THE USE OF FREEWARE NETWORK ANALYZERS IN A NETWORKING LABORATORY

Ece Yaprak
Division of Engineering Technology
Wayne State University
Detroit, Michigan 48202
313-577-8075
yaprak@eng.wayne.edu

Lisa Anneberg
Electrical and Computer Engineering
Lawrence Technological University
Southfield, Michigan 48075
248-204-2539
anneberg@ltu.edu

ABSTRACT

Most undergraduate networking classes are taught using either purchased hardware components or simulation programs. However, at Wayne State University (WSU) and Lawrence Technological University (LTU), free-ware networking programs are used to complement the laboratory material. This paper describes the application of new technologies into our networking curricula by implementing freeware-networking programs, each with a different purpose and capability. The use of these hands-on labs in addition to using more traditional laboratories gives our students an edge on the market. In addition, our lab assignments can evolve in time to meet the new technology requirements and capabilities with a minimum amount of preparation time and cost.

INTRODUCTION

Current technological advances and significant developments in the computer networking industry are positively influencing our society. These developments have transformed our way of life in many ways. We have become much more technology dependent, for example, and have developed an appetite for global information. To prepare our students for this kind of environment, so that they can compete effectively, we need to keep up with the new technology in academe. Recognizing this need, we have incorporated freely available networking tools on the Internet to our undergraduate computer networking curricula in addition to using our traditional laboratory materials, when we teach in a laboratory environment today [1-4].

Traditionally, we have used simulation laboratories (using either OPNET by Mil3, or COMNET by Compuware) in addition to using off-the-shelf network equipment such as routers, bridges, and network cards. These laboratory settings gave our students an insight into how to design computer networks and run different scenarios under different traffic conditions and topologies. Some of these packages (such as MIL 3's IT Decision Guru) let students tailor a simulation to reflect a real network by profiling the exact behavior of an application by capturing packet traces, adding background traffic levels and then investigating "what-if" scenarios.

However, we wanted to use additional laboratory materials that enable our students to capture packets off the "live" network and analyze them. Since these kinds of equipment are very expensive, we have started experimenting with free network analyzer programs that we can currently download from the Internet instead of buying the actual hardware/software. These freeware networking programs have different purposes and capabilities. For example, through using some of these programs, such as CommView, Net Probe, Etherpeek, IP Calculator, and Advanced IP, our students have been able to capture and analyze network packets passing through our dial-up connection, and examine these packets to get a better understanding of network protocols.

FREE NETWORK ANALYZER PROGRAMS

In this section, we would like to describe some of the free analyzer programs that we have experimented with in our computer networking classes.

CommView (Fig.1), for example, monitors Internet or Local Area Network (LAN) activity by capturing and analyzing network packets. CommView allows our students to see the list of network connections passing through our dial-up connections and vital IP statistics, and examine individual packets. These packets are decoded down to the lowest layer with full analysis of the most widespread protocols such as Ipv4, NetBIOS, and TCP. Even though some shareware/evaluation programs only display half of the packets, however, this is enough to give our students insight into networking protocols and its layers [5].

Another example of a network analyzer with a free demo version is **Net Probe** [6], which decodes the following protocols: TCP/IP, IPC/SPX, NetBEUI, AppleTalk, Bridge/Switch Protocols, etc. This free program monitors the overall condition of the network, and finds various problems with the network including network slowdowns. Using this program, students monitor network activity in real-time, stress-test new networks and debug difficult situations using Net Probe's packet generator and capture capability. In addition, the use of programmable alarms which sound when user-defined situations occur alleviates the need for constant monitoring making for a very interesting lab experience!

Another interesting network analyzer we have used in our computer networking lab is **Etherpeek** [7], an Ethernet network traffic and protocol analyzer designed for troubleshooting and debugging mixed-platform, multi-protocol networks. It has diagnostic and analysis capabilities, protocol, and summary statistic over time, and IP subnet filtering using CIDR notation. <u>These capabilities normally far exceed the students' initial introduction to the topic of network analysis</u>.

Another program, the **IP Subnet Calculator** (Fig.2), allows students to predict the one-way and round-trip delay of packets over multiple LANs, display information about IP addresses including the range of IP addresses in a given subnet, individual host numbers, recommended subnet masks, and other information. The packet sizes, network topology and transmission speed, and client/server processing time are all user configurable.

Another demo program, **Advanced IP**, supports forward and reverse DNS resolution, response time, address translation (Hexadecimal and Binary) and Address Class information. It calculates subnets based on Subnet Mask, Mask Bits, Host Bits, and the Number of Subnets and Hosts per Subnet. It then generates a range of subnets based on these settings and generates a report of IP addresses for any subnet. This set of Internet calculation tools is another invaluable lab, one that is state-of-the-art, and one that can keep up to date with a minimum amount of analysis, preparation time, and money.

Ethereal is another free network protocol analyzer for Unix and Windows. It allows students to examine data from a live network or from a capture file on disk. Our students interactively browse the capture data, viewing summary and detail information for each packet [8]. This free network protocol analyzer allows a student to inspect the ASCII contents of a TCP data. This can be invaluable for tracking down HTTP, SMTP, and POP server problems. Data can be captured "off the wire" from a live network connection, or read from a capture file.

Using Ethereal, live data can be read from Ethernet, FDDI, PPP, token-ring, X.25, or Classical IP over ATM interfaces. Ethereal can also read capture files from other network analyzers such as (Sniffer™ Pro, Microsoft's Network Monitor, Novell's LANalyzer, and Wildpacket's Etherpeek). It can also read traces made from Lucent/Ascend WAN routers and Toshiba ISDN routers. Students can inspect the captured data while the capture is still in progress (Fig. 3).

Sniffer is yet another free protocol analyzer program, which is designed for interception and analysis of the packets going through the network. Using the packet driver, it requests all the packets from the network card driver [9].

IMPLEMENTATION

These freeware programs may require for some packages/libraries/patches to be installed in order to compile and run, and these required programs are also free! Another beauty is that these programs also come with instructions and sites to install these libraries/patches. All you need to do is just click on these and follow the instructions to download them. Some programs, like Ethereal, also come with a source code. Ethereal is the program that we have recently been using mainly because the source code is freely available and anyone who uses can make modifications to it. The idea behind it is to send contributions (such as new protocols, either as modules, or built into the source) back to the Ethereal for others to share.

Another advantage is that these programs are available for students anywhere: at their home or work. The students don't need to physically be in the lab in order to use these programs. This is not the case when other professional programs are used because of licensing issues unless they get a free student version.

One drawback in using such freeware programs is that some of them may expire within a time limit, such as a month. However, one can download it again if needed.

Fig. 4 shows the connection of two computers using a crossover Ethernet cable. Both computers use Netmeeting as a dial-up connection and start chatting. A Sniffer program, which resides on one of the computers captures this information and decodes it. Fig. 5 shows this screen capture. The students are very excited to see their live connection captured on the screen. They are then asked to decode the information given by the Sniffer program and justify that it is correct. This tops their imagination and makes learning fun.

We ask our students to download, install, implement, and compare three different computer network packet analysis programs. The 'deliverable' for each team/group of students is a one-page memo aimed at a fictitious boss. The 'boss' is informed of the analyses of each package, limitations and strengths of each package, and must give 'hard target' examples to the boss. This type of student lab write-up is relatively easy to grade because it gets to the heart of the lab topic, simulates 'real-world' engineering tasks, and helps the students to really understand the topic.

For example, a typical memo from a student contained the following information:

a. 'Sniffer' Ufasoft records all packets regardless of the intended computer on the network, supports a wide variety of packet types (UDP, TCP, IP, and general Ethernet). It performs basic analysis only, picking apart the hex data to find source and destination addresses. However, this useful information allows network administrators to examine the flow of information for faulty addresses, duplicate addresses, and general bottleneck areas.

b. 'WinHexCom' Win 32-protocol analyzer manipulates a string of hexadecimal numbers to conform to a packet format. For instance, the phrase 'hello' is converted to 7E 7D 2E 06 19 7E when creating a PPP frame. The results of clicking 'send' sends the Hex figure to the transmit window. Simply typing in ASCII without adding a frame sends the code 0E that must translate as 'error'. This program allows programmers to verify their frame analyzers or packet creators are assembling a message properly, by giving the output a base to be compared with.

c. Advanced Log Analyzer performs basic analysis on web server log files, and prepares a web page of table/graph combinations with several types of statistics. Some of these statistics include: most common referrer (what web page brought your visitors to your site), most common browser (what web browser your visitors used), most common platform (what operating system your visitors used), most requested pages (which parts of your sites are viewed most often), hits by day/week (the number of times your site has had a page viewed). This program can be useful in helping to improve your site.

CONCLUSIONS/RECOMMENDATIONS

The use of trial-ware versions of network analyzers gives instructors another tool to help them teach. They are not too hard to install and not too hard to run, and they help students learn. When students actually capture packets from a live network, they are able to decode a packet or use a filter to collect only the data of interest. We have given an overview of some of these packages. The main drawback in using such programs is that some of them expire within a time limit and some may not have full capabilities. However, we have found out that even with limited capabilities, students gain invaluable experience.

Our students get additional experience in teamwork, oral and written communication, and real-world budget-limiting problems; which all address the a-k ABET criteria 3.

BIBLIOGRAPHY

1. Tannenbaum, "*Computer Networks,*" Prentice Hall, Inc., 1996.

2. Douglas Comer, "*Computer Networks And Internets*," Prentice Hall, 2001.

3. James F. Kurose and Keith W. Ross, *"Computer Networking: A Top-Down Approach Featuring the Internet,"* Addison-Wesley, Inc., 2000.

4. William Stallings, "Data and Computer Communications," Prentice Hall, 2000.

5. http://www.webattack.com/get/commview.shtml

6. http://www.netplusinc.com/products.html

7. www.zdnet.com

8. http://www.ethereal.com/

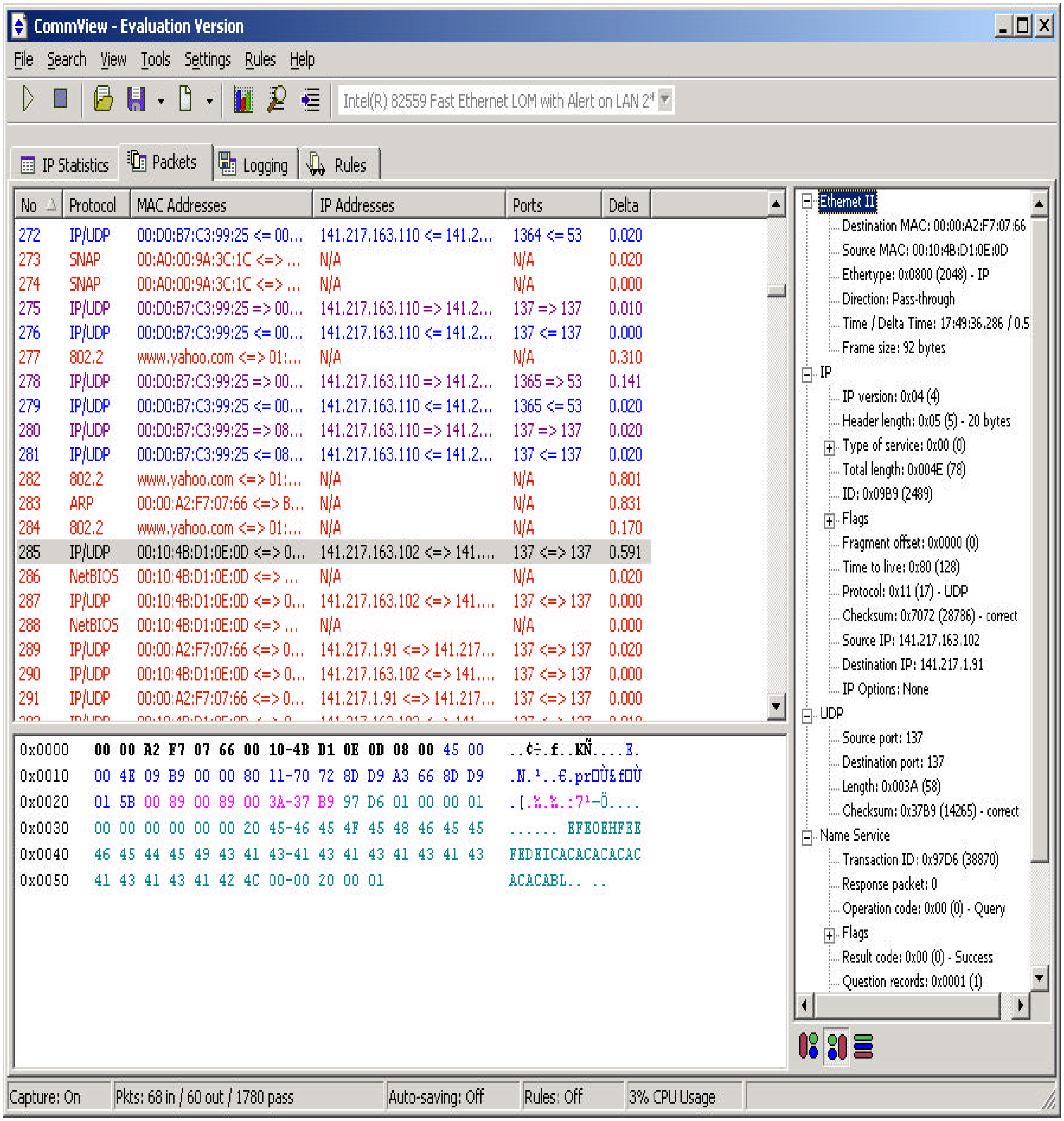
9. http://www.ufasoft.com/sniffer/

10. http://www.sstinc.com/home.html

BIOGRAPHY OF AUTHORS

Ece Yaprak: Ece Yaprak received her Ph.D. in Computer Engineering from Wayne State University in 1989. Prior to joining WSU's Division of Engineering Technology in 1993, she taught at Western Michigan University, and held technical positions at General Electric, Ford Motor Company, NASA (Lewis, Jet Propulsion Laboratory, and Ames Research Center) and Navy (SPAWAR). Her areas of interest include computer networks and communications where she has published papers and received funding from NSF and other organizations for her scholarly work. She has received excellence in teaching awards from ET Division and the College of Engineering.

Lisa Anneberg: Lisa Anneberg received her Ph.D. in Computer Engineering from Wayne State University in 1991. She has been an associate professor at Lawrence Technological University since 1990. Before joining LTU, she worked at Wayne State University, General Motors, and Schoolcraft College. Her areas of interest include software engineering and computer networking. She has published a number of papers in these areas and continues to do educational consulting on a regular basis.


CommView - Evaluation Version
File Search View Tools Settings Rules Help
Intel(R) 82559 Fast Ethernet LOM with Alert on LAN 2*
IP Statistics
Packets
Logging
Rules

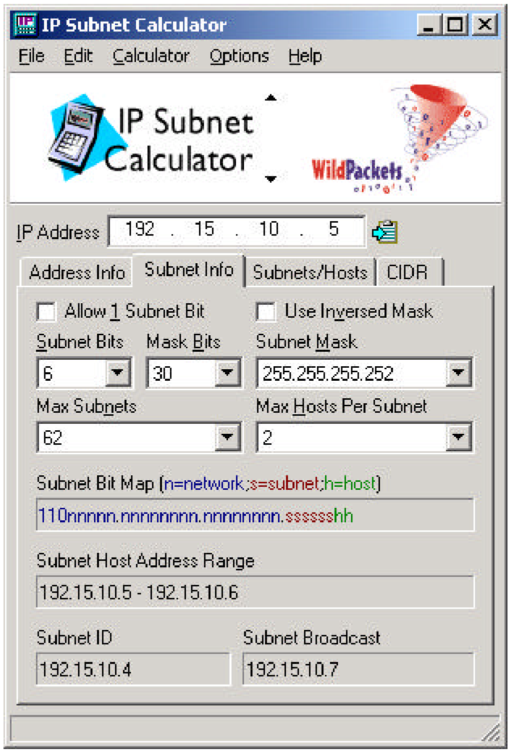
No	Protocol	MAC Addresses	IP Addresses	Ports	Delta
272	IP/UDP	00:D0:B7:C3:99:25 <= 00...	141.217.163.110 <= 141.2...	1364 <= 53	0.020
273	SNAP	00:A0:00:9A:3C:1C <=> ...	N/A	N/A	0.020
274	SNAP	00:A0:00:9A:3C:1C <=> ...	N/A	N/A	0.000
275	IP/UDP	00:D0:B7:C3:99:25 => 00...	141.217.163.110 => 141.2...	137 => 137	0.010
276	IP/UDP	00:D0:B7:C3:99:25 <= 00...	141.217.163.110 <= 141.2...	137 <= 137	0.000
277	802.2	www.yahoo.com <=> 01:...	N/A	N/A	0.310
278	IP/UDP	00:D0:B7:C3:99:25 => 00...	141.217.163.110 => 141.2...	1365 => 53	0.141
279	IP/UDP	00:D0:B7:C3:99:25 <= 00...	141.217.163.110 <= 141.2...	1365 <= 53	0.020
280	IP/UDP	00:D0:B7:C3:99:25 => 08...	141.217.163.110 => 141.2...	137 => 137	0.020
281	IP/UDP	00:D0:B7:C3:99:25 <= 08...	141.217.163.110 <= 141.2...	137 <= 137	0.020
282	802.2	www.yahoo.com <=> 01:...	N/A	N/A	0.801
283	ARP	00:00:A2:F7:07:66 <=> B...	N/A	N/A	0.831
284	802.2	www.yahoo.com <=> 01:...	N/A	N/A	0.170
285	IP/UDP	00:10:4B:D1:0E:0D <=> 0...	141.217.163.102 <=> 141....	137 <=> 137	0.591
286	NetBIOS	00:10:4B:D1:0E:0D <=> ...	N/A	N/A	0.020
287	IP/UDP	00:10:4B:D1:0E:0D <=> 0...	141.217.163.102 <=> 141....	137 <=> 137	0.000
288	NetBIOS	00:10:4B:D1:0E:0D <=> ...	N/A	N/A	0.000
289	IP/UDP	00:00:A2:F7:07:66 <=> 0...	141.217.1.91 <=> 141.217...	137 <=> 137	0.020
290	IP/UDP	00:10:4B:D1:0E:0D <=> 0...	141.217.163.102 <=> 141....	137 <=> 137	0.000
291	IP/UDP	00:00:A2:F7:07:66 <=> 0...	141.217.1.91 <=> 141.217...	137 <=> 137	0.000

```
0x0000   00 00 A2 F7 07 66 00 10-4B D1 0E 0D 08 00 45 00   ..¢÷.f..KÑ....E.
0x0010   00 4E 09 B9 00 00 80 11-70 72 8D D9 A3 66 8D D9   .N.¹..€.pr□Ù£f□Ù
0x0020   01 5B 00 89 00 89 00 3A-37 B9 97 D6 01 00 00 01   .[.‰.‰.:7¹—Ö....
0x0030   00 00 00 00 00 00 20 45-46 45 4F 45 48 46 45 45   ...... EFEOEHFEE
0x0040   46 45 44 45 49 43 41 43-41 43 41 43 41 43 41 43   FEDEICACACACACAC
0x0050   41 43 41 43 41 42 4C 00-00 20 00 01               ACACABL.. ..
```

```
⊟ Ethernet II
    Destination MAC: 00:00:A2:F7:07:66
    Source MAC: 00:10:4B:D1:0E:0D
    Ethertype: 0x0800 (2048) - IP
    Direction: Pass-through
    Time / Delta Time: 17:49:36.286 / 0.5
    Frame size: 92 bytes
⊟ IP
    IP version: 0x04 (4)
    Header length: 0x05 (5) - 20 bytes
  ⊞ Type of service: 0x00 (0)
    Total length: 0x004E (78)
    ID: 0x09B9 (2489)
  ⊞ Flags
    Fragment offset: 0x0000 (0)
    Time to live: 0x80 (128)
    Protocol: 0x11 (17) - UDP
    Checksum: 0x7072 (28786) - correct
    Source IP: 141.217.163.102
    Destination IP: 141.217.1.91
    IP Options: None
⊟ UDP
    Source port: 137
    Destination port: 137
    Length: 0x003A (58)
    Checksum: 0x37B9 (14265) - correct
⊟ Name Service
    Transaction ID: 0x97D6 (38870)
    Response packet: 0
    Operation code: 0x00 (0) - Query
  ⊞ Flags
    Result code: 0x00 (0) - Success
    Question records: 0x0001 (1)
```

Capture: On | Pkts: 68 in / 60 out / 1780 pass | Auto-saving: Off | Rules: Off | 3% CPU Usage

Fig.1: CommView – Capturing and Decoding Networking Traffic



IP Address 192 . 15 . 10 . 5

□ Allow 1 Subnet Bit □ Use Inversed Mask

Subnet Bits	Mask Bits	Subnet Mask
6	30	255.255.255.252

Max Subnets	Max Hosts Per Subnet
62	2

Subnet Bit Map (n=network;s=subnet;h=host)

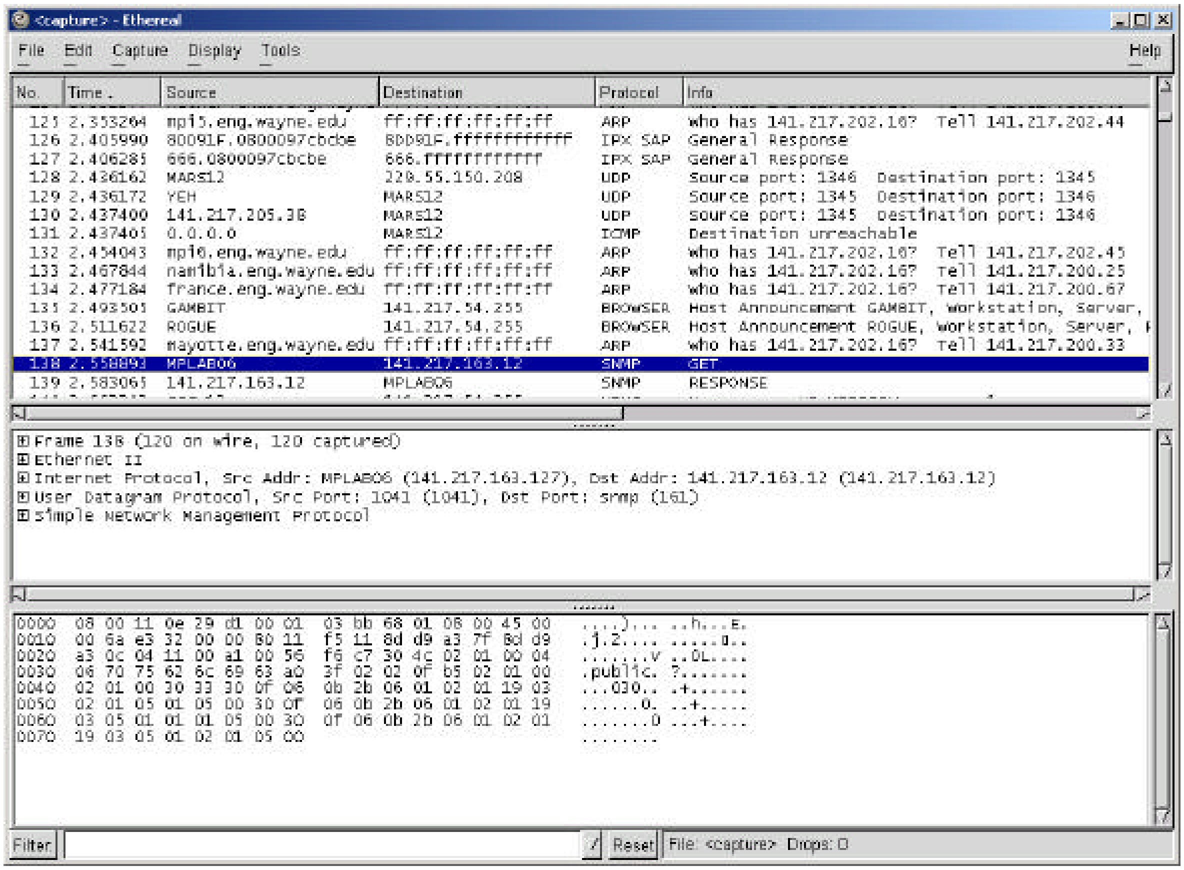
110nnnnn.nnnnnnnn.nnnnnnnn.sssssshh

Subnet Host Address Range

192.15.10.5 - 192.15.10.6

Subnet ID	Subnet Broadcast
192.15.10.4	192.15.10.7

Fig. 2: IP Subnet Calculator



```
No.  Time .    Source                 Destination        Protocol  Info
125 2.353264 mpi5.eng.wayne.edu      ff:ff:ff:ff:ff:ff  ARP       who has 141.217.202.167  Tell 141.217.202.44
126 2.405990 80091F.0800097cbcbe     80091F.ffffffffffff IPX SAP  General Response
127 2.406285 666.0800097cbcbe        666.ffffffffffff   IPX SAP   General Response
128 2.436162 MARS12                  220.55.150.208     UDP       Source port: 1346  Destination port: 1345
129 2.436172 YEH                     MARS12             UDP       Source port: 1345  Destination port: 1346
130 2.437400 141.217.205.38          MARS12             UDP       Source port: 1345  Destination port: 1346
131 2.437405 0.0.0.0                 MARS12             ICMP      Destination unreachable
132 2.454043 mpi6.eng.wayne.edu      ff:ff:ff:ff:ff:ff  ARP       who has 141.217.202.167  Tell 141.217.202.45
133 2.467844 namibia.eng.wayne.edu   ff:ff:ff:ff:ff:ff  ARP       who has 141.217.202.167  Tell 141.217.200.25
134 2.477184 france.eng.wayne.edu    ff:ff:ff:ff:ff:ff  ARP       who has 141.217.202.167  Tell 141.217.200.67
135 2.493505 GAMBIT                  141.217.54.255     BROWSER   Host Announcement GAMBIT, Workstation, Server,
136 2.511622 ROGUE                   141.217.54.255     BROWSER   Host Announcement ROGUE, Workstation, Server, 
137 2.541592 mayotte.eng.wayne.edu   ff:ff:ff:ff:ff:ff  ARP       who has 141.217.202.167  Tell 141.217.200.33
138 2.558893 MPLAB06                 141.217.163.12     SNMP      GET
139 2.583065 141.217.163.12          MPLAB06            SNMP      RESPONSE
```

```
⊞ Frame 138 (120 on wire, 120 captured)
⊞ Ethernet II
⊞ Internet Protocol, Src Addr: MPLAB06 (141.217.163.127), Dst Addr: 141.217.163.12 (141.217.163.12)
⊞ User Datagram Protocol, Src Port: 1041 (1041), Dst Port: snmp (161)
⊞ Simple Network Management Protocol
```

```
0000  08 00 11 0e 29 d1 00 01  03 bb 68 01 08 00 45 00   ....)... ..h...E.
0010  00 6a e3 32 00 00 80 11  f5 11 8d d9 a3 7f 8d d9   .j.2.... ......0..
0020  a3 0c 04 11 00 a1 00 56  f6 c7 30 4c 02 01 00 04   .......V ..0L....
0030  06 70 75 62 6c 69 63 a0  3f 02 02 0f b5 02 01 00   .public. ?.......
0040  02 01 00 30 33 30 0f 06  0b 2b 06 01 02 01 19 03   ...030.. .+......
0050  02 01 05 01 05 00 30 0f  06 0b 2b 06 01 02 01 19   ......0. ..+.....
0060  03 05 01 01 01 05 00 30  0f 06 0b 2b 06 01 02 01   .......0 ...+....
0070  19 03 05 01 02 01 05 00                            ........
```

File: <capture> Drops: 0

Fig.3: Ethereal – Packet capture

"Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition Copyright @ 2002, American Society for Engineering Education"

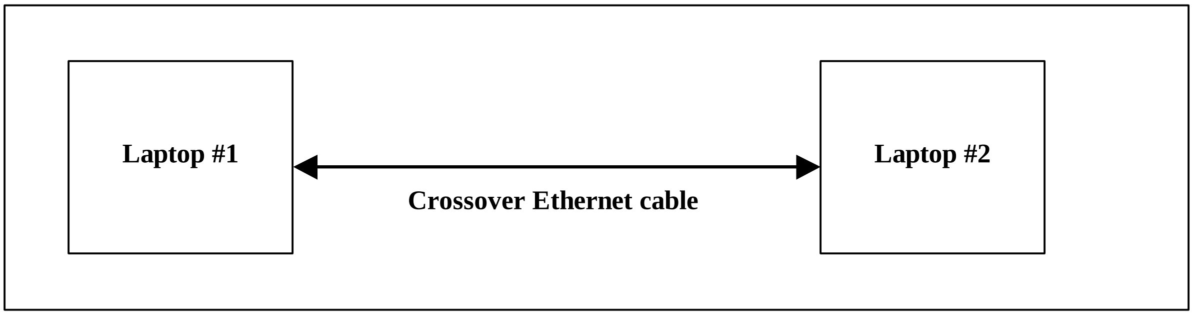
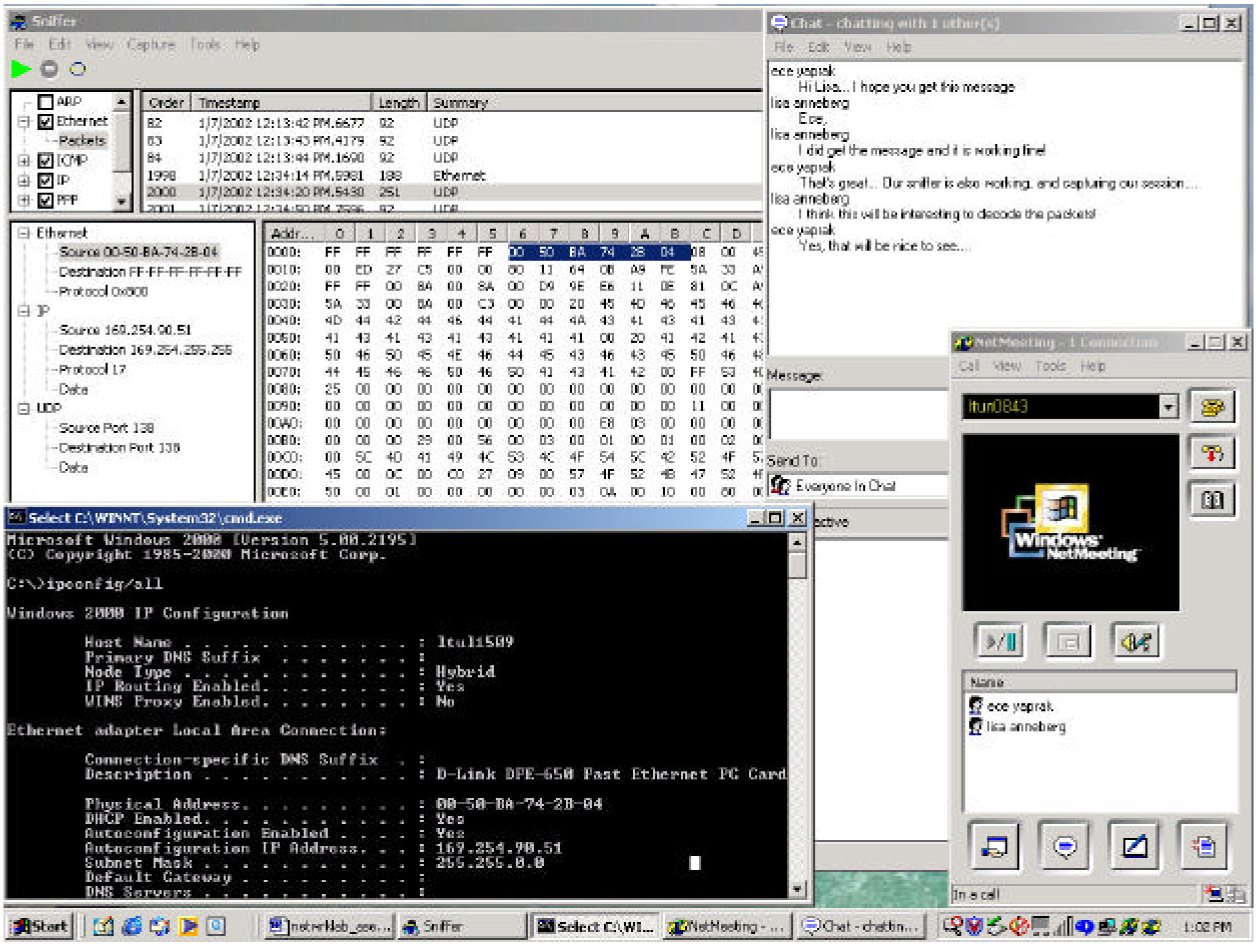


Fig.4: Capturing packets between two computers



Order	Timestamp	Length	Summary
82	1/7/2002 12:13:42 PM.6677	92	UDP
83	1/7/2002 12:13:43 PM.4179	92	UDP
84	1/7/2002 12:13:44 PM.1690	92	UDP
1998	1/7/2002 12:34:14 PM.5981	188	Ethernet
2000	1/7/2002 12:34:20 PM.5430	251	UDP
2001	1/7/2002 12:34:50 PM.7596	92	UDP

```
Ethernet
    Source 00-50-BA-74-2B-04
    Destination FF-FF-FF-FF-FF-FF
    Protocol 0x800
IP
    Source 169.254.90.51
    Destination 169.254.255.255
    Protocol 17
    Data
UDP
    Source Port 138
    Destination Port 138
    Data
```

```
Addr...  0  1  2  3  4  5  6  7  8  9  A  B  C  D
0000:   FF FF FF FF FF FF 00 50 BA 74 2B 04 08 00
0010:   00 ED 27 C5 00 00 80 11 64 0B A9 FE 5A 33
0020:   FF FF 00 8A 00 8A 00 D9 9E E6 11 0E 81 0C
0030:   5A 33 00 8A 00 C3 00 00 20 45 4D 46 45 46
0040:   4D 44 42 44 46 44 41 44 4A 43 41 43 41 43
0050:   41 43 41 43 41 43 41 41 41 00 20 41 42 41
0060:   50 46 50 45 4E 46 44 45 43 46 43 45 50 46
0070:   44 45 46 46 50 46 50 41 43 41 42 00 FF 53
0080:   25 00 00 00 00 00 00 00 00 00 00 00 00 00
0090:   00 00 00 00 00 00 00 00 00 00 00 00 11 00
00A0:   00 00 00 00 00 00 00 00 00 E8 03 00 00 00
00B0:   00 00 00 29 00 56 00 03 00 01 00 01 00 02
00C0:   00 5C 40 41 49 4C 53 4C 4F 54 5C 42 52 4F
00D0:   45 00 0C 00 C0 27 09 00 57 4F 52 4B 47 52
00E0:   50 00 01 00 00 00 00 00 03 0A 00 10 00 80
```

ece yaprak
Hi Lisa... I hope you get this message
lisa anneberg
Ece,
lisa anneberg
I did get the message and it is working fine!
ece yaprak
That's great... Our sniffer is also working, and capturing our session....
lisa anneberg
I think this will be interesting to decode the packets!
ece yaprak
Yes, that will be nice to see....

Select C:\WINNT\System32\cmd.exe

```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\>ipconfig/all

Windows 2000 IP Configuration

        Host Name . . . . . . . . . . . . : ltul1509
        Primary DNS Suffix  . . . . . . . :
        Node Type . . . . . . . . . . . . : Hybrid
        IP Routing Enabled. . . . . . . . : Yes
        WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        Description . . . . . . . . . . . : D-Link DFE-650 Fast Ethernet PC Card

        Physical Address. . . . . . . . . : 00-50-BA-74-2B-04
        DHCP Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        Autoconfiguration IP Address. . . : 169.254.90.51
        Subnet Mask . . . . . . . . . . . : 255.255.0.0
        Default Gateway . . . . . . . . . :
        DNS Servers . . . . . . . . . . . :
```

Fig. 5: Sniffer Program capturing live connection