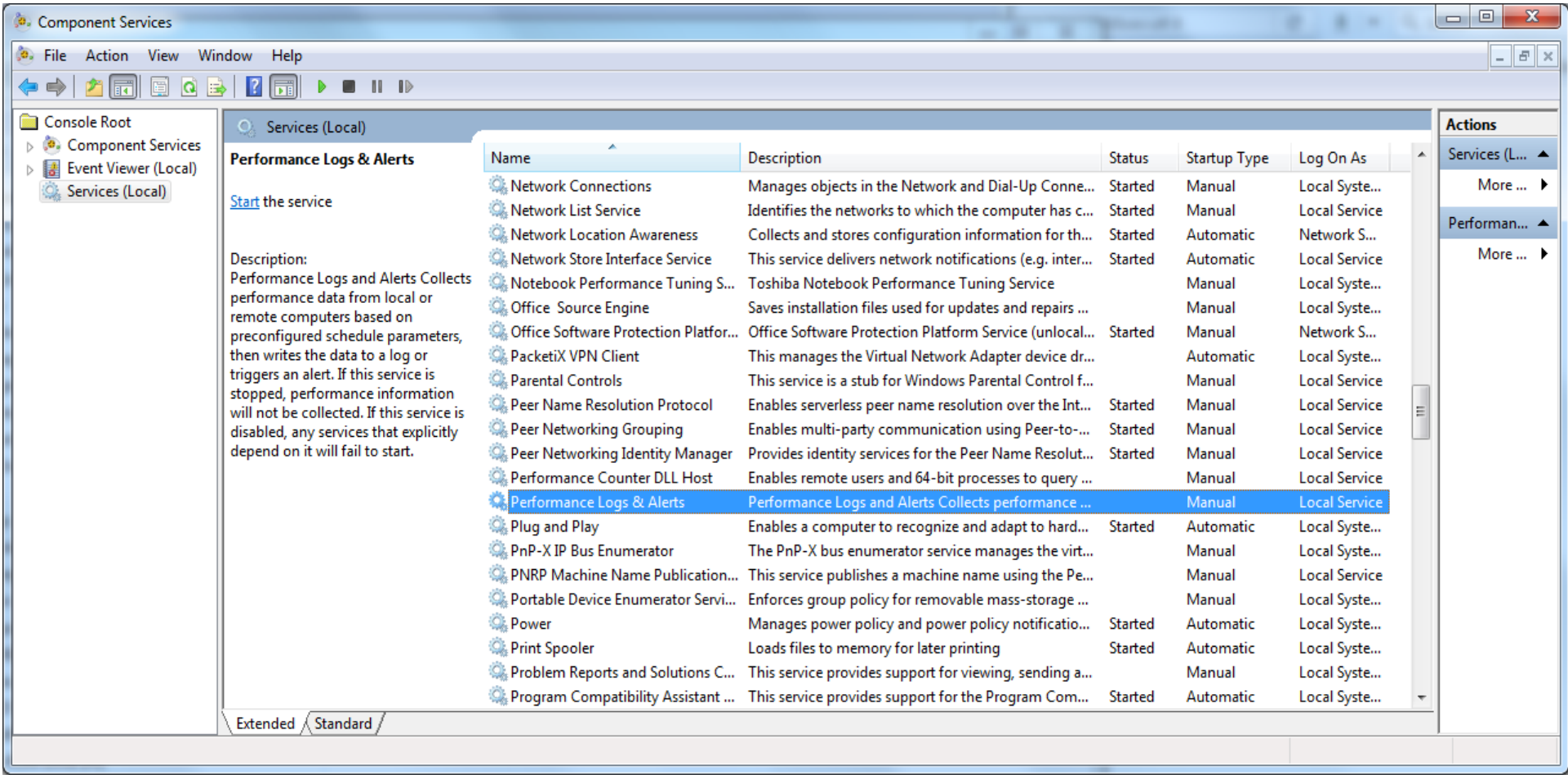
Logs and Alerts

The primary monitoring tools in Windows OS are in the Performance Console. These are System Monitor, and Performance Logs and Alerts. They can be configured to show a wide range of network related events in graphical form to help diagnose performance problems.

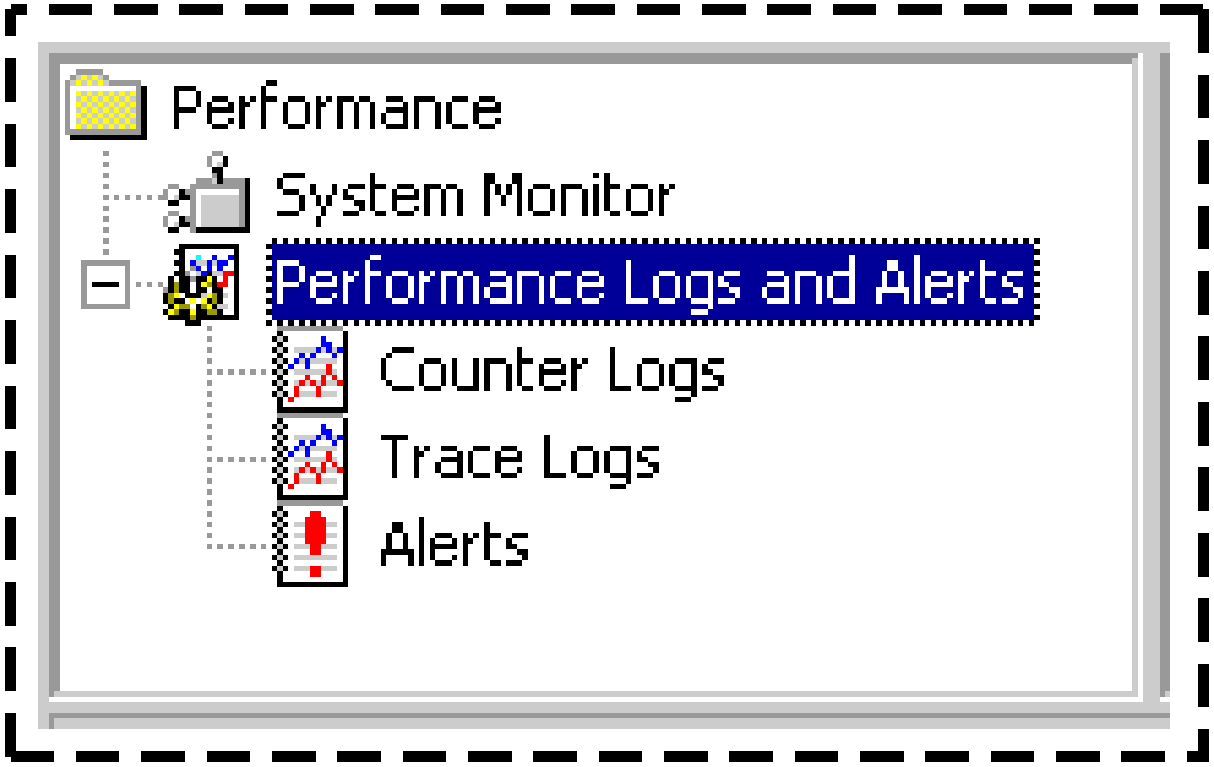
System Monitor



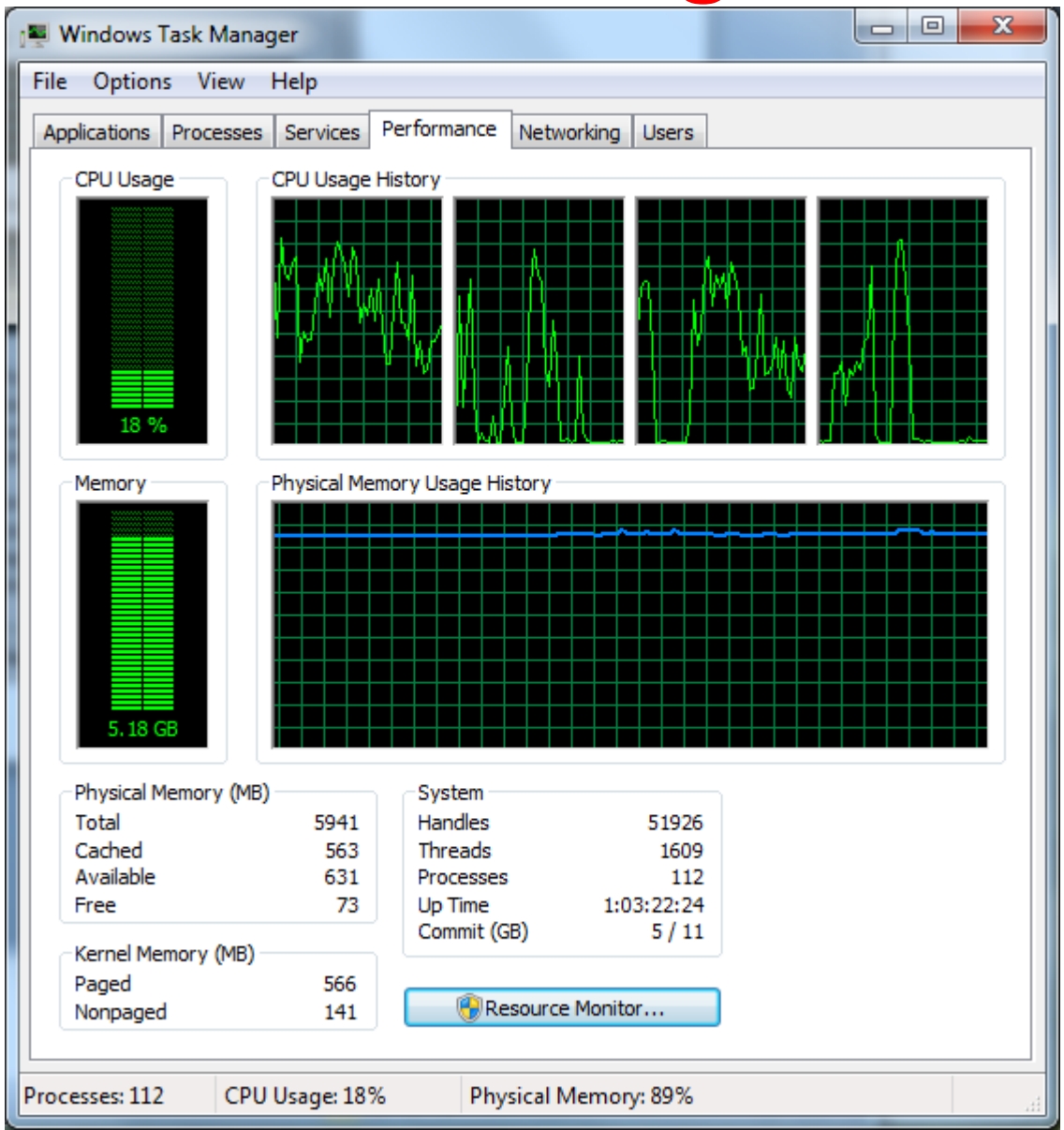
System Monitor



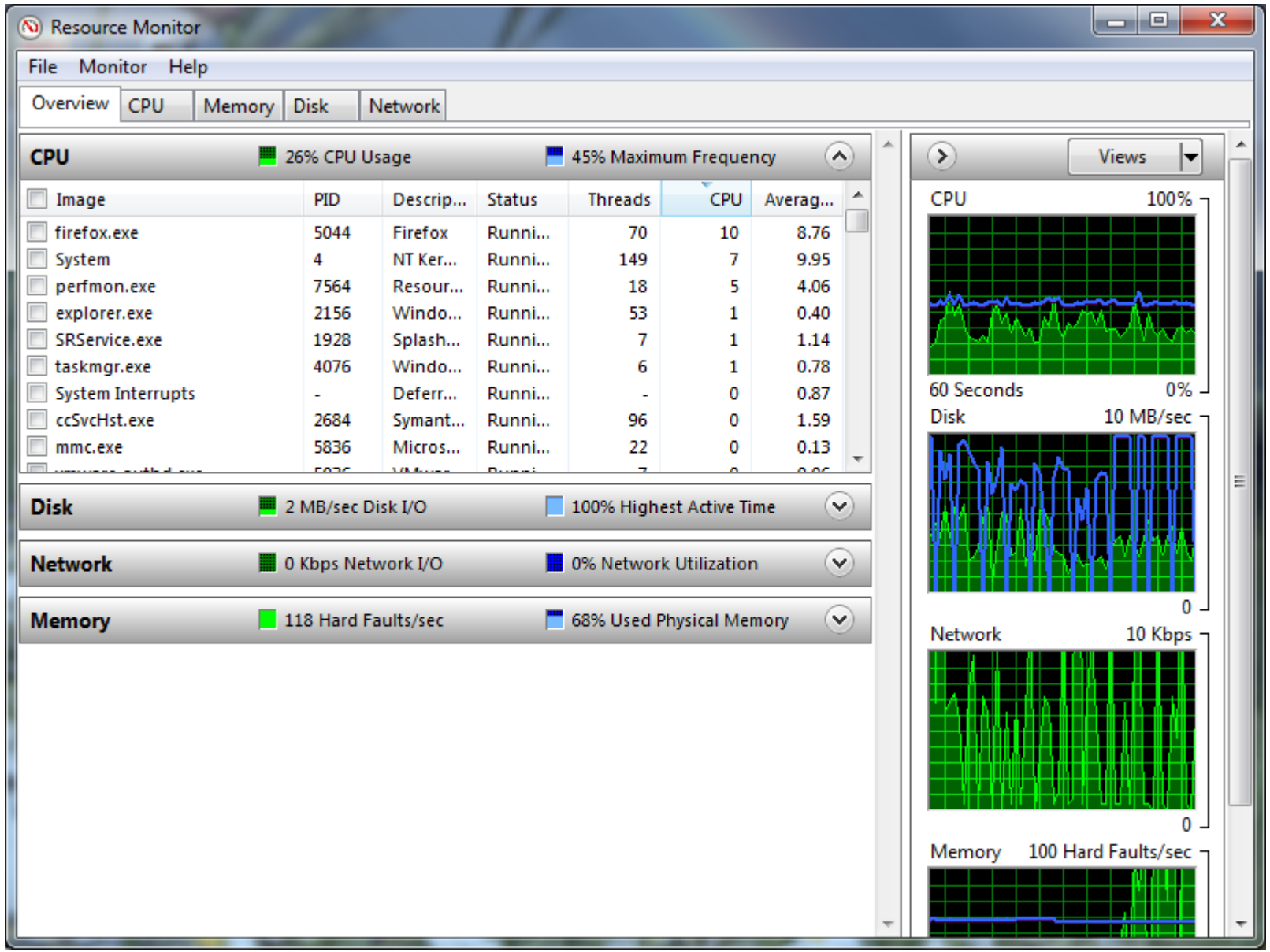
Logs and Alerts



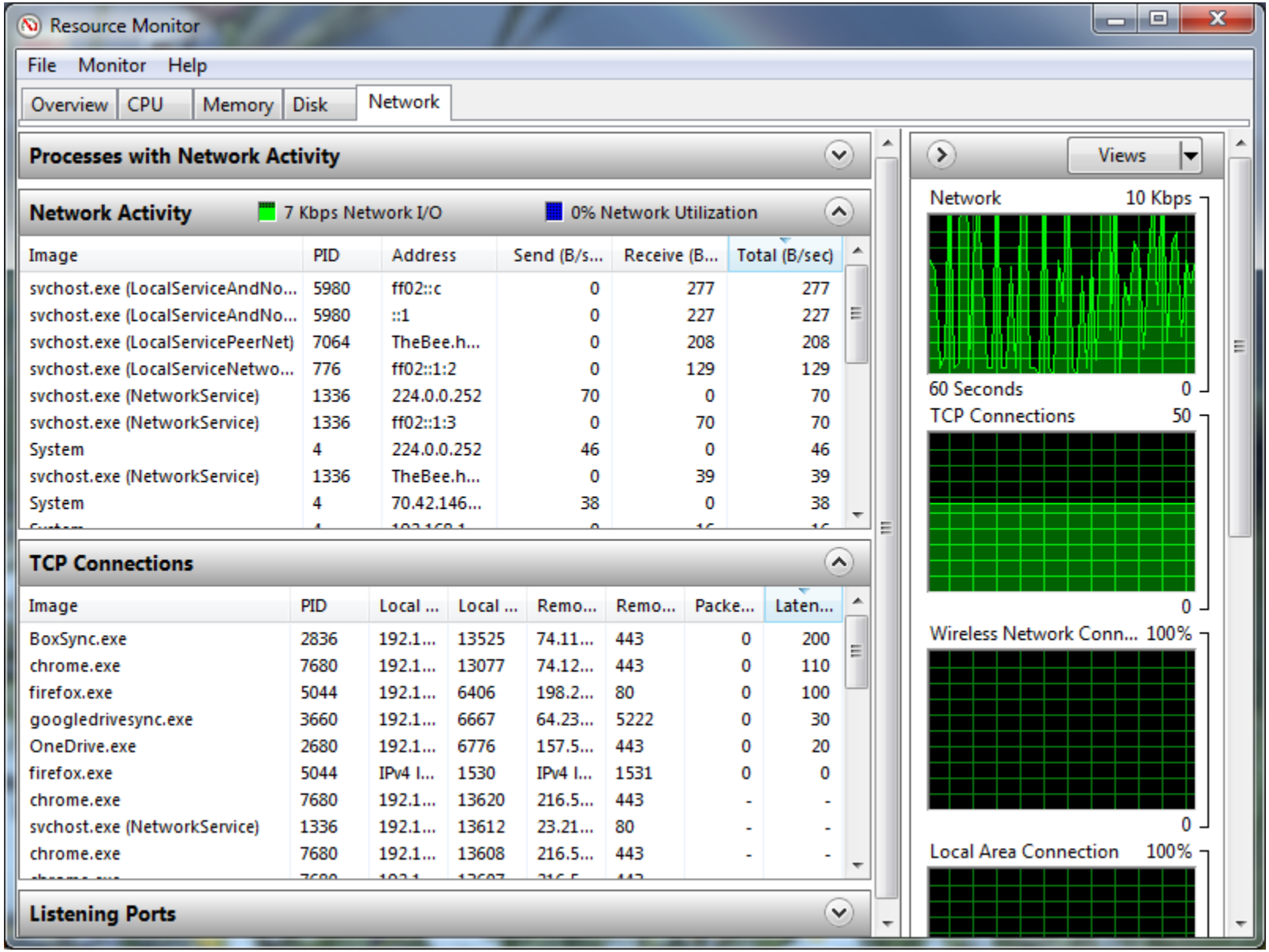
Task Manager



Resource Monitor



Network Monitor



TCP/IP & NetBIOS Utilities

- TCP/IP and NetBIOS diagnostic utilities included with Microsoft Windows 7.
- TCP/IP are useful to identify and resolve TCP/IP networking problems.
- When you troubleshoot a network begin by checking the TCP/IP configuration on the computer that is experiencing the problem.
- Then go on to test the connectivity with other network computers.

A more complete description of using these tools is in the laboratory reference guide: useutils_15.docx in Resources folder of BB site.

arp	View the ARP (Address Resolution Protocol) cache on the interface of the local computer to detect invalid entries.
hostname	Display the host name of the computer.
ipconfig	Display current TCP/IP network configuration values, and update or release Dynamic Host Configuration Protocol (DHCP) allocated leases, and display, register, or flush Domain Name System (DNS) names.
nbtstat	Check the state of current NetBIOS over TCP/IP connections, update the NetBIOS name cache, and determine the registered names and scope ID.
netstat	Display statistics for current TCP/IP connections.
netdiag	Check all aspects of the network connection.
nslookup	Check records, domain host aliases, domain host services, and operating system information by querying Internet domain name servers. Nslookup is discussed in detail in "Windows 2000 DNS" in this book.
pathping	Trace a path to a remote system and report packet losses at each router along the way.
ping	Send ICMP Echo Requests to verify that TCP/IP is configured correctly and that a remote TCP/IP system is available.
route	Display the IP routing table, and add or delete IP routes.
tracert	Trace a path to a remote system.

Tools for Performance Measurement

- There are many open source and third part testing tools available from network test specialists such as **Ixia**
- **Ixia's Ix Chariot** is the market leader for network testing and **Ixia Q-Check** is a very useful free tool – used in labs
- **iPerf** is a very useful testing tool available for Linux and Windows OS – used in labs
- Many websites offer broadband testing tools

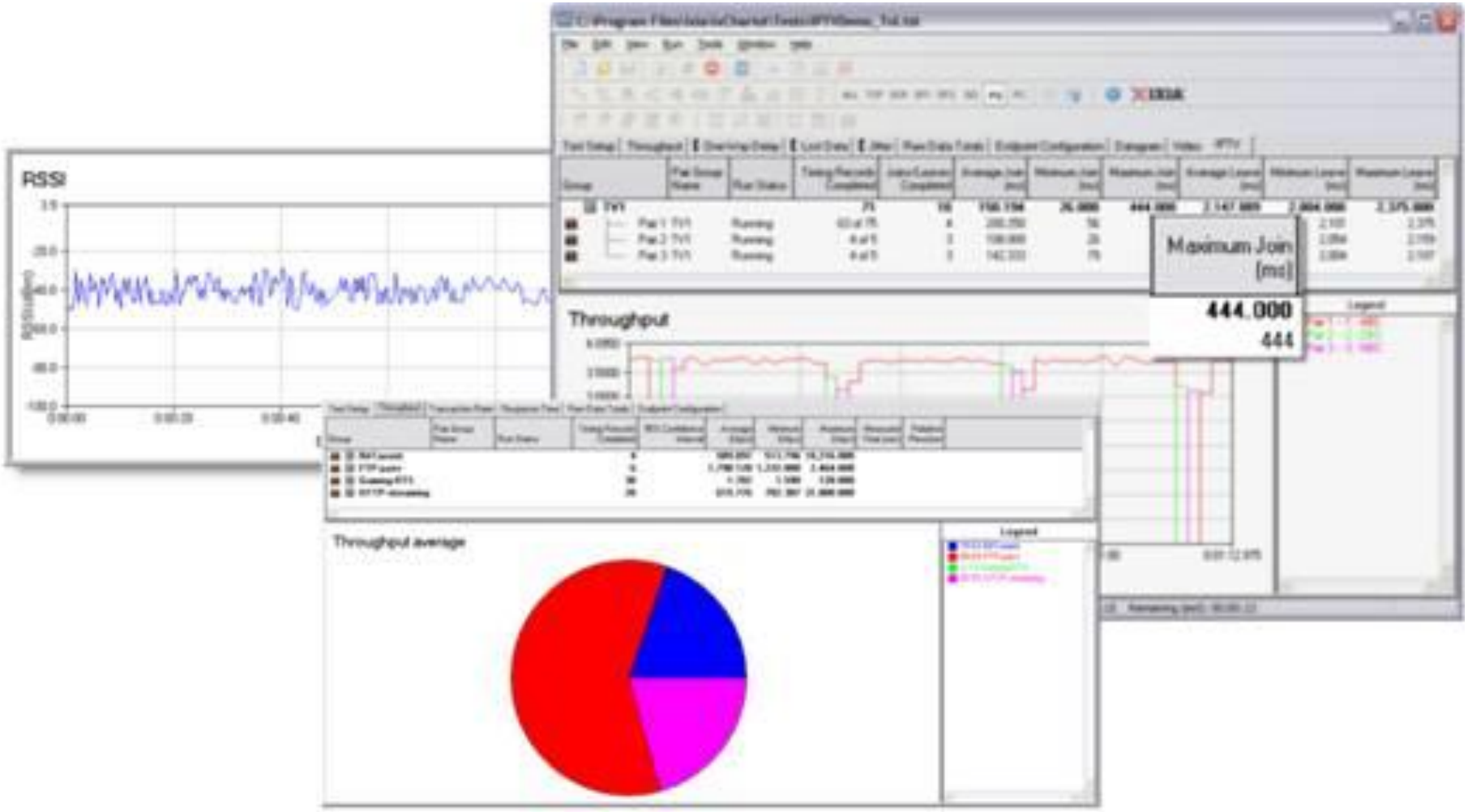
Ping for Performance Tests

- The best known, and most widely used active measurement tool is *ping*
- A sender generates an ICMP echo request packet, and directs it to a target system and starts a timer.
- The target system simply reverses the ICMP headers and sends the packet back to the sender as an ICMP echo reply.
- When the packet arrives at the original sender's system, the timer is halted and the elapsed time is reported along with TTL and packet loss

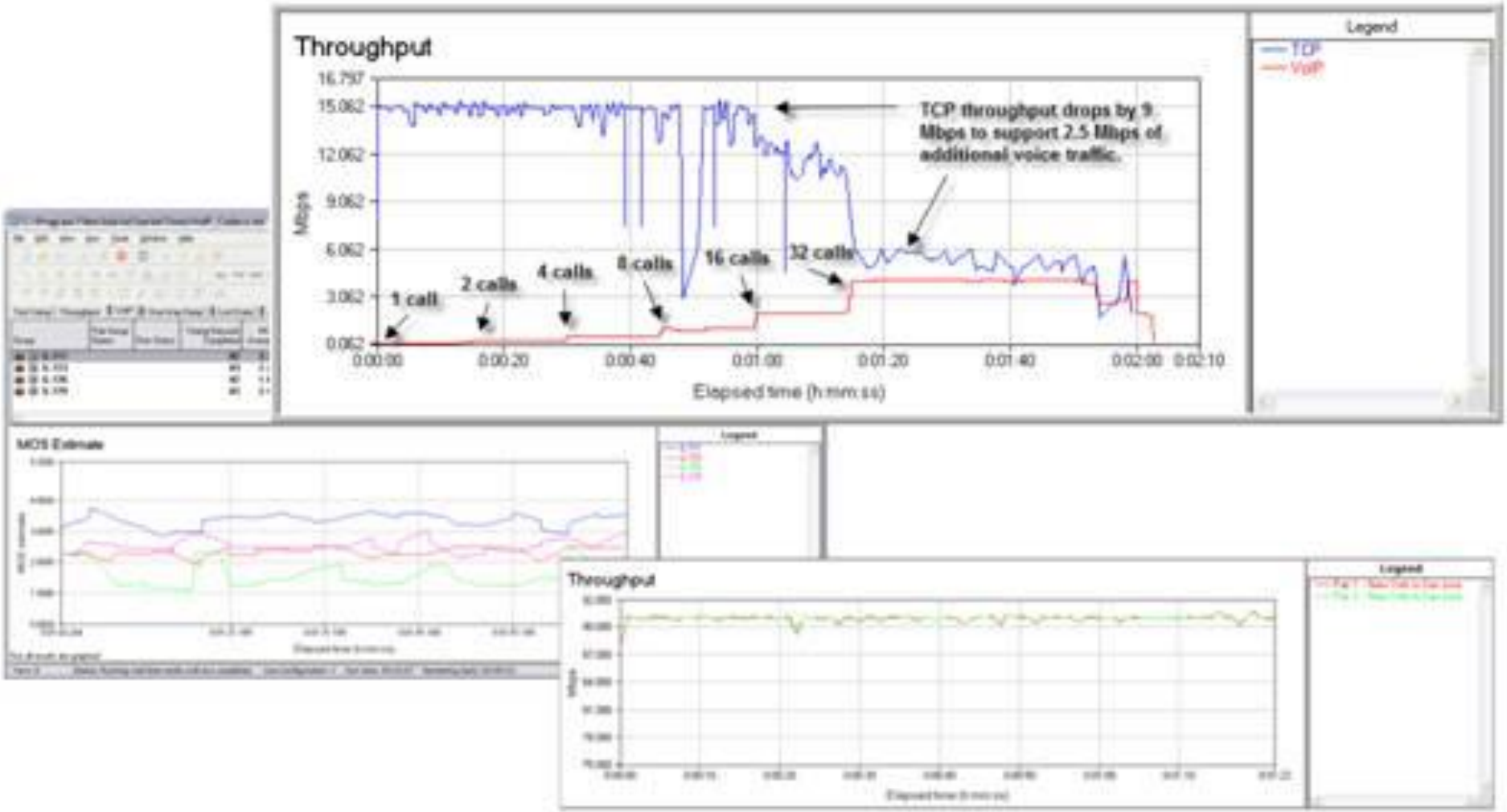
Ixia Endpoints

- IxChariot test agents, called Performance Endpoints, are installed on distributed systems to measure throughput, latency, loss and jitter using stateful TCP and UDP traffic.
- IxChariot assesses the performance of many popular enterprise applications running across local, global or virtualized infrastructures.

IxChariot GUI

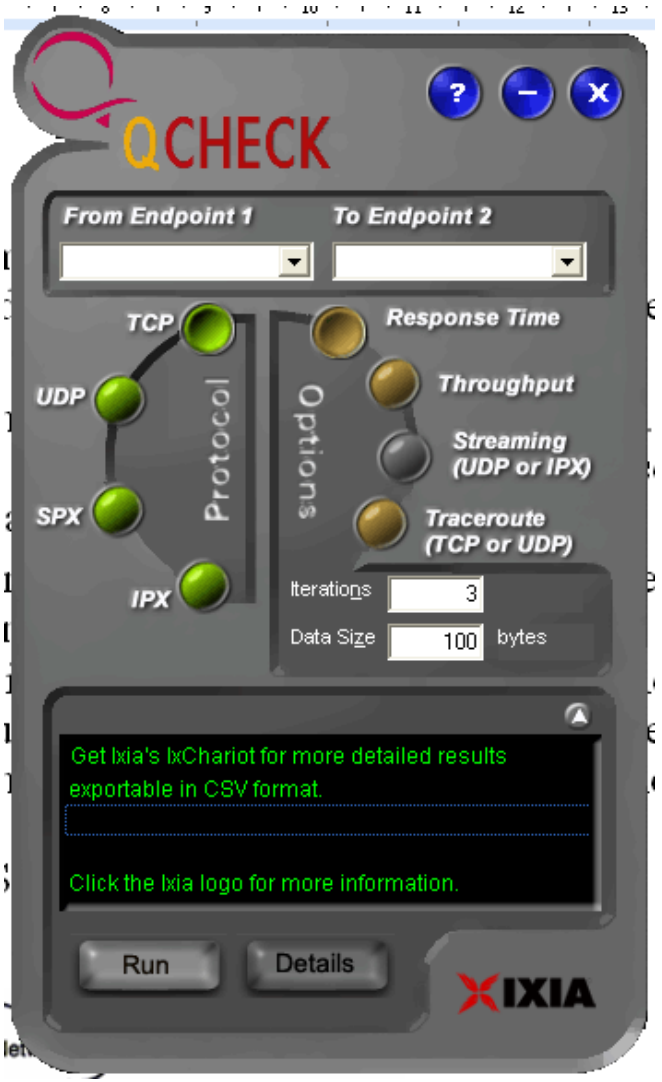


IxChariot Statistics



QCheck

QCheck is a free software built using IxChariot technology. IxChariot is a powerful network assessment tool which is used to test networks and Wireless Devices.



QCheck Features

- For a response time test, QCheck returns the minimum, maximum and average number of seconds it took to complete a transaction
- For a throughput test, QCheck returns the amount of data per second that was successfully sent between the two endpoints
- For a streaming test, QCheck returns the rate at which the streaming data was received by the second endpoint and the amount of packet loss that occurred
- For a traceroute test, QCheck returns the number of hops, average hop latency, and the address and names of the host at each hop

Problem Solving with QCheck

Your Problem	The Qcheck Solution
Someone in accounting calls the Help Desk saying he can't access the database server.	A Qcheck response time test determines if this is a network connectivity problem or not. Qcheck can also determine if this is a problem being experienced by one user, one department, or many employees.
I've got a lot of remote employees connected to my network by 56 Kbps dial-up modems. I wonder what kind of throughput they see.	A Qcheck throughput test indicates how quickly a computer can transmit data across any network. And, from your desk, you can drive Qcheck tests between any two computers on your network.
The reception from the company's videoconferencing system is lousy.	A Qcheck streaming test evaluates the network's ability to support multimedia traffic, letting you know the rate at which traffic is received and how many packets get lost along the way.
You've detected a slow connection between New York & San Francisco but you're in Houston. How do you isolate the problem?	A Qcheck on-demand traceroute initiates a traceroute test between any two workstations on your network, regardless of their location.

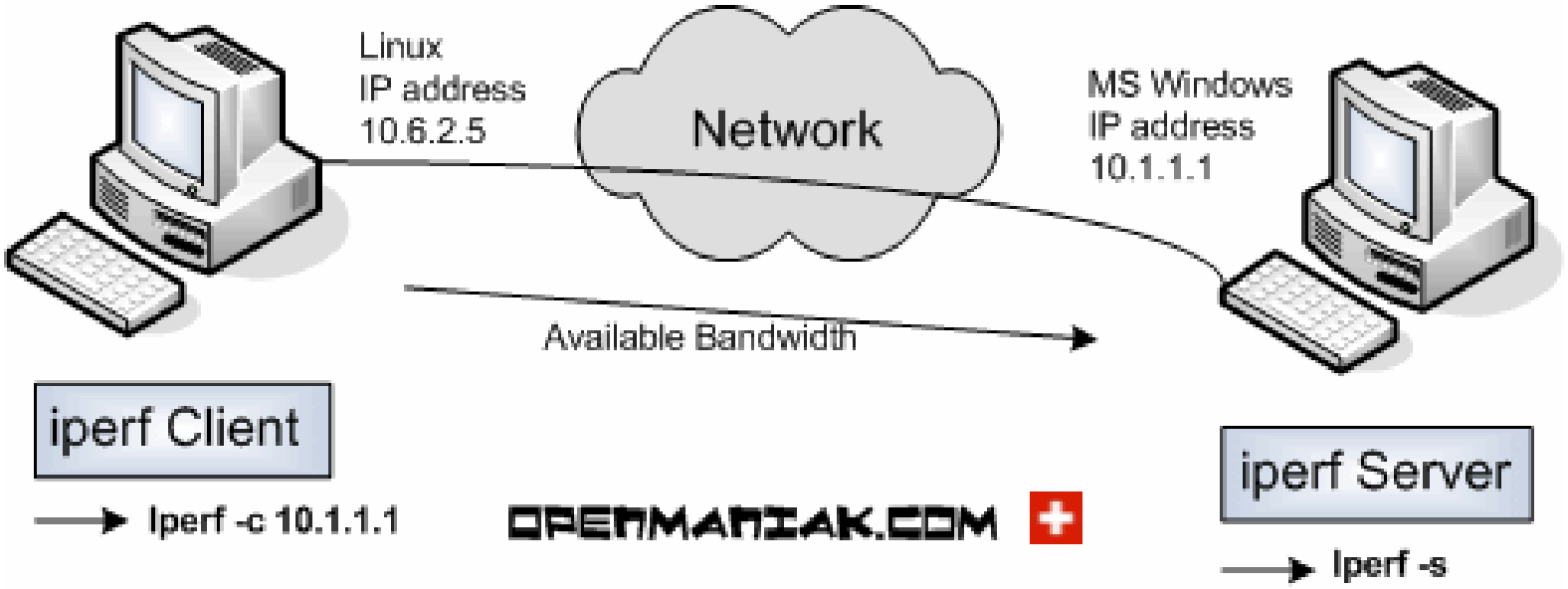
iPerf

- **iPerf** is a command line network testing tool
- It can create TCP and UDP data streams and measure the throughput of a network link.
- It employs a client and server functionality, to measure the throughput between the two ends, either unidirectonally or bidirectionally.
- When used for testing UDP capacity, iPerf allows the user to specify the datagram size.
- There is a GUI front end available called jPerf.

iPerf

- iPerf was developed to simplify TCP performance tuning by making it easy to measure maximum throughput and bandwidth.
- When used with UDP, iPerf can also measure datagram loss and jitter.
- iPerf can be run over any kind of IP network, including local Ethernet LANs, Internet access links, and Wi-Fi networks.

iPerf Use



By default, the iPerf client connects to the iPerf server on TCP port 5001 and the bandwidth displayed by iPerf is the bandwidth from the client to the server.

If you want to do UDP tests, use the –u argument. The –d and –r iPerf client arguments measure bi-directional bandwidths.

Running iPerf

- By default, iPerf runs a 10 second test, measuring total bytes transmitted and the resulting estimated bandwidth.
- Test length can be controlled by specifying time (-t seconds) or number of buffers (-n buffers).
- You can also view test results at regular intervals (-i seconds):

```
Usage: iperf [-s|-c host] [options]
       iperf [-h|--help] [-v|--version]

Client/Server:
  -f, --format    [kmKM]   format to report: Kbits, Mbits, KBytes, MBytes
  -i, --interval  #        seconds between periodic bandwidth reports
  -l, --len       #[KM]    length of buffer to read or write (default 8 KB)
  -m, --print_mss          print TCP maximum segment size (MTU - TCP/IP header)
  -o, --output    <filename> output the report or error message to this specifie
d file
  -p, --port      #        server port to listen on/connect to
  -u, --udp                use UDP rather than TCP
  -w, --window    #[KM]    TCP window size (socket buffer size)
  -B, --bind      <host>   bind to <host>, an interface or multicast address
  -C, --compatibility      for use with older versions does not sent extra msgs
  -M, --mss       #        set TCP maximum segment size (MTU - 40 bytes)
  -N, --nodelay            set TCP no delay, disabling Nagle's Algorithm
  -V, --IPv6Version        Set the domain to IPv6

Server specific:
  -s, --server             run in server mode
  -U, --single_udp         run in single threaded UDP mode
  -D, --daemon             run the server as a daemon
  -R, --remove             remove service in win32

Client specific:
  -b, --bandwidth #[KM]    for UDP, bandwidth to send at in bits/sec
                           (default 1 Mbit/sec, implies -u)
  -c, --client    <host>   run in client mode, connecting to <host>
  -d, --dualtest           Do a bidirectional test simultaneously
  -n, --num       #[KM]    number of bytes to transmit (instead of -t)
  -r, --tradeoff           Do a bidirectional test individually
  -t, --time      #        time in seconds to transmit for (default 10 secs)
  -F, --fileinput <name>   input the data to be transmitted from a file
  -I, --stdin              input the data to be transmitted from stdin
  -L, --listenport #       port to recieve bidirectional tests back on
  -P, --parallel  #        number of parallel client threads to run
  -T, --ttl       #        time-to-live, for multicast (default 1)
  -Z, --linux-congestion <algo>  set TCP congestion control algorithm (Linux onl
y)

Miscellaneous:
  -x, --reportexclude [CDMSV]   exclude C(connection) D(data) M(multicast) S(set
tings) V(server) reports
  -y, --reportstyle C      report as a Comma-Separated Values
  -h, --help               print this message and quit
  -v, --version            print version information and quit

[KM] Indicates options that support a K or M suffix for kilo- or mega-

The TCP window size option can be set by the environment variable
TCP_WINDOW_SIZE. Most other options can be set by an environment variable
IPERF_<long option name>, such as IPERF_BANDWIDTH.
```

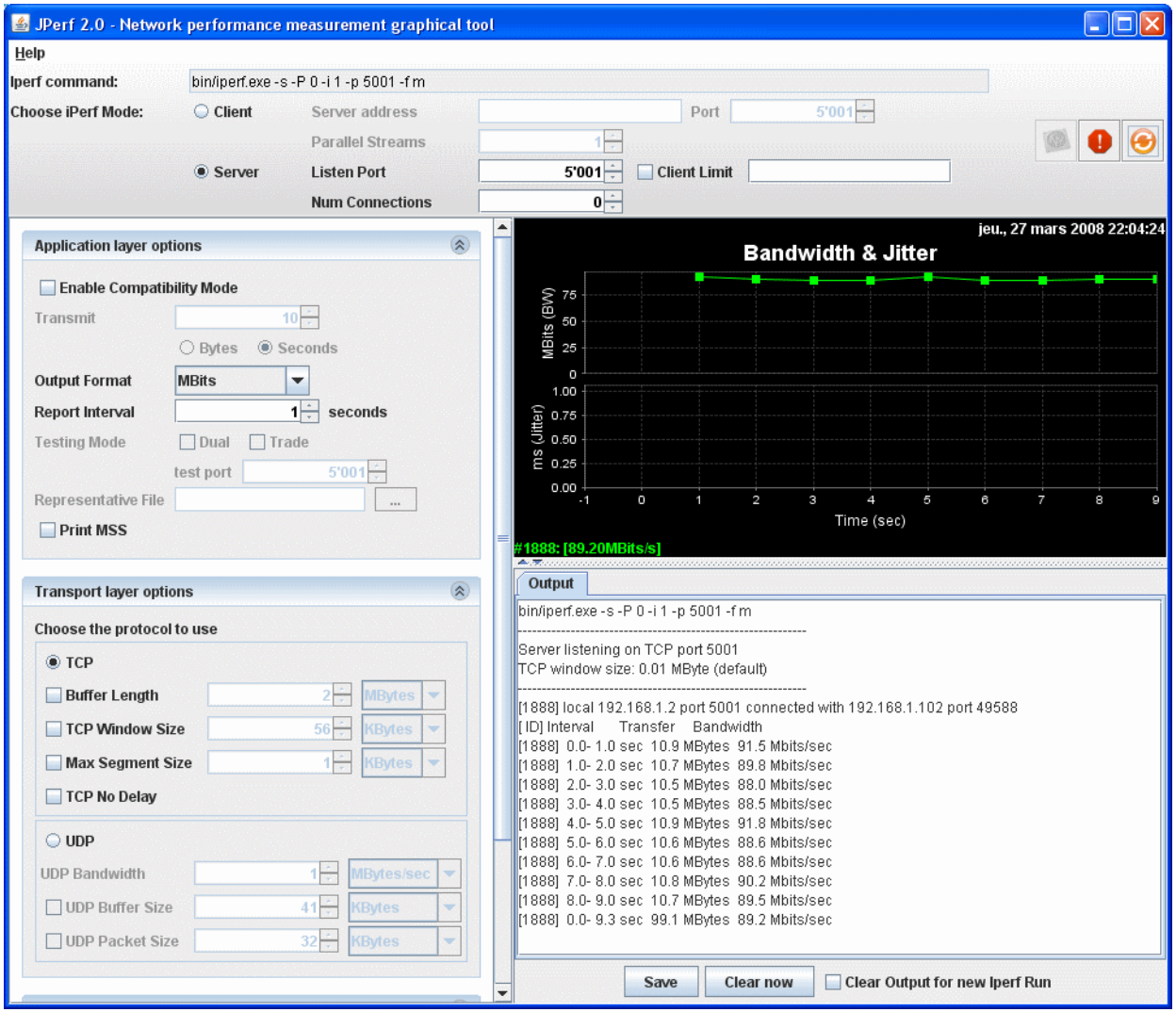
Measuring TCP Throughput

- To determine max TCP throughput, iPerf tries to send data as quickly as it can from client to server.
- Default data is sent from an 8 KB buffer, using the operating system's default TCP window size.
- To mimic a specific TCP application, you can tell your iPerf client to send data from a specified file (-F filename) or enter it interactively (-I).

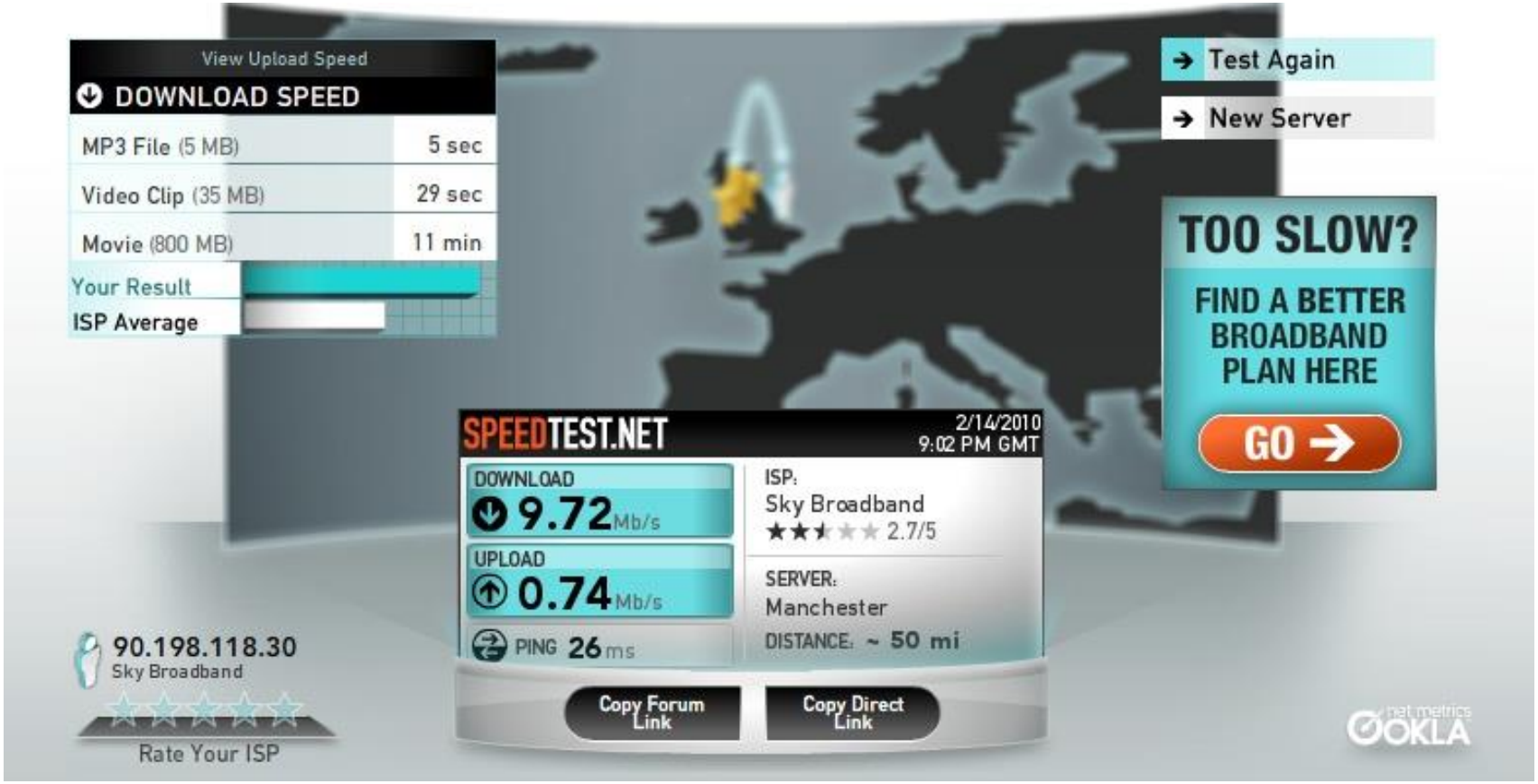
UDP Loss and Delay

- iPerf can also be used to measure UDP datagram throughput, loss, and delay. Unlike TCP tests, UDP tests do not send traffic as quickly as possible.
- Instead, iPerf tries to send 1 Mbps of traffic, packaged in 1470 byte UDP datagrams (fits into one Ethernet frame).
- This rate can be increased by supplying a target bandwidth parameter, specified in Kbps or Mbps (-b #K or --b #M).

jPerf

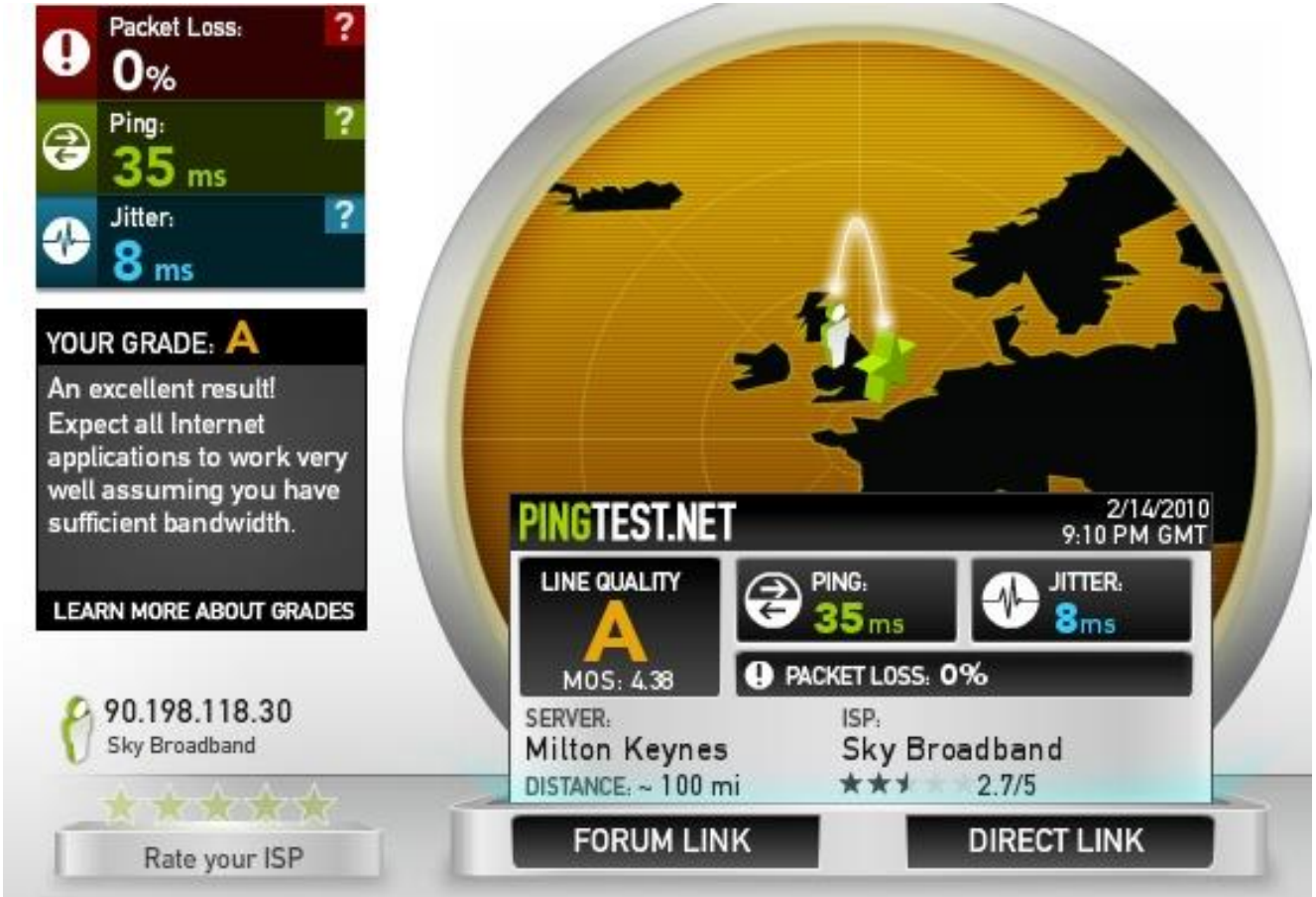

JPerf 2.0 - Network performance measurement graphical tool
Help
Iperf command:
bin/iperf.exe -s -P 0 -i 1 -p 5001 -f m
Choose iPerf Mode:
Client
Server address
Port
5'001
Parallel Streams
Server
Listen Port
5'001
Client Limit
Num Connections
0
Application layer options
Enable Compatibility Mode
Transmit
10
Bytes
Seconds
Output Format
MBits
Report Interval
1
seconds
Testing Mode
Dual
Trade
test port
5'001
Representative File
Print MSS
Transport layer options
Choose the protocol to use
TCP
Buffer Length
MBytes
TCP Window Size
KBytes
Max Segment Size
TCP No Delay
UDP
UDP Bandwidth
MBytes/sec
UDP Buffer Size
UDP Packet Size
jeu., 27 mars 2008 22:04:24
Bandwidth & Jitter
MBits (BW)
ms (Jitter)
Time (sec)
#1888: [89.20MBits/s]
Output
bin/iperf.exe -s -P 0 -i 1 -p 5001 -f m
Server listening on TCP port 5001
TCP window size: 0.01 MByte (default)
[1888] local 192.168.1.2 port 5001 connected with 192.168.1.102 port 49588
[ID] Interval Transfer Bandwidth
[1888] 0.0- 1.0 sec 10.9 MBytes 91.5 Mbits/sec
[1888] 1.0- 2.0 sec 10.7 MBytes 89.8 Mbits/sec
[1888] 2.0- 3.0 sec 10.5 MBytes 88.0 Mbits/sec
[1888] 3.0- 4.0 sec 10.5 MBytes 88.5 Mbits/sec
[1888] 4.0- 5.0 sec 10.9 MBytes 91.8 Mbits/sec
[1888] 5.0- 6.0 sec 10.6 MBytes 88.6 Mbits/sec
[1888] 6.0- 7.0 sec 10.6 MBytes 88.6 Mbits/sec
[1888] 7.0- 8.0 sec 10.8 MBytes 90.2 Mbits/sec
[1888] 8.0- 9.0 sec 10.7 MBytes 89.5 Mbits/sec
[1888] 0.0- 9.3 sec 99.1 MBytes 89.2 Mbits/sec
Save
Clear now
Clear Output for new Iperf Run

Broadband Speedtest



http://www.speedtest.net/

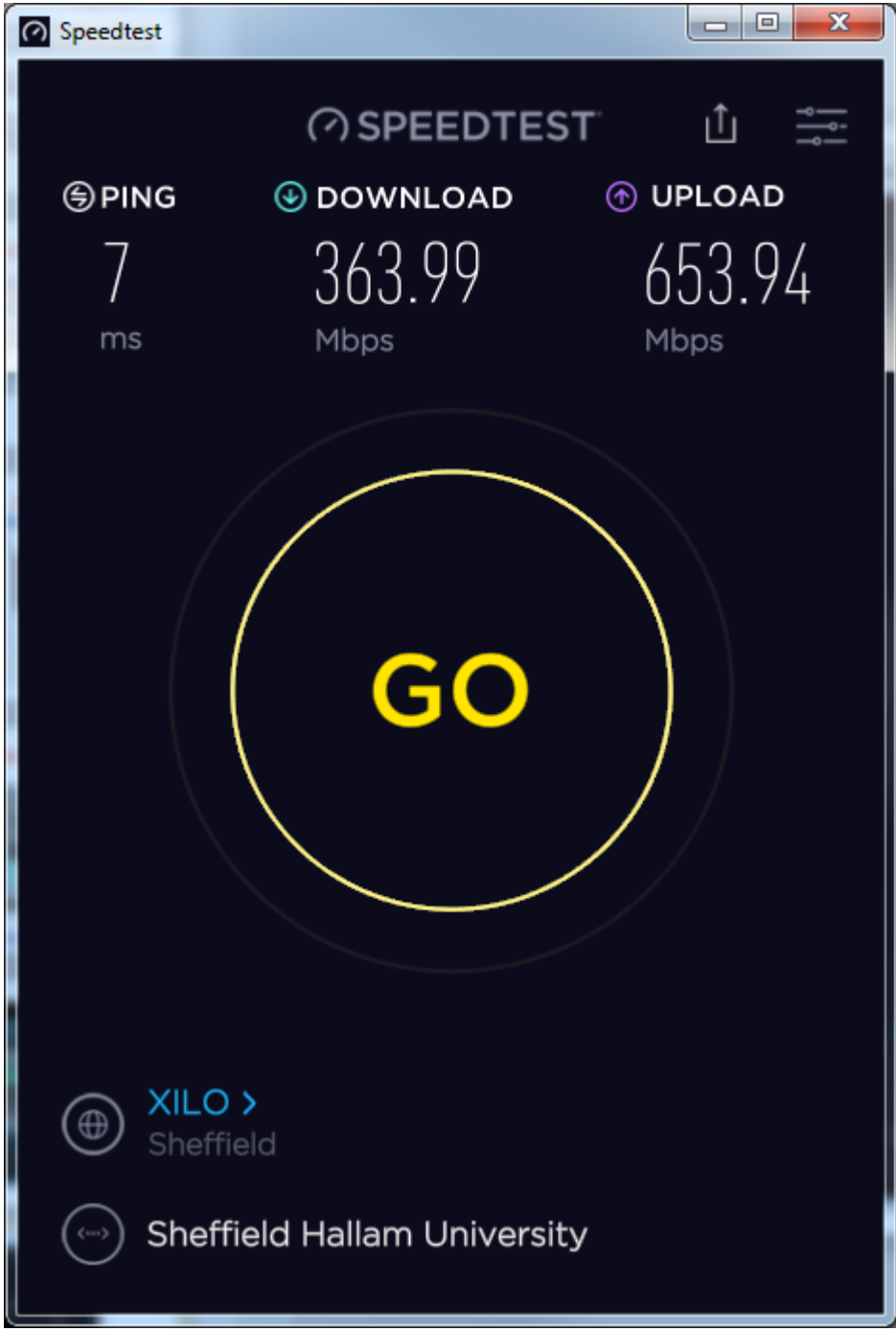

SPEEDTEST
ADVERTISE
BECOME A HOST
MY RESULTS
SUPPORT
Best Broadband By Area
Cheap Fiber Optic Cables
High Speed Internet
Best Local Internet
10 Fastest Internet
Fast DNS Servers
PING
7 ms
DOWNLOAD SPEED
657.45 Mbps
UPLOAD SPEED
816.30 Mbps
NEW SERVER
TEST AGAIN
SHARE THIS RESULT
SLOW PC PERFORMANCE?
Run a test to identify issues
and speed up your PC
START NOW
Are you on
Sheffield Hallam University?
Take our Broadband Internet Survey!
1. IMPROVE WI-FI SPEED
2. TOP WIFI INTERNET DEALS
3. BANDWIDTH SPEED TEST
143.52.85.91
Sheffield Hallam University
Rate Your ISP
Sheffield
Hosted by
XILO

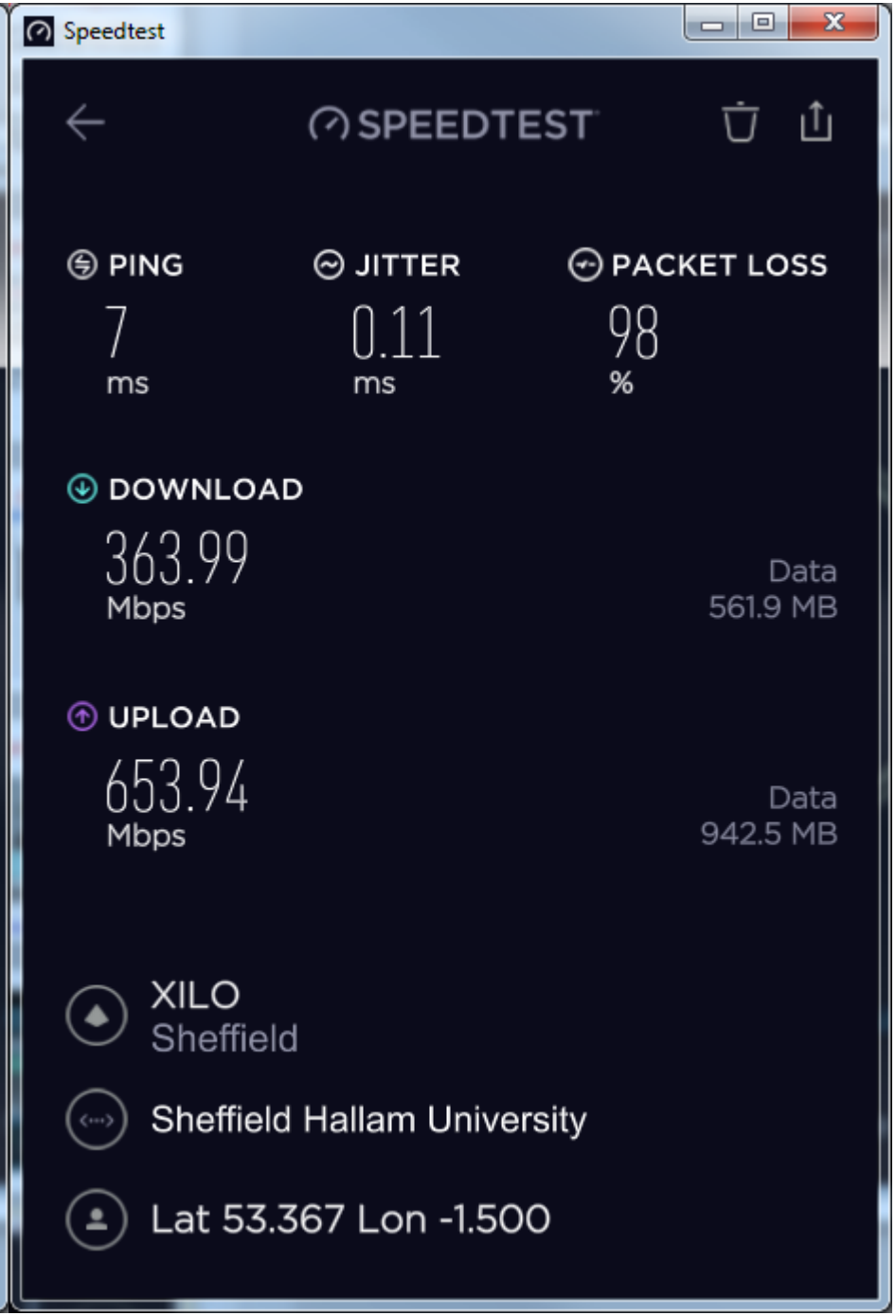


What is Pingtest.net?

Use Pingtest.net to determine the quality of your broadband Internet connection. Streaming media, voice, video communications, and online gaming require more than just raw speed. Test your connection now to get your Pingtest.net

Become
Host a Ping
region for c

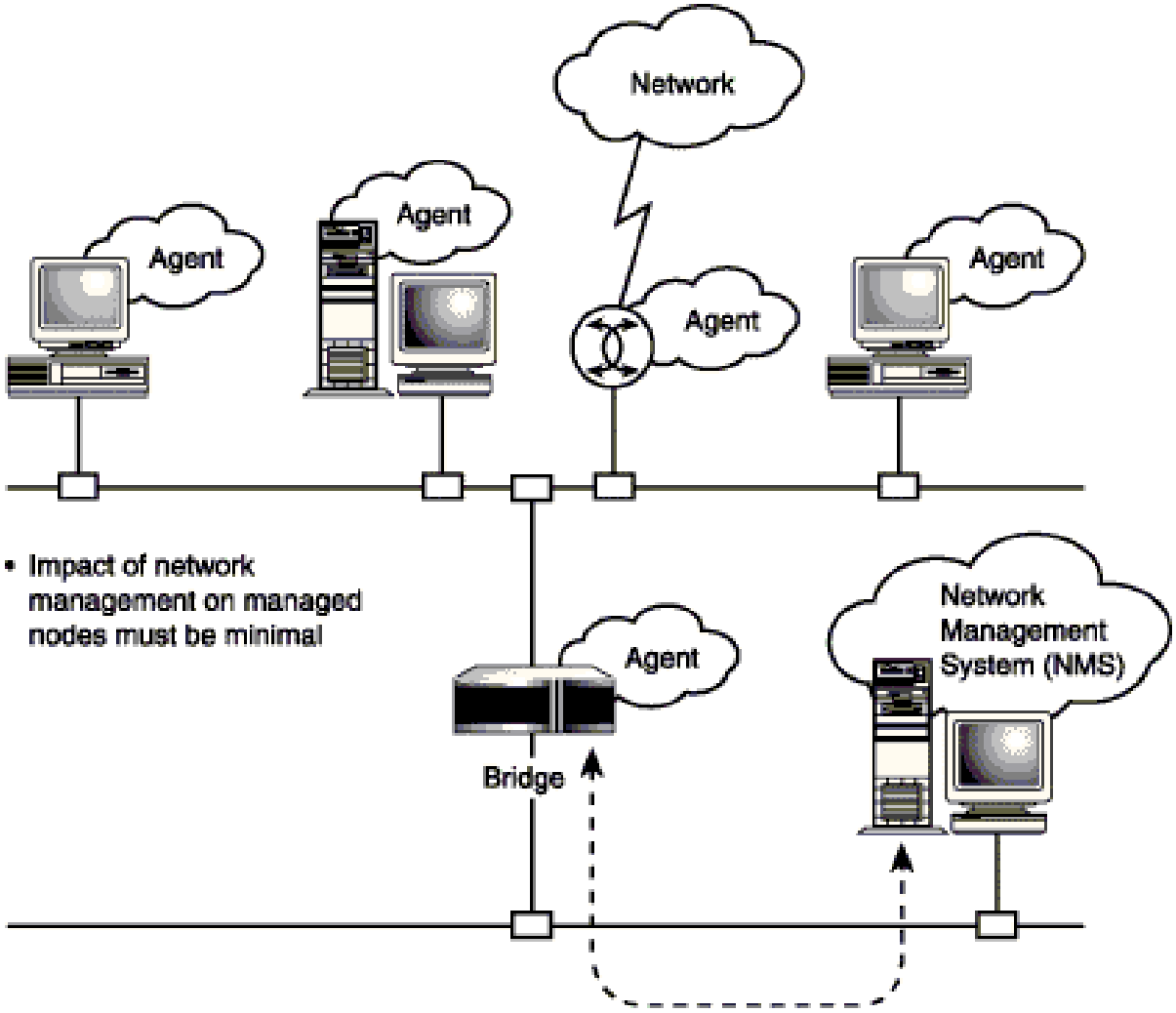

Speedtest
SPEEDTEST
PING
DOWNLOAD
UPLOAD
7
ms
363.99
Mbps
653.94
Mbps
GO
XILO >
Sheffield
Sheffield Hallam University


Speedtest
SPEEDTEST
PING
JITTER
PACKET LOSS
7
ms
0.11
ms
98
%
DOWNLOAD
363.99
Mbps
Data
561.9 MB
UPLOAD
653.94
Mbps
Data
942.5 MB
XILO
Sheffield
Sheffield Hallam University
Lat 53.367 Lon -1.500

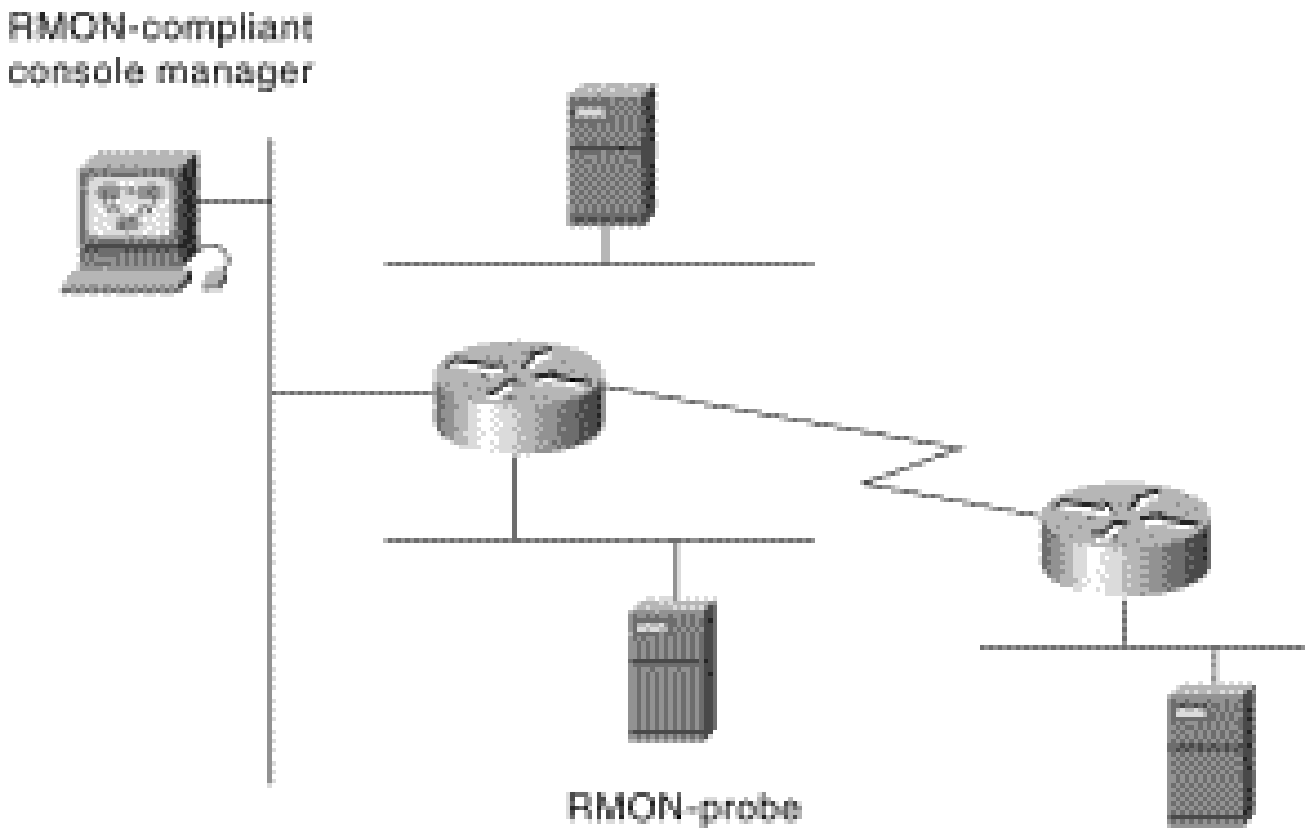
SNMP and RMON

- Simple Network Management Protocol (SNMP) was developed as a solution for network management on TCP/IP networks.
- Remote Monitoring (RMON) is used by remote monitoring agents to analyse network performance.
- RMON is a standard monitoring specification that enables various network monitors and console systems to exchange network-monitoring data.

SNMP Components



RMON Components

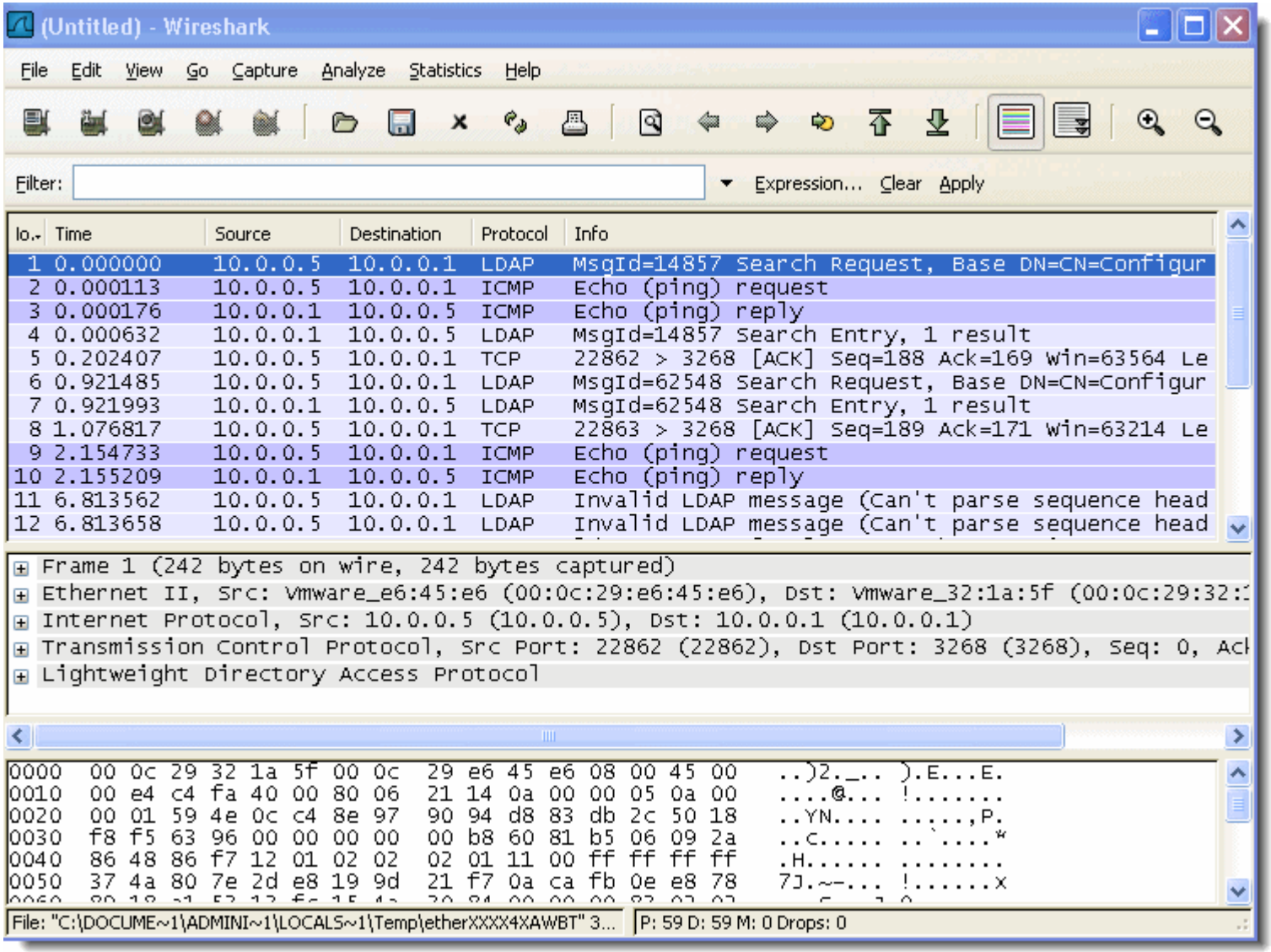


Network Monitors and Analysers

- ***Network Monitoring and Analysis*** tools are available from several vendors either as fully integrated hardware devices or software packages designed to run on notebook or desktop computers equipped with promiscuous mode LAN adapters.
- Open Source analysers such as **Wireshark** are also available.

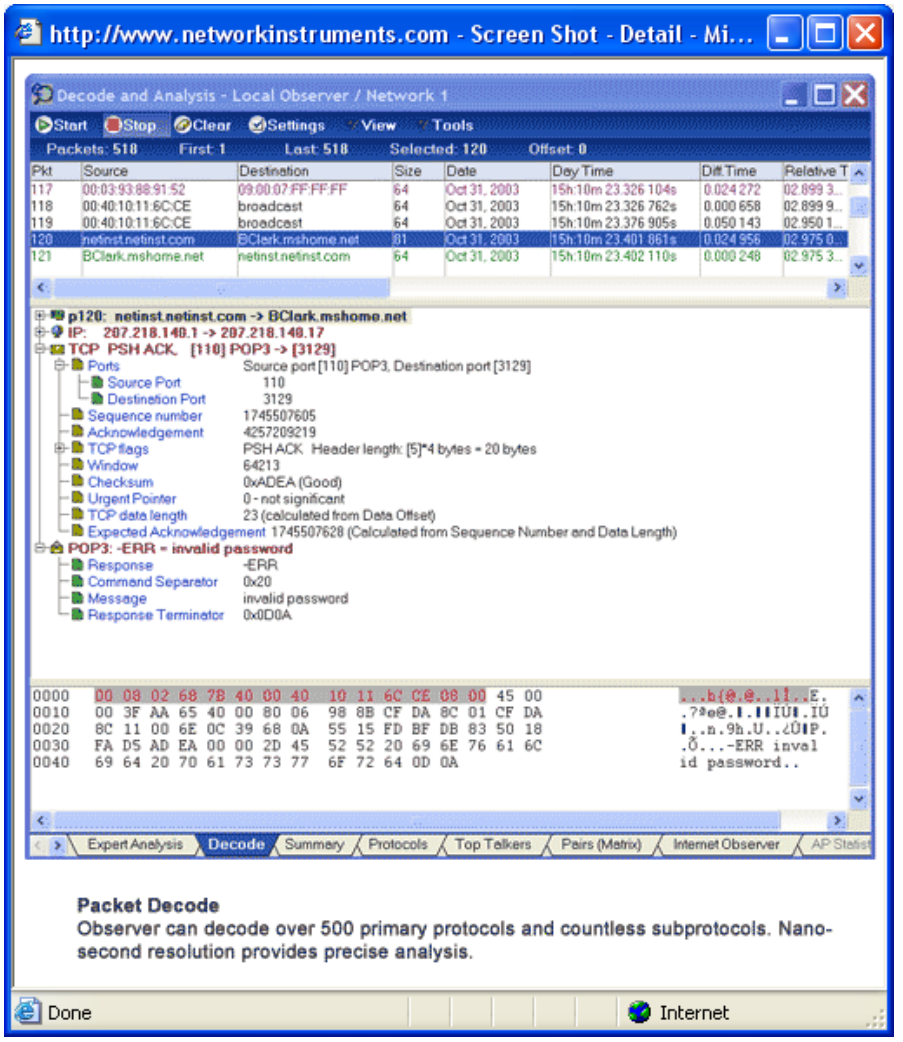
OptiView

Wireshark Packet Decode

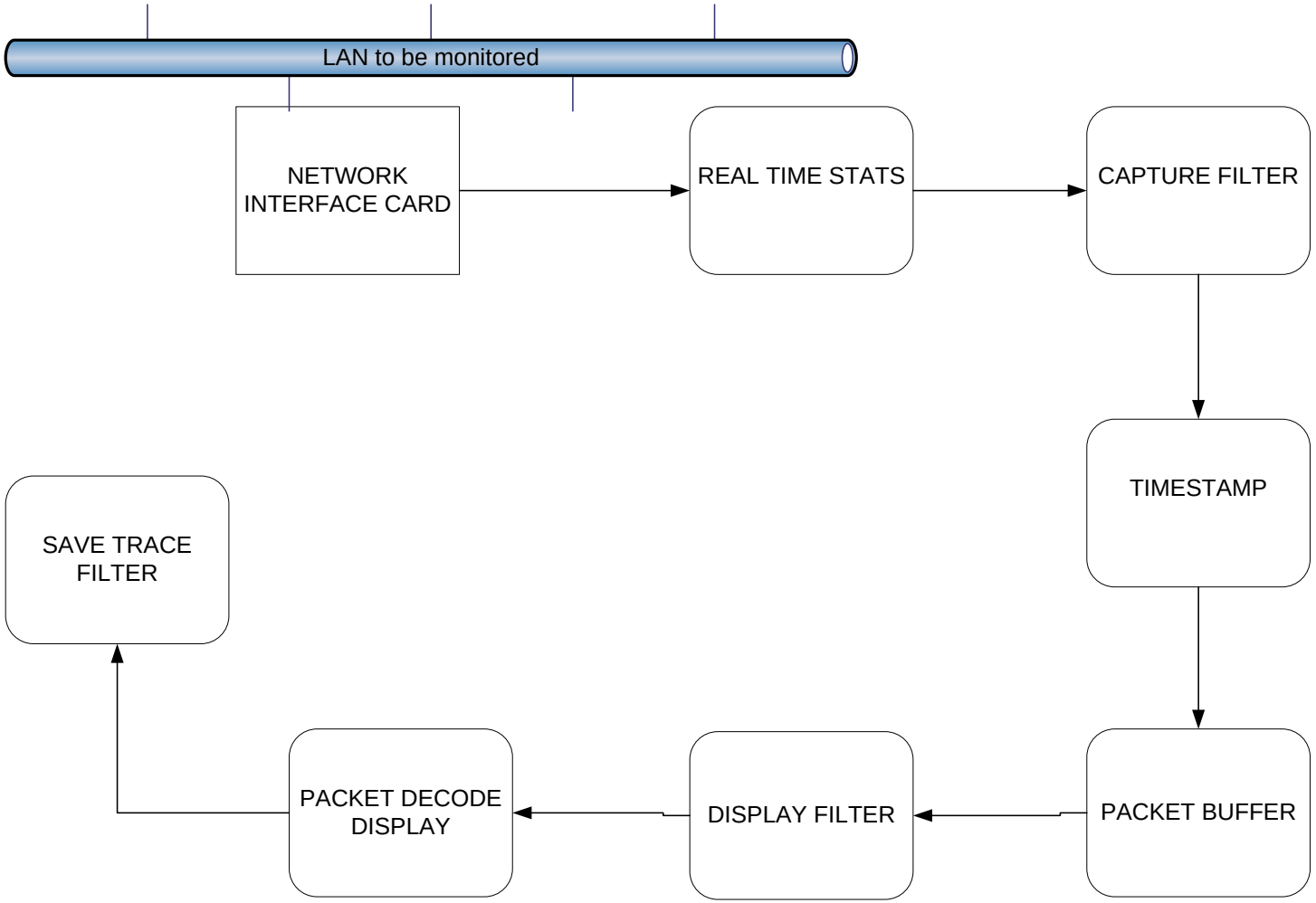


Observer and Sniffer

Popular commercial software to monitor and analyse LAN



Basic LAN Protocol Analyzer Operation

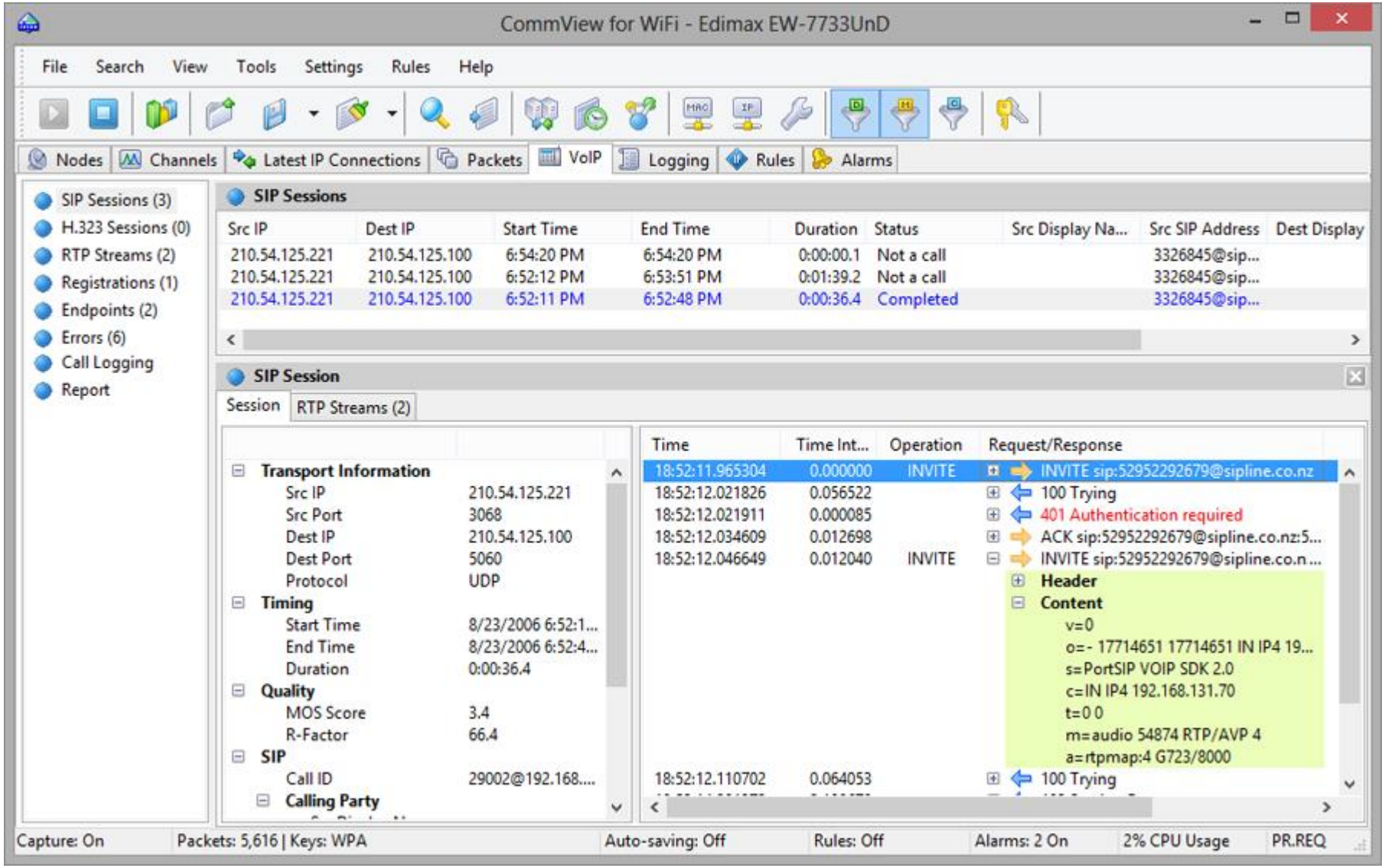


Wireless Monitoring and Measurement

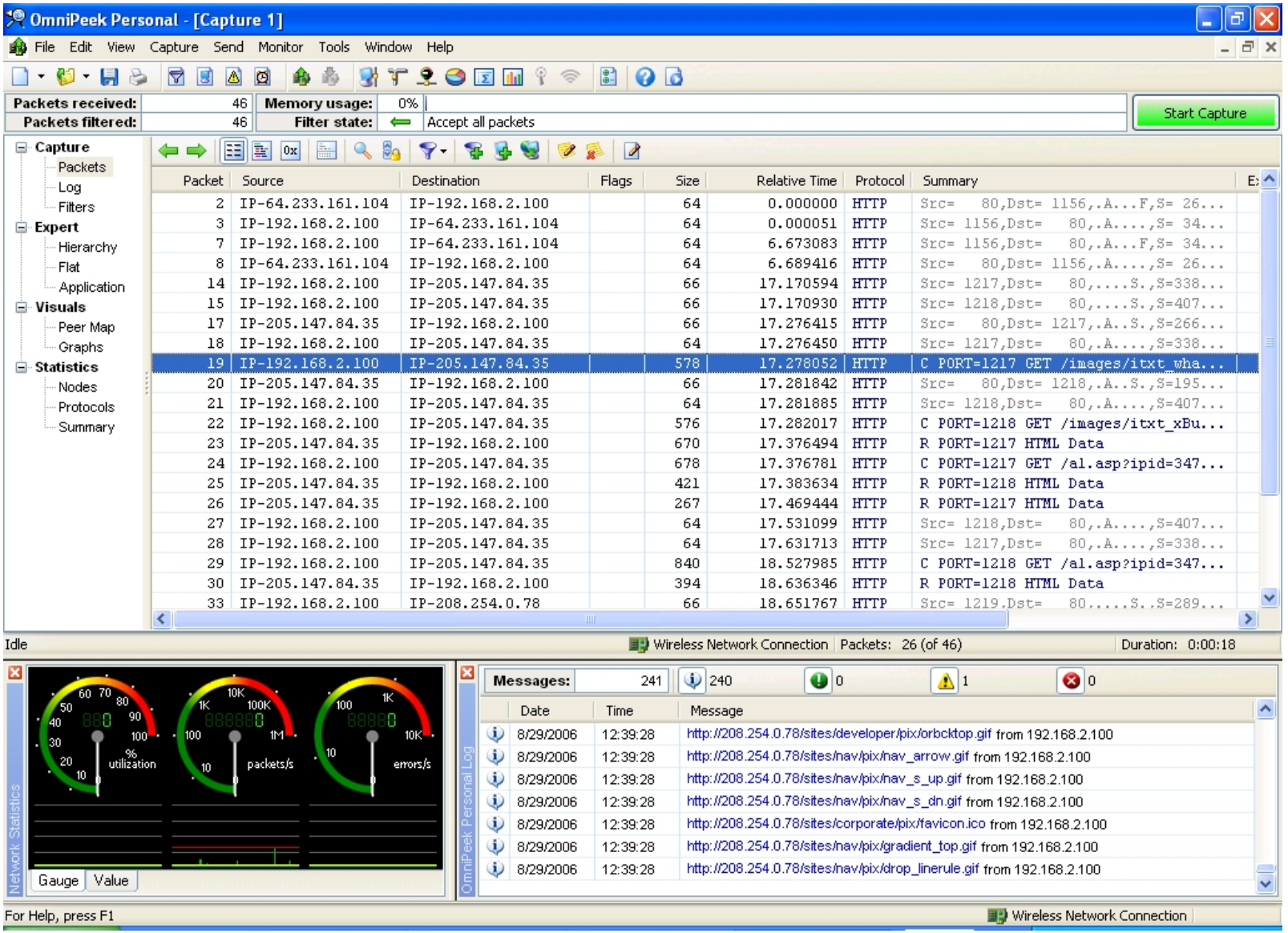
- Wireshark and Observer can be used to capture and analyze IEEE 802.11 wireless network traffic when using a computer with suitable wireless LAN adapter that can operate in monitor mode.
- Several vendors also offer more specialist WLAN tools such as **Airopeek** (now called **OmniPeek**) and **Airmagnet** or the lower cost **CommView** for WiFi.
- Spectrum analysers are invaluable for detecting and dealing RF interference problems.

AirMagnet

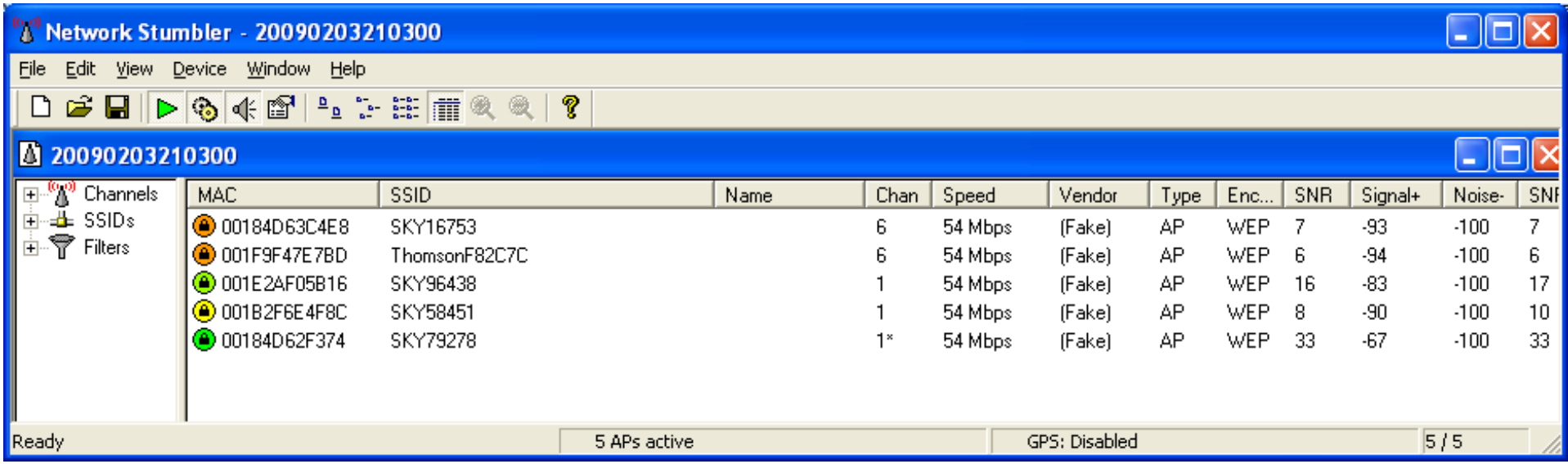
CommView for WiFi



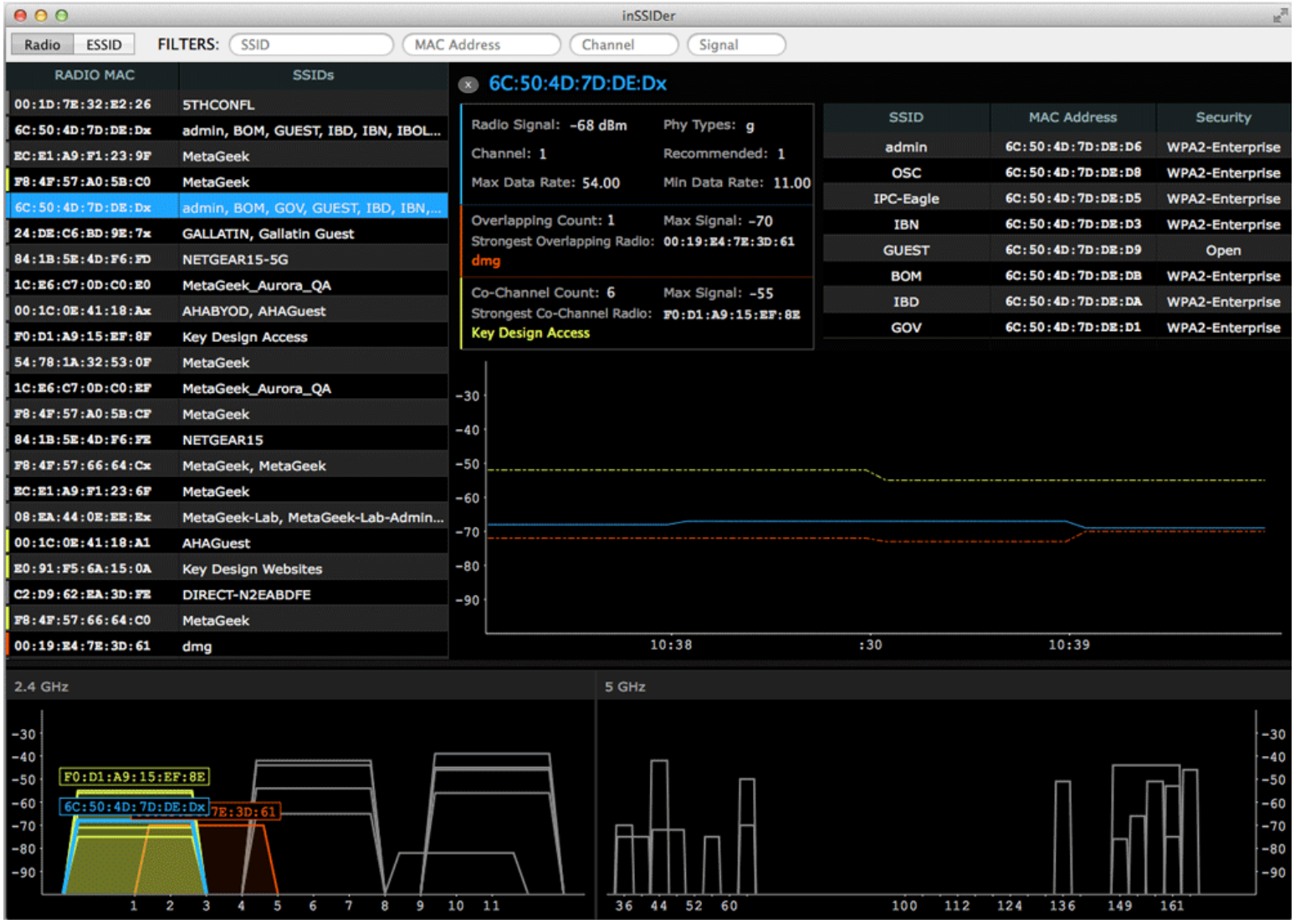
OmniPeek


OmniPeek Personal - [Capture 1]
File Edit View Capture Send Monitor Tools Window Help
Packets received: 46
Packets filtered: 46
Memory usage: 0%
Filter state: Accept all packets
Start Capture
Capture
Packets
Log
Filters
Expert
Hierarchy
Flat
Application
Visuals
Peer Map
Graphs
Statistics
Nodes
Protocols
Summary
Packet Source Destination Flags Size Relative Time Protocol Summary
2 IP-64.233.161.104 IP-192.168.2.100 64 0.000000 HTTP Src= 80,Dst= 1156,.A...F,S= 26...
3 IP-192.168.2.100 IP-64.233.161.104 64 0.000051 HTTP Src= 1156,Dst= 80,.A....,S= 34...
7 IP-192.168.2.100 IP-64.233.161.104 64 6.673083 HTTP Src= 1156,Dst= 80,.A...F,S= 34...
8 IP-64.233.161.104 IP-192.168.2.100 64 6.689416 HTTP Src= 80,Dst= 1156,.A....,S= 26...
14 IP-192.168.2.100 IP-205.147.84.35 66 17.170594 HTTP Src= 1217,Dst= 80,....S.,S=338...
15 IP-192.168.2.100 IP-205.147.84.35 66 17.170930 HTTP Src= 1218,Dst= 80,....S.,S=407...
17 IP-205.147.84.35 IP-192.168.2.100 66 17.276415 HTTP Src= 80,Dst= 1217,.A..S.,S=266...
18 IP-192.168.2.100 IP-205.147.84.35 64 17.276450 HTTP Src= 1217,Dst= 80,.A....,S=338...
19 IP-192.168.2.100 IP-205.147.84.35 578 17.278052 HTTP C PORT=1217 GET /images/itxt_wha...
20 IP-205.147.84.35 IP-192.168.2.100 66 17.281842 HTTP Src= 80,Dst= 1218,.A..S.,S=195...
21 IP-192.168.2.100 IP-205.147.84.35 64 17.281885 HTTP Src= 1218,Dst= 80,.A....,S=407...
22 IP-192.168.2.100 IP-205.147.84.35 576 17.282017 HTTP C PORT=1218 GET /images/itxt_xBu...
23 IP-205.147.84.35 IP-192.168.2.100 670 17.376494 HTTP R PORT=1217 HTML Data
24 IP-192.168.2.100 IP-205.147.84.35 678 17.376781 HTTP C PORT=1217 GET /al.asp?ipid=347...
25 IP-205.147.84.35 IP-192.168.2.100 421 17.383634 HTTP R PORT=1218 HTML Data
26 IP-205.147.84.35 IP-192.168.2.100 267 17.469444 HTTP R PORT=1217 HTML Data
27 IP-192.168.2.100 IP-205.147.84.35 64 17.531099 HTTP Src= 1218,Dst= 80,.A....,S=407...
28 IP-192.168.2.100 IP-205.147.84.35 64 17.631713 HTTP Src= 1217,Dst= 80,.A....,S=338...
29 IP-192.168.2.100 IP-205.147.84.35 840 18.527985 HTTP C PORT=1218 GET /al.asp?ipid=347...
30 IP-205.147.84.35 IP-192.168.2.100 394 18.636346 HTTP R PORT=1218 HTML Data
33 IP-192.168.2.100 IP-208.254.0.78 66 18.651767 HTTP Src= 1219,Dst= 80,....S.,S=289...
Idle
Wireless Network Connection
Packets: 26 (of 46)
Duration: 0:00:18
% utilization
packets/s
errors/s
Gauge
Value
Network Statistics
Messages: 241
240
0
1
0
Date Time Message
8/29/2006 12:39:28 http://208.254.0.78/sites/developer/pix/orbcktop.gif from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/nav/pix/nav_arrow.gif from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/nav/pix/nav_s_up.gif from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/nav/pix/nav_s_dn.gif from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/corporate/pix/favicon.ico from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/nav/pix/gradient_top.gif from 192.168.2.100
8/29/2006 12:39:28 http://208.254.0.78/sites/nav/pix/drop_linerule.gif from 192.168.2.100
OmniPeek Personal Log
For Help, press F1
Wireless Network Connection

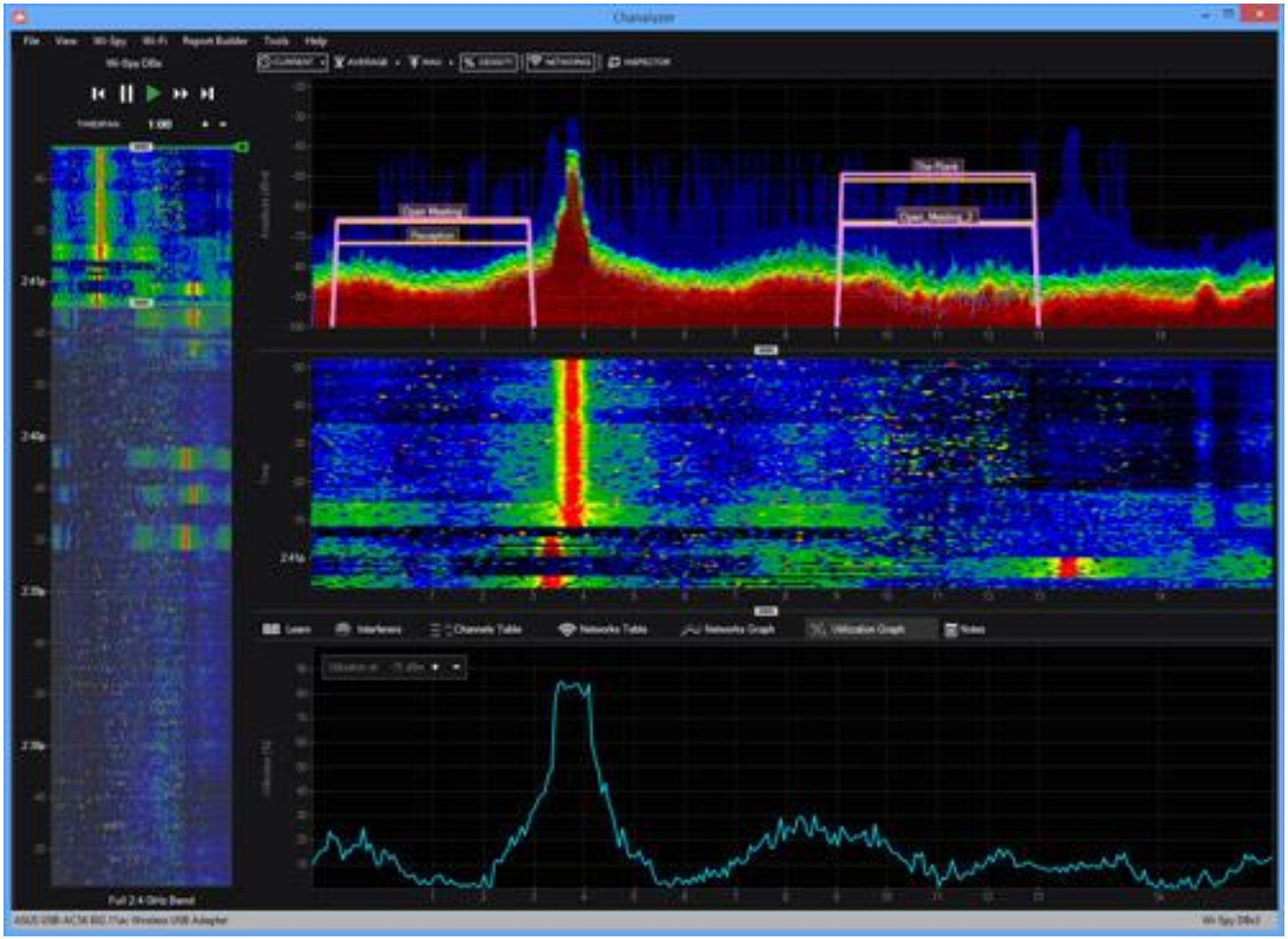
NetStumbler



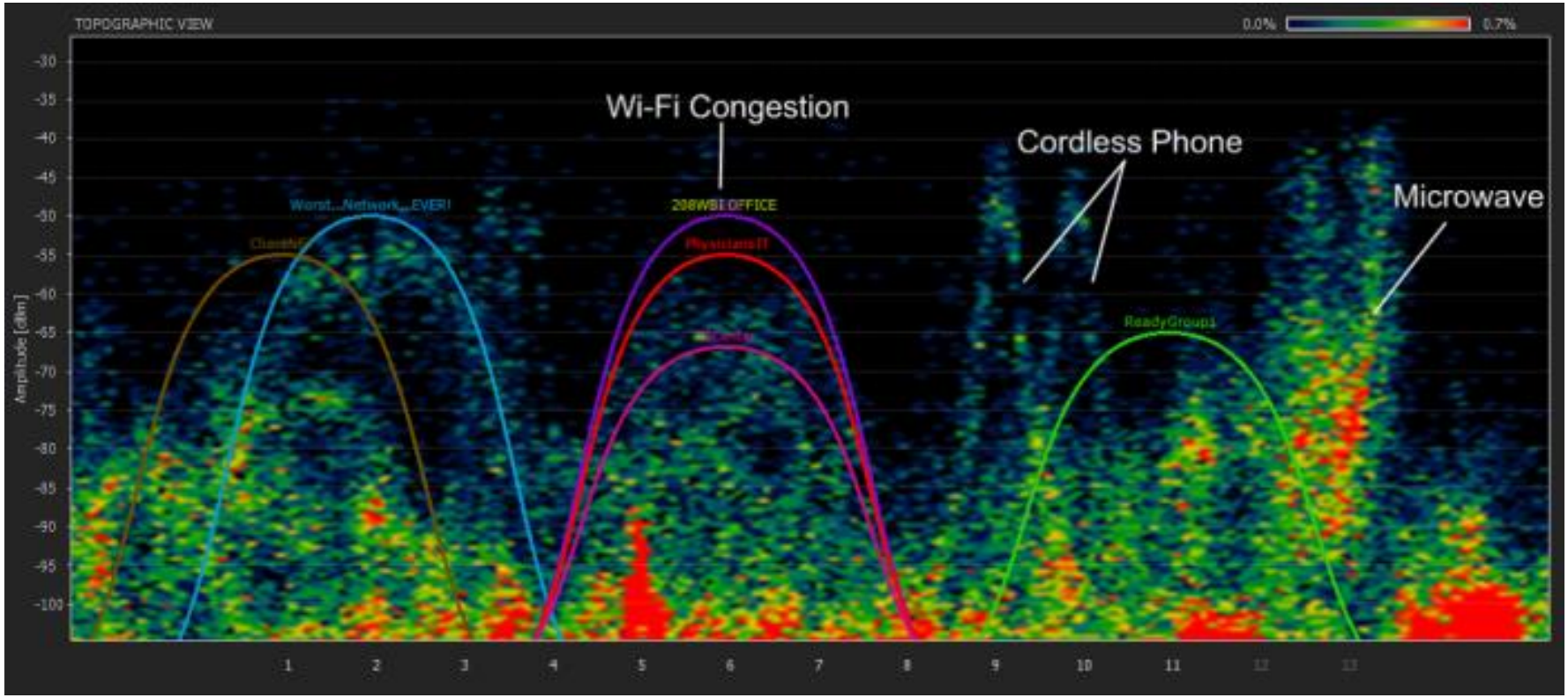
inSSIDer


inSSIDer
Radio
ESSID
FILTERS:
SSID
MAC Address
Channel
Signal
RADIO MAC
SSIDs
00:1D:7E:32:E2:26 5THCONFL
6C:50:4D:7D:DE:Dx admin, BOM, GUEST, IBD, IBN, IBOL...
EC:E1:A9:F1:23:9F MetaGeek
F8:4F:57:A0:5B:C0 MetaGeek
6C:50:4D:7D:DE:Dx admin, BOM, GOV, GUEST, IBD, IBN,...
24:DE:C6:BD:9E:7x GALLATIN, Gallatin Guest
84:1B:5E:4D:F6:FD NETGEAR15-5G
1C:E6:C7:0D:C0:E0 MetaGeek_Aurora_QA
00:1C:0E:41:18:Ax AHABYOD, AHAGuest
F0:D1:A9:15:EF:8F Key Design Access
54:78:1A:32:53:0F MetaGeek
1C:E6:C7:0D:C0:EF MetaGeek_Aurora_QA
F8:4F:57:A0:5B:CF MetaGeek
84:1B:5E:4D:F6:FE NETGEAR15
F8:4F:57:66:64:Cx MetaGeek, MetaGeek
EC:E1:A9:F1:23:6F MetaGeek
08:EA:44:0E:EE:Ex MetaGeek-Lab, MetaGeek-Lab-Admin...
00:1C:0E:41:18:A1 AHAGuest
E0:91:F5:6A:15:0A Key Design Websites
C2:D9:62:EA:3D:FE DIRECT-N2EABDFE
F8:4F:57:66:64:C0 MetaGeek
00:19:E4:7E:3D:61 dmg
6C:50:4D:7D:DE:Dx
Radio Signal: -68 dBm
Phy Types: g
Channel: 1
Recommended: 1
Max Data Rate: 54.00
Min Data Rate: 11.00
Overlapping Count: 1
Max Signal: -70
Strongest Overlapping Radio: 00:19:E4:7E:3D:61
dmg
Co-Channel Count: 6
Max Signal: -55
Strongest Co-Channel Radio: F0:D1:A9:15:EF:8E
Key Design Access
SSID
MAC Address
Security
admin 6C:50:4D:7D:DE:D6 WPA2-Enterprise
OSC 6C:50:4D:7D:DE:D8 WPA2-Enterprise
IPC-Eagle 6C:50:4D:7D:DE:D5 WPA2-Enterprise
IBN 6C:50:4D:7D:DE:D3 WPA2-Enterprise
GUEST 6C:50:4D:7D:DE:D9 Open
BOM 6C:50:4D:7D:DE:DB WPA2-Enterprise
IBD 6C:50:4D:7D:DE:DA WPA2-Enterprise
GOV 6C:50:4D:7D:DE:D1 WPA2-Enterprise
-30
-40
-50
-60
-70
-80
-90
10:38
:30
10:39
2.4 GHz
F0:D1:A9:15:EF:8E
6C:50:4D:7D:DE:Dx
7E:3D:61
1 2 3 4 5 6 7 8 9 10 11
5 GHz
36 44 52 60 100 112 124 136 149 161

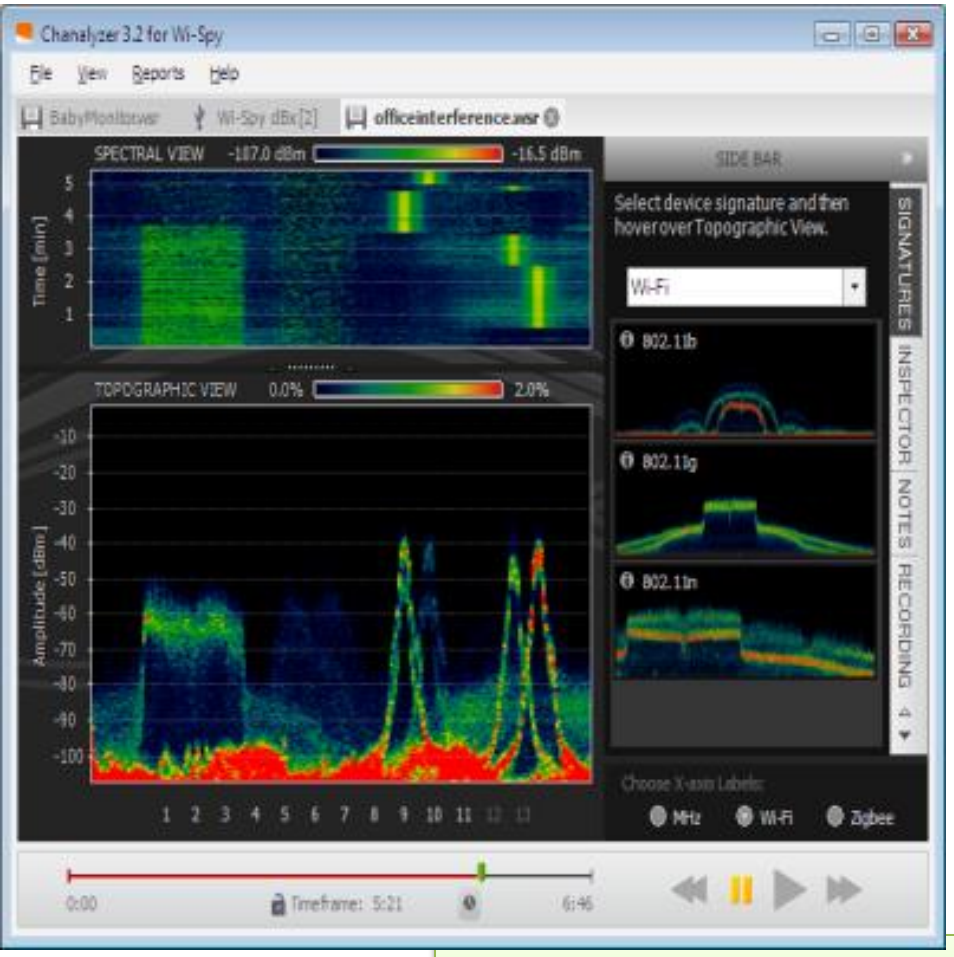
WiSpy Spectrum Analyser



WiSpy Interference Detection



WiSpy and Chanelyzer



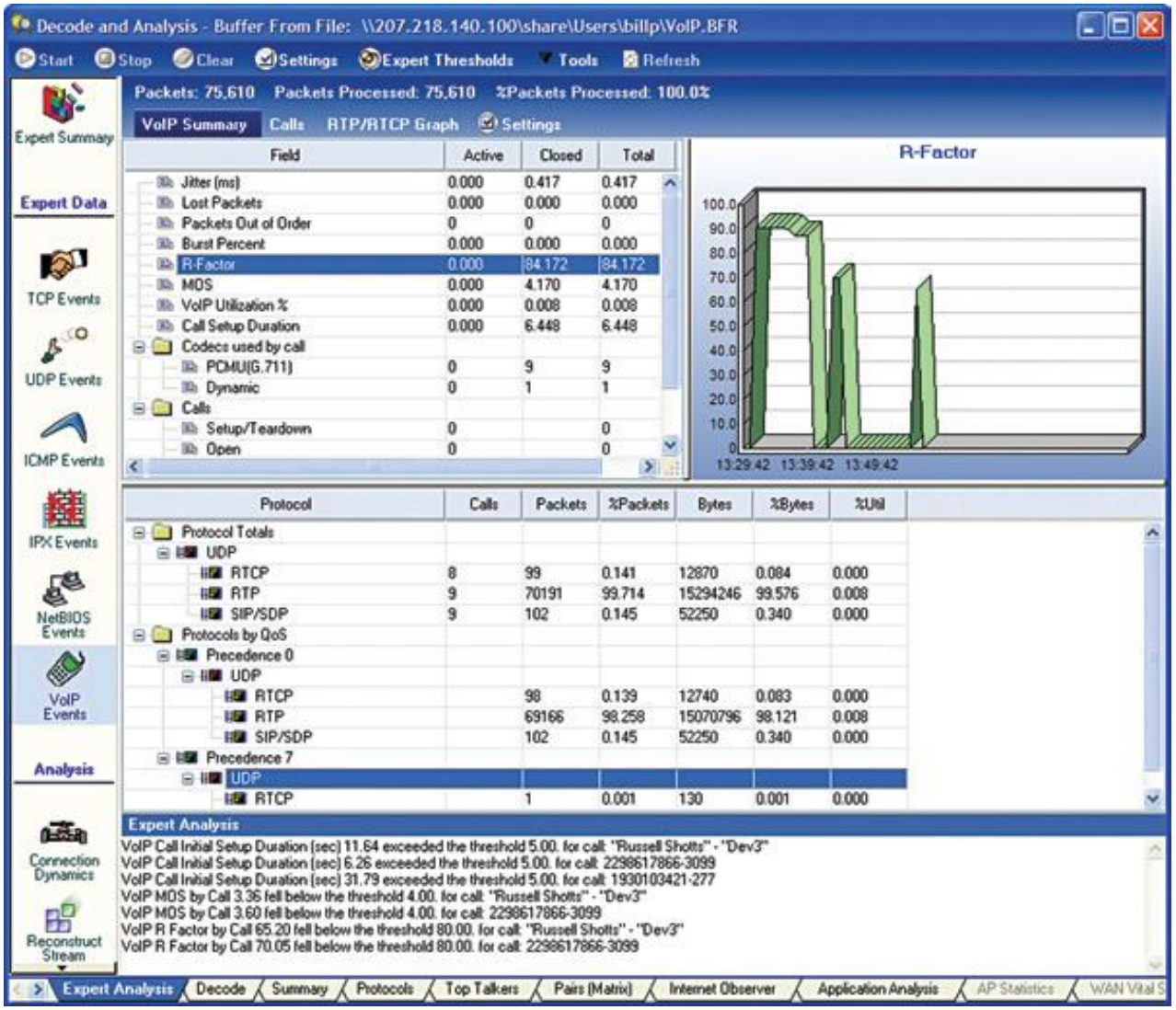
Chanalyzer turns data collected from a Wi-Spy into highly interactive charts and graphs, allowing users to “visualize” their wireless landscape. Together, Wi-Spy and Chanalyzer enable both enterprise and small business users to visualize, troubleshoot, and optimize their wireless networks.

See video at
http://www.metageek.net/products/wi-spy-24x

Real Time Quality Measurements

- Quality of Service (QoS) measurements are essential in measuring the real time performance of networks.
- Observer, Wireshark and some of the other network analyzers now have expert analysis features that can measure the metrics from which QoS assessments can be made.
- Observer VoIP Expert for instance will decode RTP traffic and format the results for latency, jitter and packet loss to provide graphical information and overall MOS scores or R-Factor.

Observer VoIP Expert

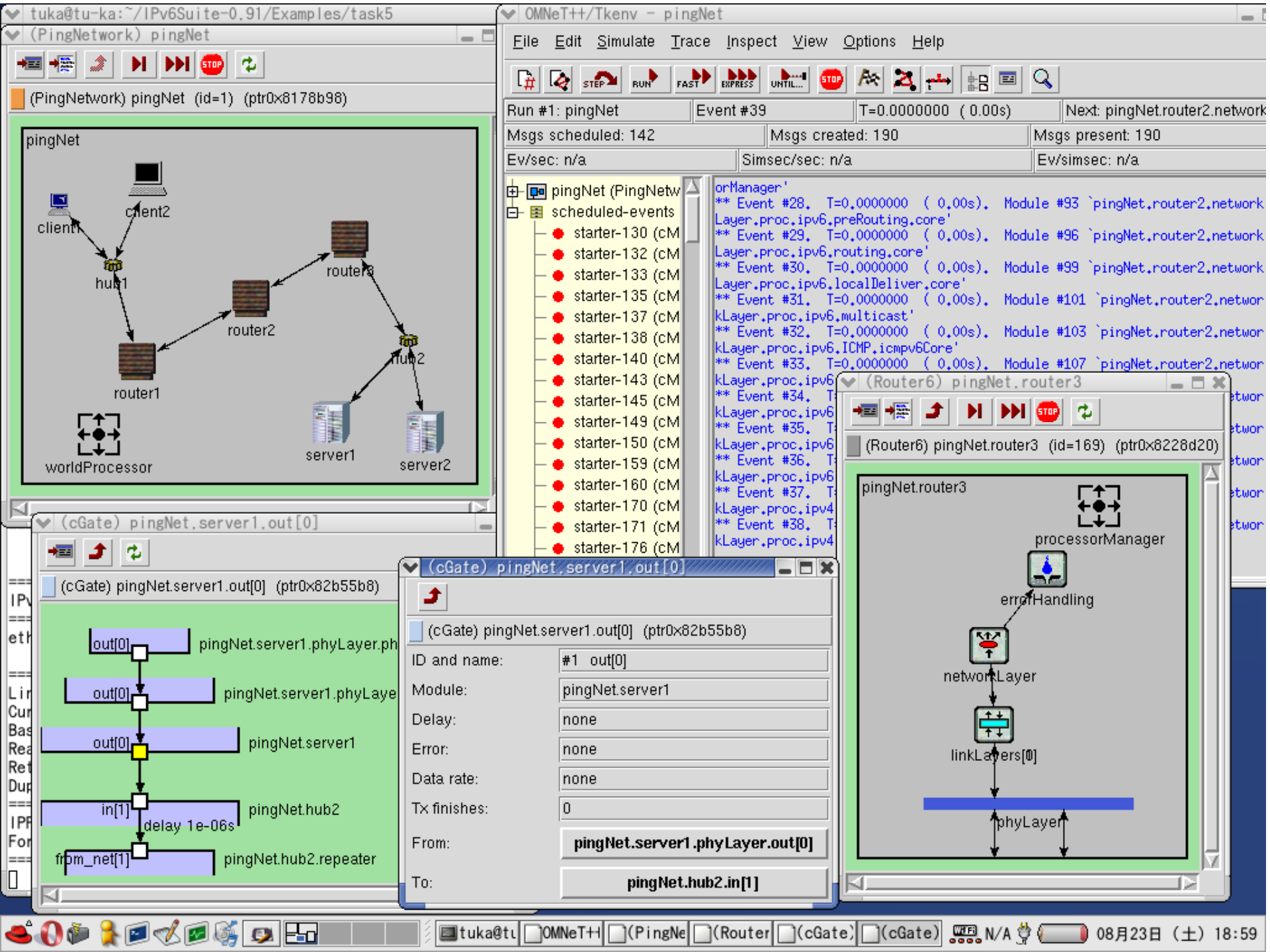


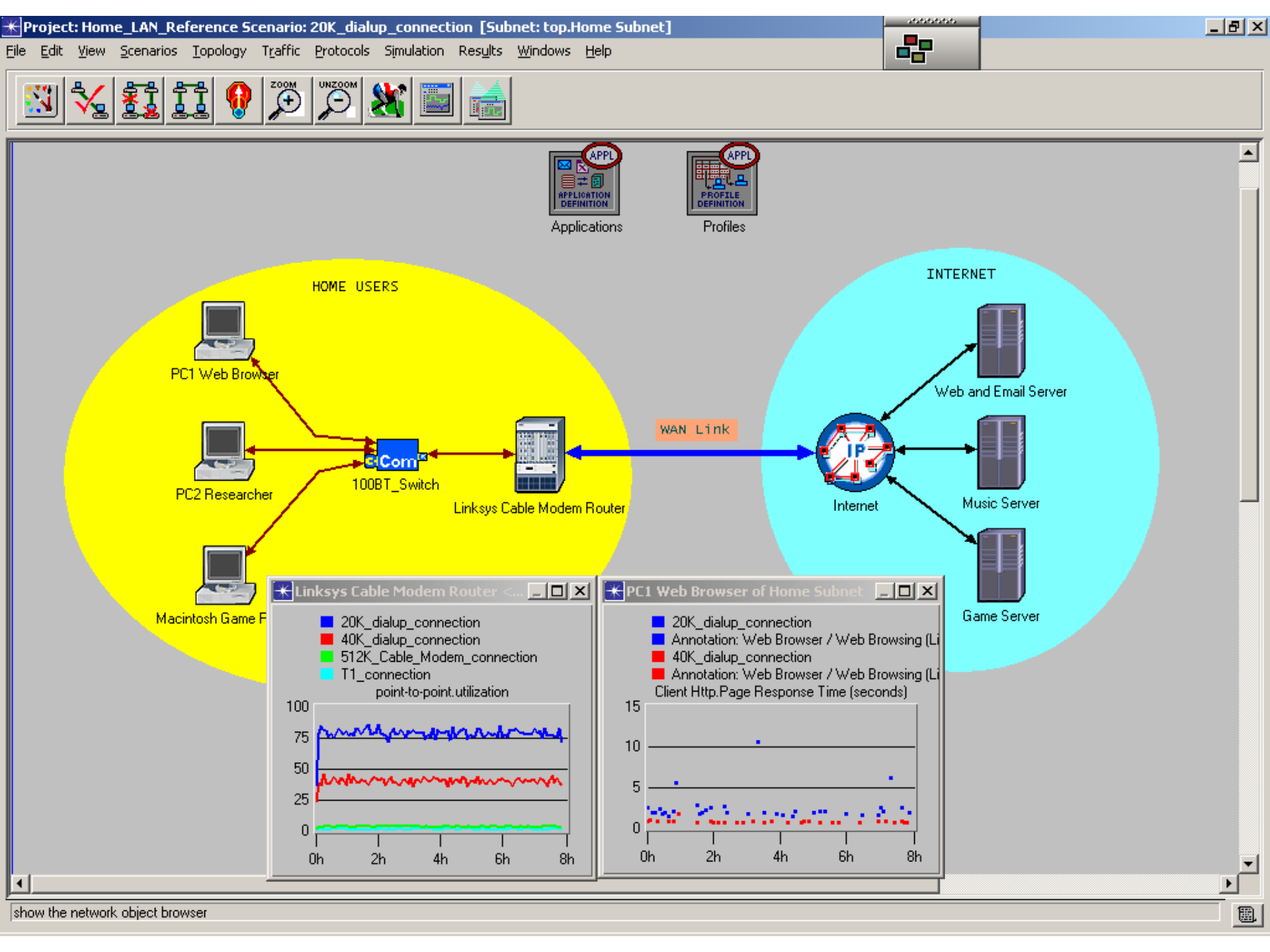
Simulation

- To diagnose problems or test new applications on a complex network, you may need to simulate the network.
- Use a simulation program to build a software model of key network elements and test how well the model functions with various traffic loads or network designs.

Simulators

- *Riverbed Modeler (OPNET) is a* commercial packages, which allow networks to be simulated either to assist with the research and design of new networks or to model an existing network and help predict the effect of changes on network performance (change simulation or What-if Analysis).
- Open source simulator software such as NS-2, NS-3 and OMNeT++ are also available.


tuka@tu-ka:~/IPv6Suite-0.91/Examples/task5
(PingNetwork) pingNet
(PingNetwork) pingNet (id=1) (ptr0x8178b98)
pingNet
client2
client1
hub1
router3
router2
router1
hub2
server1
server2
worldProcessor
OMNeT++/Tkenv - pingNet
File Edit Simulate Trace Inspect View Options Help
STEP RUN FAST EXPRESS UNTIL... STOP
Run #1: pingNet
Event #39
T=0.0000000 (0.00s)
Next: pingNet.router2.network
Msgs scheduled: 142
Msgs created: 190
Msgs present: 190
Ev/sec: n/a
Simsec/sec: n/a
Ev/simsec: n/a
pingNet (PingNetw
scheduled-events
starter-130 (cM
starter-132 (cM
starter-133 (cM
starter-135 (cM
starter-137 (cM
starter-138 (cM
starter-140 (cM
starter-143 (cM
starter-145 (cM
starter-149 (cM
starter-150 (cM
starter-159 (cM
starter-160 (cM
starter-170 (cM
starter-171 (cM
starter-176 (cM
orManager'
** Event #28. T=0.0000000 (0.00s). Module #93 `pingNet.router2.network
Layer.proc.ipv6.preRouting.core'
** Event #29. T=0.0000000 (0.00s). Module #96 `pingNet.router2.network
Layer.proc.ipv6.routing.core'
** Event #30. T=0.0000000 (0.00s). Module #99 `pingNet.router2.network
Layer.proc.ipv6.localDeliver.core'
** Event #31. T=0.0000000 (0.00s). Module #101 `pingNet.router2.networ
kLayer.proc.ipv6.multicast'
** Event #32. T=0.0000000 (0.00s). Module #103 `pingNet.router2.networ
kLayer.proc.ipv6.ICMP.icmpv6Core'
** Event #33. T=0.0000000 (0.00s). Module #107 `pingNet.router2.networ
(Router6) pingNet.router3
(Router6) pingNet.router3 (id=169) (ptr0x8228d20)
pingNet.router3
processorManager
errorHandling
networkLayer
linkLayers[0]
phyLayer
(cGate) pingNet.server1.out[0]
(cGate) pingNet.server1.out[0] (ptr0x82b55b8)
out[0] pingNet.server1.phyLayer.ph
out[0] pingNet.server1.phyLaye
out[0] pingNet.server1
in[1] pingNet.hub2
delay 1e-06s
from_net[1] pingNet.hub2.repeater
(cGate) pingNet.server1.out[0]
(cGate) pingNet.server1.out[0] (ptr0x82b55b8)
ID and name: #1 out[0]
Module: pingNet.server1
Delay: none
Error: none
Data rate: none
Tx finishes: 0
From: pingNet.server1.phyLayer.out[0]
To: pingNet.hub2.in[1]
08月23日 (土) 18:59


Project: Home_LAN_Reference Scenario: 20K_dialup_connection [Subnet: top.Home Subnet]
File Edit View Scenarios Topology Traffic Protocols Simulation Results Windows Help
ZOOM
UNZOOM
APPL
APPLICATION DEFINITION
Applications
APPL
PROFILE DEFINITION
Profiles
HOME USERS
PC1 Web Browser
PC2 Researcher
Macintosh Game P
100BT_Switch
Linksys Cable Modem Router
WAN Link
INTERNET
Internet
Web and Email Server
Music Server
Game Server
Linksys Cable Modem Router <...
20K_dialup_connection
40K_dialup_connection
512K_Cable_Modem_connection
T1_connection
point-to-point.utilization
100
75
50
25
0
0h
2h
4h
6h
8h
PC1 Web Browser of Home Subnet
20K_dialup_connection
Annotation: Web Browser / Web Browsing (Li
40K_dialup_connection
Annotation: Web Browser / Web Browsing (Li
Client Http.Page Response Time (seconds)
15
10
5
0
0h
2h
4h
6h
8h
show the network object browser

Emulators

- A network emulator is a device that sits on a network and mimics the behavior of network devices such as routers or parts of the system such as subnets. Actual traffic measurements are made under the control of the emulator.
- Emulators lie between simulators and live systems. They allow experiments with a high degree of reproducibility. For example, an emulator might duplicate or approximate the behaviour of an attached network device.

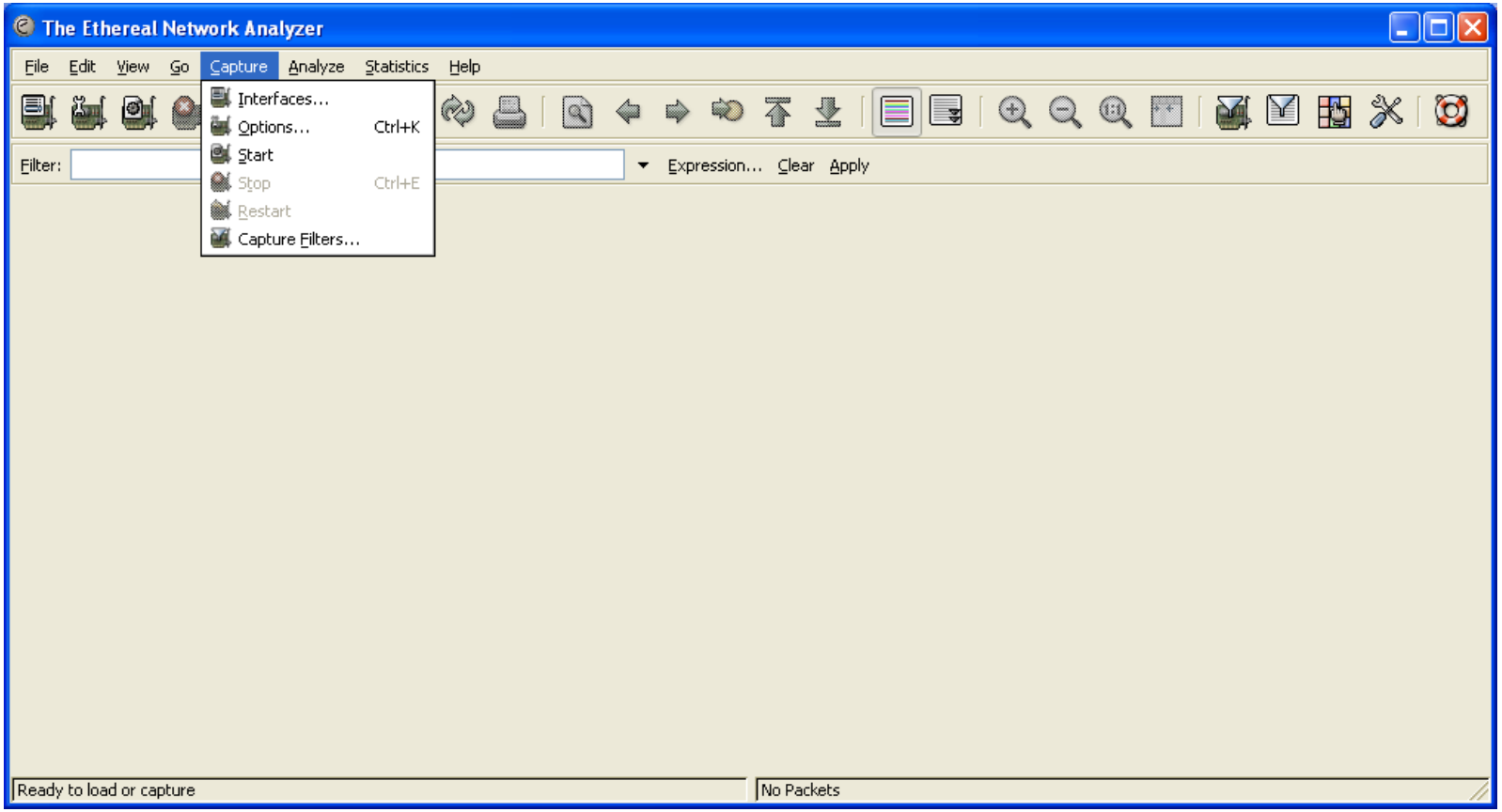
NISTNET

- NISTNET is an emulator software that is useful for internetwork testing.
- It turns a Linux computer into a router with variable performance metrics.
- Jitter, latency, bandwidth and packet loss can be altered to test the effects on network performance.

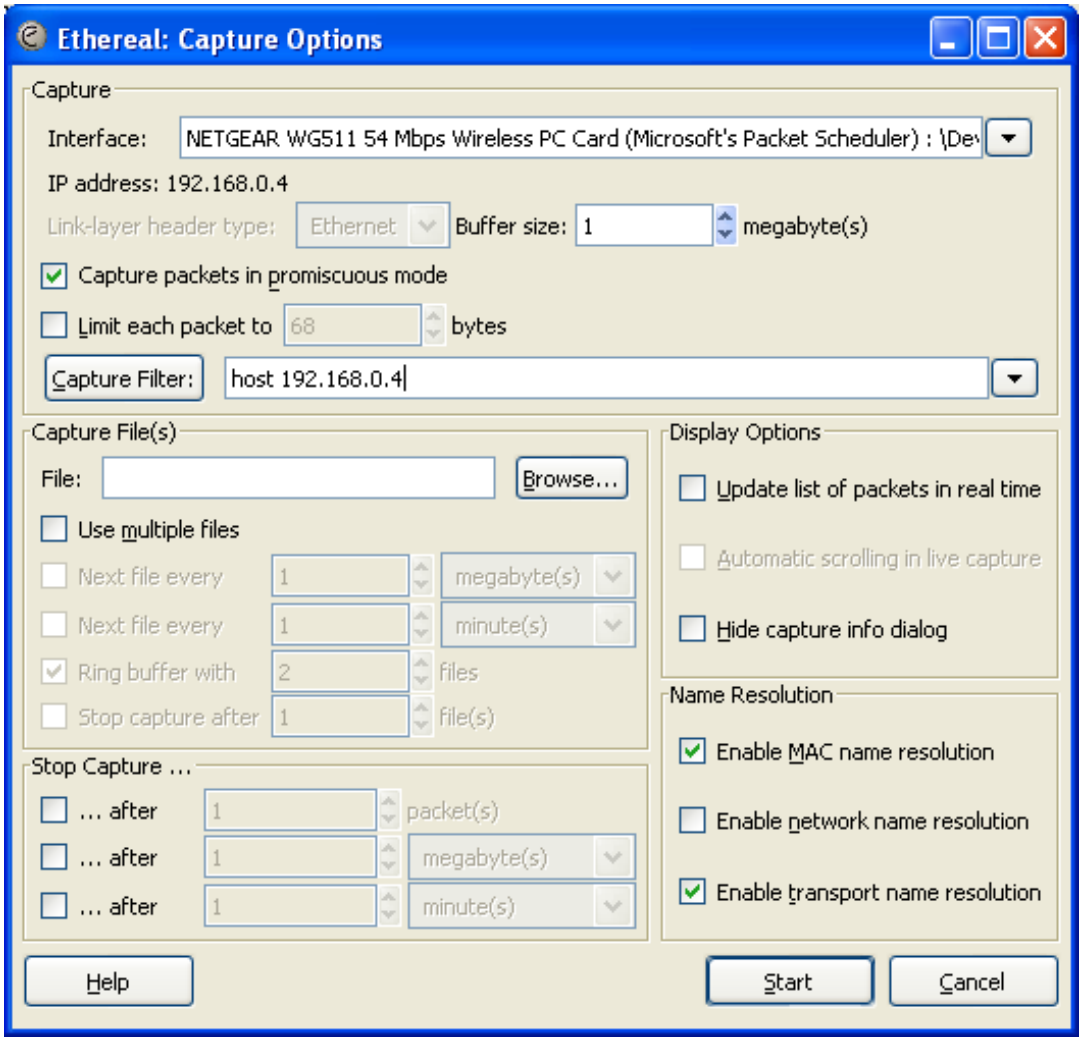
Using Wireshark

- Wireshark (formerly called Ethereal) is a GUI network protocol analyzer.
- It can examine data from a live network or from a capture file on disk.
- You can interactively browse the capture data, viewing summary and detail information for each packet.
- Wireshark can assemble all the packets in a TCP conversation and show you the ASCII (or EBCDIC, or hex) data in that conversation.
- Display filters in Wireshark are very powerful; more fields are filterable than in other protocol analyzers.
  Features: (see http://www.wireshark.org for a complete list of features and video tutorial)
- For other video guides and examples see BB resources folder

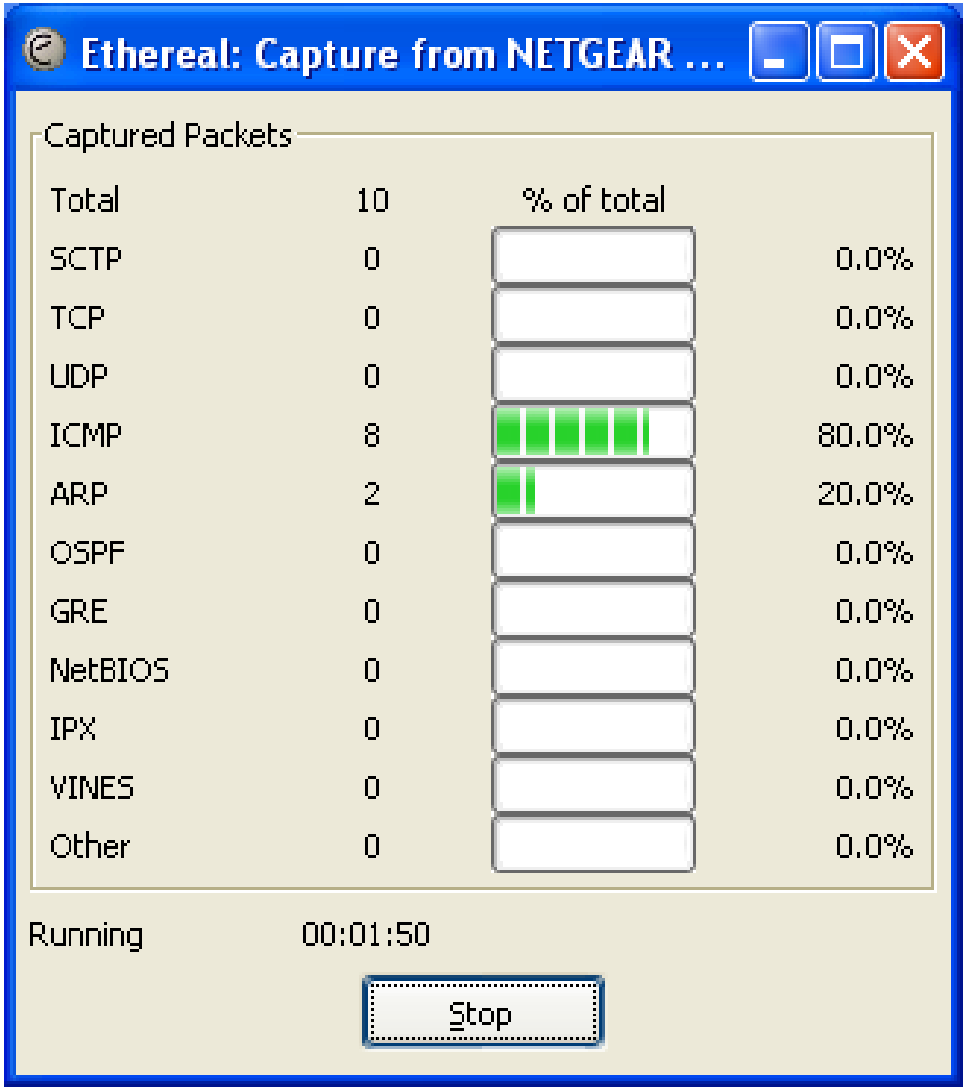
Capture Screen



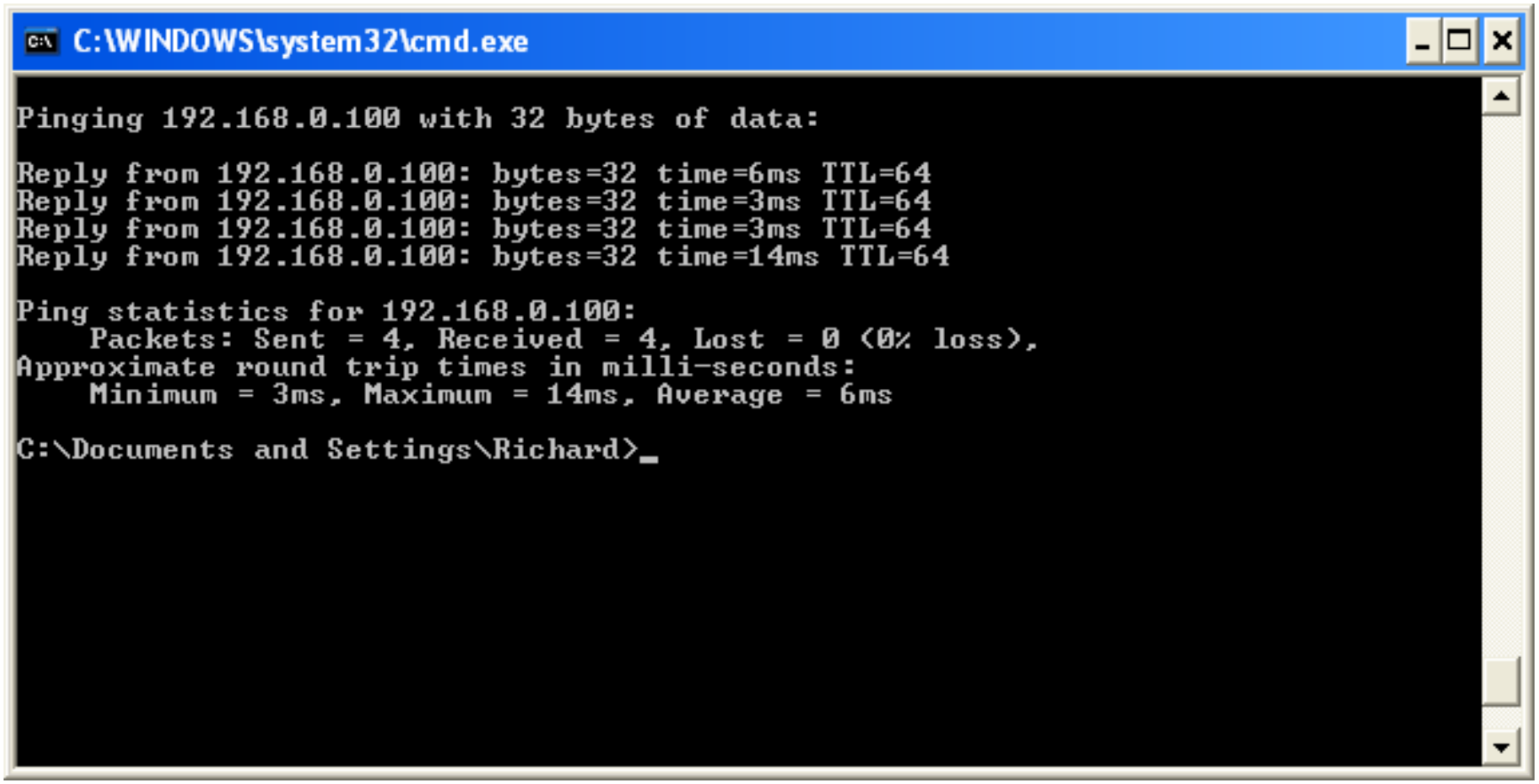
Capture Filter



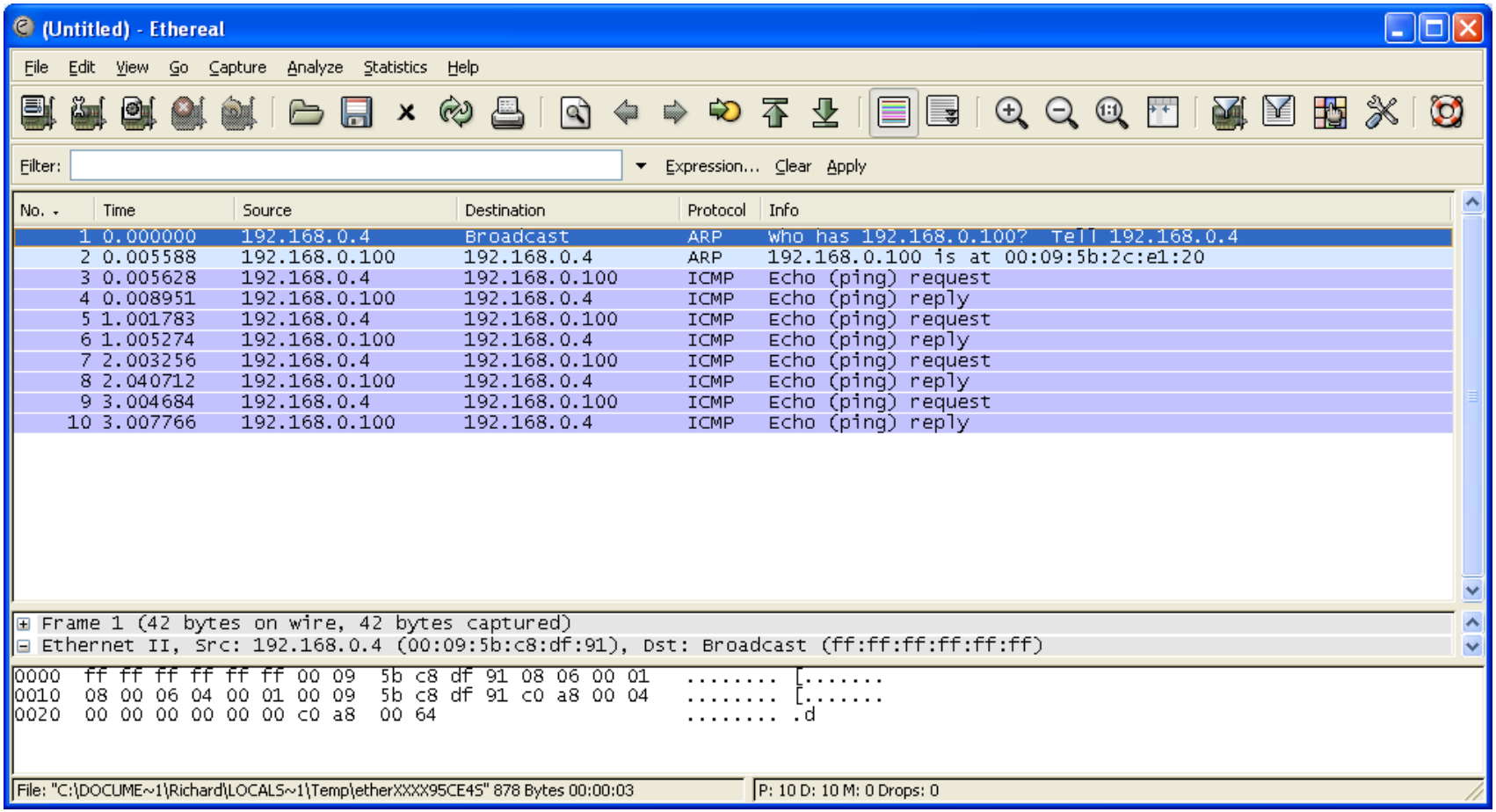
Protocols Screen



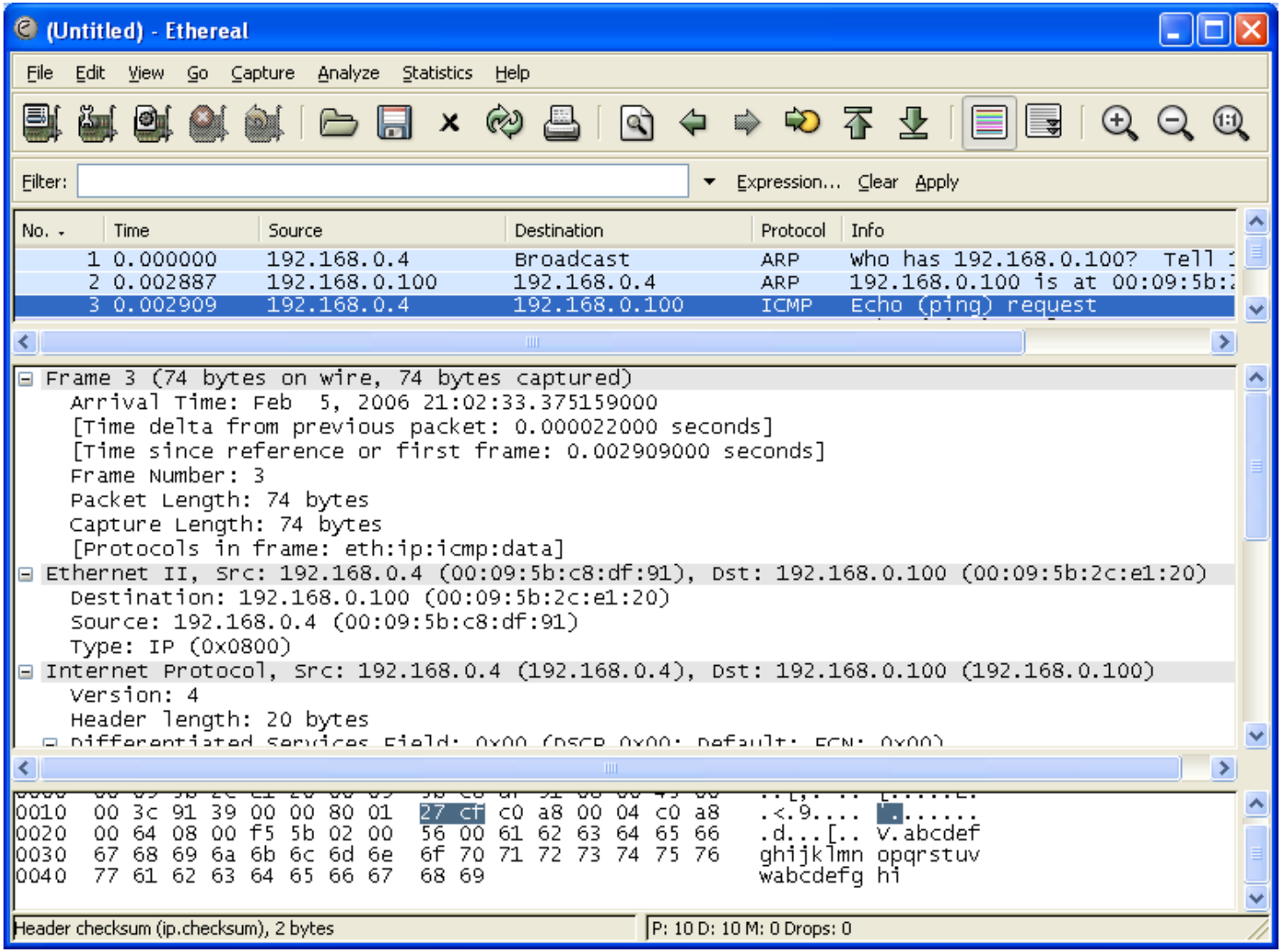
Command Prompt



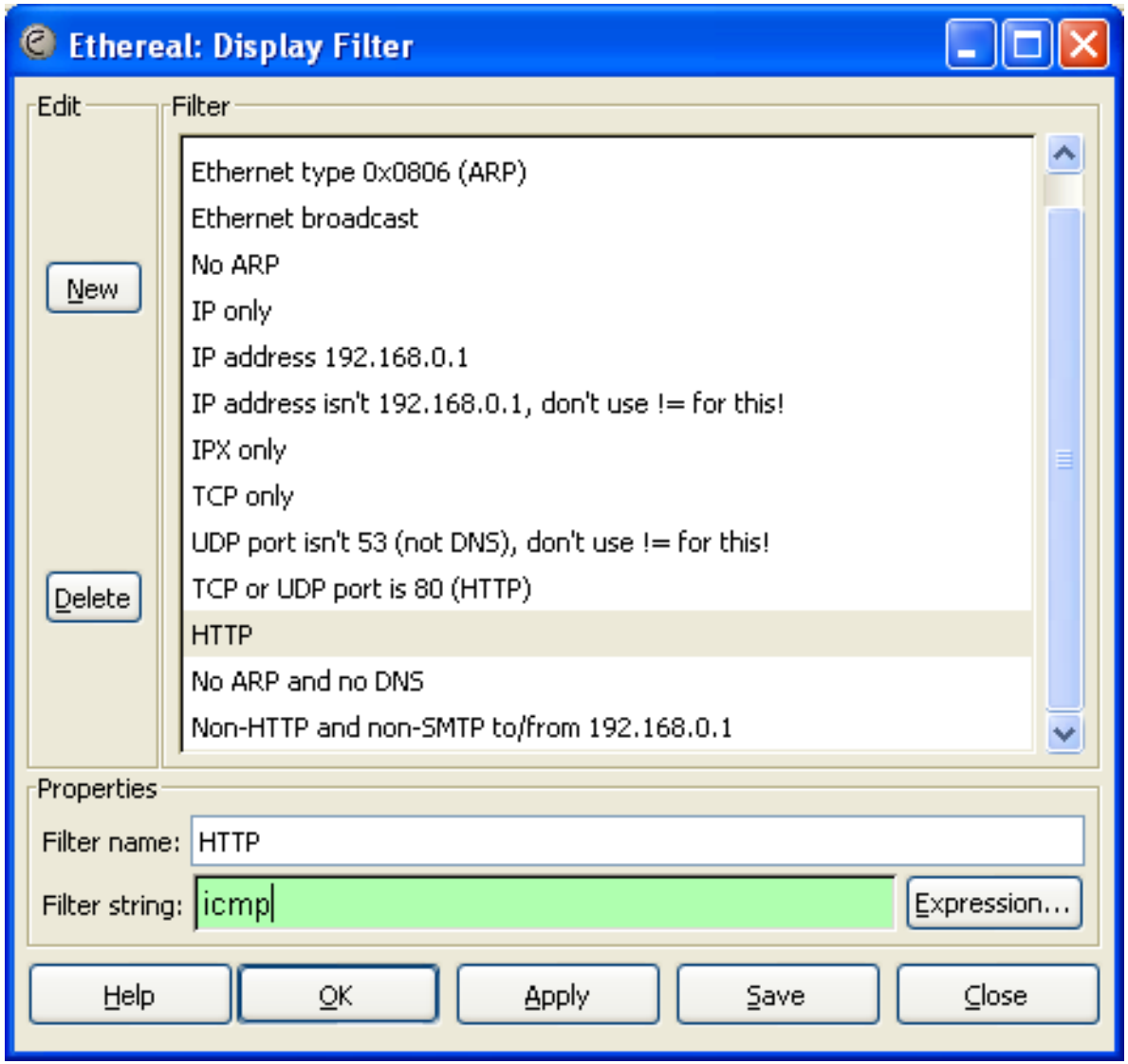
Capturing a Ping Trace



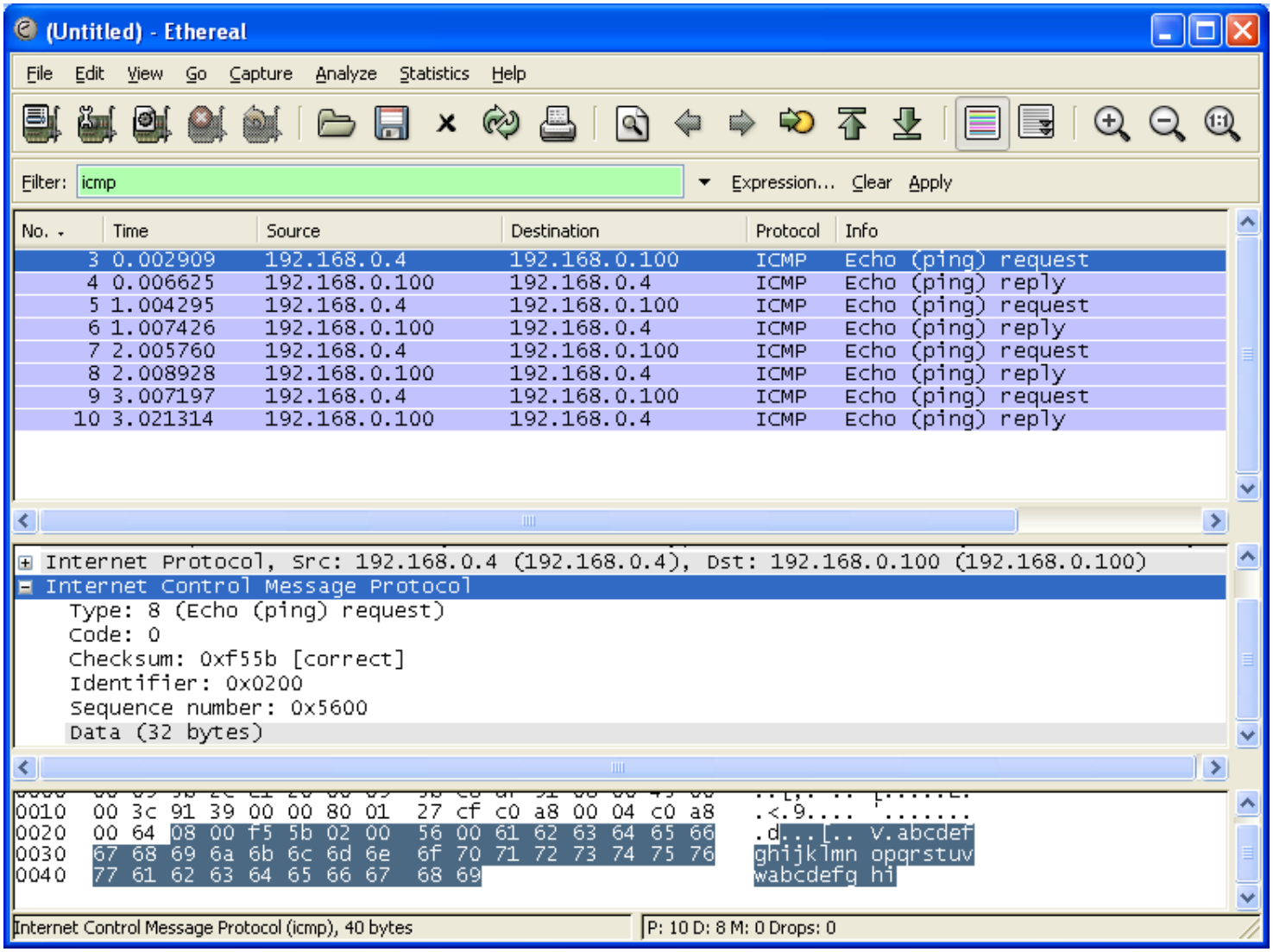
Expand Details



Apply ICMP Display Filter



Display Filter Result



Download Wireshark

- Download the latest version of Wireshark for free from http://www.wireshark.org. and install it on your home PC if you have one and read the documentation
- Then you can have a go at the following Exercises.
- Documentation is available 'on line' or for download as a PDF file
- Also read Ch4 and Ch5 of the ‘Wireshark and Ethereal Network Analyser Toolkit’ book.

Exercise 1

- Wireshark Filters

- You can learn a lot about network protocols by decoding and analysing packet captures. You must first, however, learn to filter out the traffic that you are not interested in.
- Create filters that will only show the packets indicated in both capture and display formats:
  1. Ethernet broadcasts destination
  2. Source MAC address 00-40-96-AD-28-6E
  3. IP address 192.168.1.1
  4. IP source address 192.168.1.1 and IP destination address 192.168.2.2
  5. Web traffic using TCP port 80
  6. Ping echo response packets

Exercise 2

- A simple Wireshark Flowgraph

- Download the ICMPtest.cap file from your Blackboard site (Support Resources - Wireshark - capfiles) and open it in Wireshark.
- Filter for ICMP packets then format the results to show the packet interchange using the Flowgraph feature found under the Statistics tab of Wireshark.

Exercise 3

- Analyse TCP flow to determine Telnet password
- Download the telnet.cap file from your Blackboard site and open it in Wireshark.
- Filter for Telnet packets then format the results to show the packet interchange using the Follow TCP feature found under the Analyze tab of Wireshark.
- Notice the echoplexing used by the Telnet protocol and locate the password

Exercise 4

- Analyse a VoIP voice conversation with Wireshark

- Download the VoIPtest.cap file from your Blackboard site and open it in Wireshark.
- Filter for SIP and display the Flowgraph under statics tab.
- Use the VoIP Calls analysis tool found under the statistics tab or telephony tab on later releases to:
  I. display the Flowgraph of the overall voice call
  II. replay the conversation in each direction using the player facility.

Exercise 5

- Observer Download

- Download the latest version (17) and use it in demo mode to investigate the operation and features it offers.
- Compare the protocol analyser features of Observer with Wireshark

Study Resources

- Network Monitoring Tools https://www.youtube.com/watch?v=bUGuwirOCC0
- Using Protocol Analyzers https://www.youtube.com/watch?v=UQ1NKip5tcc
- Interface Monitoring https://www.youtube.com/watch?v=n2aJ9WgZRvU