Ethereal

The Open Source Network Protocol Analyzer

Eric Raeburn
Hewlett-Packard

Jan 30, 2003

Agenda

- *Ethereal*: what is it?
- Overview of features
- Comparison to Microsoft *Network Monitor*
- Getting started: capture filters (*tcpdump* syntax)
- Isolating data: display filters (*ethereal* C-style syntax)
- Tracing for indefinite periods
- Conversion to and from other formats
- Installation and dependencies
- Resources, mailing lists
- Questions

What is it?

- Open Source Network Protocol Analyzer
- Released under GNU Public License (it's free)
- Runs on all flavors of Unix, Linux, Windows
- Prebuilt binaries and source code are available
- Original author: Gerald Combs
- Over 200 contributors, including members of Samba Team
- Defacto standard among open source community
- Website: www.ethereal.com

For the impatient… (What is it, part 2)

cifsclient-krb5-auth-ok-0.trace - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
1	0.000000	hpntc263.cup.hp.com	hpntc825.cup.hp.com	KRB5	AS-REQ
2	0.003225	hpntc825.cup.hp.com	hpntc263.cup.hp.com	KRB5	KRB-ERROR
3	1.725024	hpntc263.cup.hp.com	hpntc825.cup.hp.com	KRB5	AS-REQ
4	1.732516	hpntc825.cup.hp.com	hpntc263.cup.hp.com	KRB5	AS-REP
11	9.600269	hpntc263.cup.hp.com	hpntc723.cup.hp.com	NBSS	Session request, to HPNTC723<20> from HPNTC263<20>
12	9.600621	hpntc723.cup.hp.com	hpntc263.cup.hp.com	NBSS	Positive session response
13	9.604385	hpntc263.cup.hp.com	hpntc723.cup.hp.com	SMB	Negotiate Protocol Request
14	9.605424	hpntc723.cup.hp.com	hpntc263.cup.hp.com	SMB	Negotiate Protocol Response
15	9.644947	hpntc263.cup.hp.com	hpntc825.cup.hp.com	KRB5	TGS-REQ
16	9.652970	hpntc825.cup.hp.com	hpntc263.cup.hp.com	KRB5	TGS-REP
18	9.734999	hpntc263.cup.hp.com	hpntc723.cup.hp.com	SMB	Session Setup AndX Request
19	9.740916	hpntc723.cup.hp.com	hpntc263.cup.hp.com	SMB	Session Setup AndX Response[Unreassembled Packet]

```
⊞ Frame 4 (1421 bytes on wire, 1421 bytes captured)
⊞ Ethernet II, Src: 08:00:09:cb:99:8a, Dst: 00:10:83:03:9f:26
⊞ Internet Protocol, Src Addr: hpntc825.cup.hp.com (15.13.115.184), Dst Addr: hpntc263.cup.hp.com (15.13.114.212)
⊞ User Datagram Protocol, Src Port: kerberos5 (88), Dst Port: 53016 (53016)
⊟ Kerberos
    Version: 5
    MSG Type: AS-REP
  ⊟ Pre-Authentication
      Type: PA-PW-SALT
      Value: 524B57494E324B2D4E41544956452E43...
```

```
0000  00 10 83 03 9f 26 08 00  09 cb 99 8a 08 00 45 00   .....&.. ......E.
0010  05 7f 91 94 00 00 80 11  9f 33 0f 0d 73 b8 0f 0d   ........ .3..s...
0020  72 d4 00 58 cf 18 05 6b  1b a6 6b 82 05 5f 30 82   r..X...k ..k.._0.
0030  05 5b a0 03 02 01 05 a1  03 02 01 0b a2 2a 30 28   .[...... .....*0(
0040  30 26 a1 03 02 01 03 a2  1f 04 1d 52 4b 57 49 4e   0&...... ...RKWIN
0050  32 4b 2d 4e 41 54 49 56  45 2e 43 55 50 2e 48 50   2K-NATIV E.CUP.HP
0060  2e 43 4f 4d 65 72 69 63  a3 1b 1b 19 52 4b 57 49   .COMeric ....RKWI
0070  4e 32 4b 2d 4e 41 54 49  56 45 2e 43 55 50 2e 48   N2K-NATI VE.CUP.H
```

Filter: nbss || smb || kerberos Reset Apply File: cifsclient-krb5-auth-ok-0.trace

Features…

- Graphical user interface
- Rich syntax for capture and display filters
- Over 370 network protocols decoded, as of latest version; Ver. 0.9.9, released Jan. 23, 2002, includes GSS-API, NTLM, SPNEGO, Win2k security blobs
- Reads and writes capture files in many formats:
  - *libpcap* (tcpdump)
  - *Network Monitor* (Microsoft)
  - *LanAnalyzer* (Novell)
  - *Sniffer* and *NetXray* (Network Associates)
  - *…and several others*
  - *nettl* (HP-UX)
  - *iptrace* (AIX)
  - *snoop* (Sun)

...Features

- Interactive GUI facility for building display filters
- Distributions include text-based interface (*tethereal*) similar to *tcpdump*, programmatic capture-editor and converter (*editcap*), manpages for Unix and Linux (or via web for Windows)
- Analysis of live or saved network traces (packets can be examined while capture is active)
- Prints captures as plain text or postscript to file or printer
- Updated often (1 - 3 month intervals) with new protocol decodings or enhancements to existing decoders

Comparison to *Network Monitor*...

	Ethereal	*Network Monitor*
Free	yes	no
Updated often	yes	no
Windows installation	easy	easy
Linux installation	easy	not available
Initial HP-UX installation	*swinstall*[1]	not available
Unix updates	easy	not available
New decoder availability for various protocols	under continuous development	difficult-to-impossible to obtain

[1]see *Installation and Dependencies*

...Comparison to *Network Monitor*...

	Ethereal	*Network Monitor*
Supports complex display filters	yes	no
Can run multiple instances	yes	yes
Reads and writes formats of most other vendors' sniffers	yes	no
Number of protocols decoded	~370 and counting	78
To capture traffic between *host_A* and *host_B*	specify hostnames	manually add hostnames to database by ip or hardware address, then select

...Comparison to *Network Monitor*...

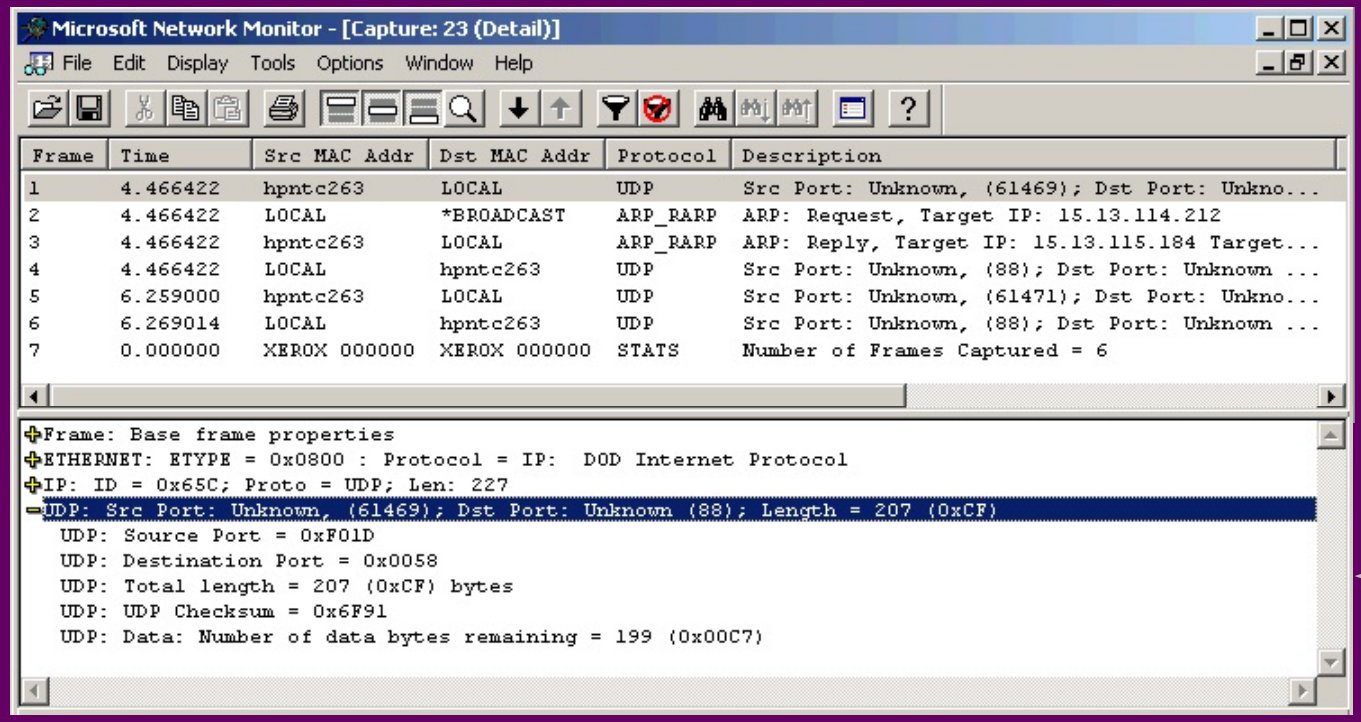
	Ethereal	*Network Monitor*
Decodes CAP_UNIX bit	yes	no
Decodes CIFS Unix Extensions	yes	no
Opens any number of packets each in its own window	yes	no
Allows filters to be saved	yes	yes
Supports fancy color configuration, by protocol	yes	yes
Features powerful GUI filter-expression builder	yes	no

...Comparison to *Network Monitor*...

Screenshot 1a: **`kinit`**(1) captured with *Network Monitor* 5.0 on Win2k, filter definition:

- *host_A* → *host_B*
- broadcast → *host_A*
- broadcast → *host_B*

KRB5_AS_REQ/REP packets not recognized; displayed only as encapsulated UDP data



...Comparison to *Network Monitor*...

Screenshot 1b: hex dump of KRB5_AS_REQ with *Network Monitor*

- non-printing characters represented by “smileys”
- 16-byte continuous rows
- lines spaced at 1.5

```
00000000  08 00 09 CB 99 8A 00 10 83 03 9F 26 08 00 45 00   ◘.○╦Öè.►â♥ƒ&◘.E.
00000010  00 E3 D5 85 00 00 40 11 9F DE 0F 0D 72 D4 0F 0D   .π╒à..@◄ƒ▄♫♪r╘♫♪
00000020  73 B8 DC EC 00 58 00 CF 70 1F 6A 81 C4 30 81 C1   s╕▄∞.X.╧p▼jü─0ü┴
00000030  A1 03 02 01 05 A2 03 02 01 0A A4 81 B4 30 81 B1   í♥☺♠ó♥☺◙ñü┤0ü▒
00000040  A0 07 03 05 00 00 00 00 00 A1 11 30 0F A0 03 02   á•♥♣.....í◄0♫á♥☻
00000050  01 01 A1 08 30 06 1B 04 65 72 69 63 A2 1B 1B 19   ☺☺í◘0♠←♦ericó←←↓
00000060  52 4B 57 49 4E 32 4B 2D 4E 41 54 49 56 45 2E 43   RKWIN2K-NATIVE.C
00000070  55 50 2E 48 50 2E 43 4F 4D A3 2E 30 2C A0 03 02   UP.HP.COMú.0,á♥☻
00000080  01 00 A1 25 30 23 1B 06 6B 72 62 74 67 74 1B 19   ☺.í%0#←♠krbtgt←↓
00000090  52 4B 57 49 4E 32 4B 2D 4E 41 54 49 56 45 2E 43   RKWIN2K-NATIVE.C
000000A0  55 50 2E 48 50 2E 43 4F 4D A4 11 18 0F 32 30 30   UP.HP.COMñ◄↑♫200
000000B0  32 31 31 31 32 32 33 31 39 33 33 5A A5 11 18 0F   211122319333Z╤◄↑♫
000000C0  32 30 30 32 31 31 31 33 30 39 31 39 33 33 5A A7   200211130919333Z°
000000D0  06 02 04 3D D1 8C 85 A8 05 30 03 02 01 03 A9 11   ♠☻♦=╤îà¿♣0♥☻☺♥┌◄
000000E0  30 0F 30 0D A0 03 02 01 02 A1 06 04 04 0F 0D 72   0♫0♪á♥☻☻í♠♦♦♫♪r
000000F0  D4                                                ╘
```

User Datagram Protocol packet | F#: 1/5 | Off: 34 (x22) | L: 8 (x8)

...Comparison to *Network Monitor*...

Screenshot 2a: **`kinit`**(1) captured with *Ethereal* 0.9.4 on HP-UX 11.0, filter definition:

- *host_A → host_B*

- KRB5_AS_REQ/REP packets recognized, and...

kinit.libpcap - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
1	0.000000	15.13.114.212	15.13.115.184	KRB5	AS-REQ
2	0.003210	HP_cb:99:8a	Broadcast	ARP	Who has 15.13.114.212? Tell 15.13.115.184
3	0.003288	Hewlett-_03:9f:26	HP_cb:99:8a	ARP	15.13.114.212 is at 00:10:83:03:9f:26
4	0.003636	15.13.115.184	15.13.114.212	KRB5	KRB-ERROR
5	1.801654	15.13.114.212	15.13.115.184	KRB5	AS-REQ
6	1.807832	15.13.115.184	15.13.114.212	KRB5	AS-REP

...Comparison to *Network Monitor*...

Screenshot 2b: KRB5_AS_REQ with *Ethereal*

- ...Kerberos packets are decoded in detail

```
⊞ Frame 1 (241 bytes on wire, 241 bytes captured)
⊞ Ethernet II, Src: 00:10:83:03:9f:26, Dst: 08:00:09:cb:99:8a
⊞ Internet Protocol, Src Addr: 15.13.114.212 (15.13.114.212), Dst Addr: 15.13.115.184 (15.13.115.184)
⊞ User Datagram Protocol, Src Port: 61469 (61469), Dst Port: kerberos5 (88)
⊟ Kerberos
    Version: 5
    MSG Type: AS-REQ
  ⊟ Request
      Options: 0000000000
    ⊟ Client Name: eric
        Type: Principal
        Name: eric
      Realm: RKWIN2K-NATIVE.CUP.HP.COM
    ⊟ Server Name: krbtgt
        Type: Unknown
        Name: krbtgt
        Name: RKWIN2K-NATIVE.CUP.HP.COM
      Start Time: 2003-01-27 07:38:38 (Z)
      End Time: 2003-01-27 17:38:38 (Z)
      Random Number: 1043653118
    ⊟ Encryption Types
        Type: des-cbc-md5
  ⊟ Addresses
      Type: IPv4
      Value: 15.13.114.212
```

...Comparison to *Network Monitor*...

Screenshot 2c: KRB5_AS_REQ with *Ethereal*

- non-printing characters represented by dots
- 16-byte rows divided down middle
- lines spaced at 1.0

```
0000  08 00 09 cb 99 8a 00 10  83 03 9f 26 08 00 45 00  ........ ...&..E.
0010  00 e3 d5 85 00 00 40 11  9f de 0f 0d 72 d4 0f 0d  ......@. ....r...
0020  73 b8 dc ec 00 58 00 cf  70 1f 6a 81 c4 30 81 c1  s....X.. p.j..0..
0030  a1 03 02 01 05 a2 03 02  01 0a a4 81 b4 30 81 b1  ........ .....0..
0040  a0 07 03 05 00 00 00 00  00 a1 11 30 0f a0 03 02  ........ ...0....
0050  01 01 a1 08 30 06 1b 04  65 72 69 63 a2 1b 1b 19  ....0... eric....
0060  52 4b 57 49 4e 32 4b 2d  4e 41 54 49 56 45 2e 43  RKWIN2K- NATIVE.C
0070  55 50 2e 48 50 2e 43 4f  4d a3 2e 30 2c a0 03 02  UP.HP.CO M..0,...
0080  01 00 a1 25 30 23 1b 06  6b 72 62 74 67 74 1b 19  ...%0#.. krbtgt..
0090  52 4b 57 49 4e 32 4b 2d  4e 41 54 49 56 45 2e 43  RKWIN2K- NATIVE.C
00a0  55 50 2e 48 50 2e 43 4f  4d a4 11 18 0f 32 30 30  UP.HP.CO M....200
00b0  32 31 31 31 32 32 33 31  39 33 33 5a a5 11 18 0f  21112231 933Z....
00c0  32 30 30 32 31 31 31 33  30 39 31 39 33 33 5a a7  20021113 091933Z.
00d0  06 02 04 3d d1 8c 85 a8  05 30 03 02 01 03 a9 11  ...=.... .0......
00e0  30 0f 30 0d a0 03 02 01  02 a1 06 04 04 0f 0d 72  0.0..... .......r
00f0  d4                                                 .
Filter:                                  Reset  Apply  File: <capture>  Drops: 0
```

...Comparison to *Network Monitor*...

- Note *Ethereal*'s superior clock resolution (*time* column) in the summary panes to that of *Network Monitor*. *Ethereal* on Windows 2000 yields similarly impressive results.

kinit.libpcap - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
1	0.000000	15.13.114.212	15.13.115.184	KRB5	AS-REQ
2	0.003210	HP_cb:99:8a	Broadcast	ARP	Who has 15.13.114.212? Tell 15.13.115.184
3	0.003288	Hewlett-_03:9f:26	HP_cb:99:8a	ARP	15.13.114.212 is at 00:10:83:03:9f:26
4	0.003636	15.13.115.184	15.13.114.212	KRB5	KRB-ERROR
5	1.801654	15.13.114.212	15.13.115.184	KRB5	AS-REQ
6	1.807832	15.13.115.184	15.13.114.212	KRB5	AS-REP

Microsoft Network Monitor - [Capture: 23 (Detail)]

File Edit Display Tools Options Window Help

Frame	Time	Src MAC Addr	Dst MAC Addr	Protocol	Description
1	4.466422	hpntc263	LOCAL	UDP	Src Port: Unknown, (61469); Dst Port: Unkno...
2	4.466422	LOCAL	*BROADCAST	ARP_RARP	ARP: Request, Target IP: 15.13.114.212
3	4.466422	hpntc263	LOCAL	ARP_RARP	ARP: Reply, Target IP: 15.13.115.184 Target...
4	4.466422	LOCAL	hpntc263	UDP	Src Port: Unknown, (88); Dst Port: Unknown ...
5	6.259000	hpntc263	LOCAL	UDP	Src Port: Unknown, (61471); Dst Port: Unkno...
6	6.269014	LOCAL	hpntc263	UDP	Src Port: Unknown, (88); Dst Port: Unknown ...
7	0.000000	XEROX 000000	XEROX 000000	STATS	Number of Frames Captured = 6

Getting started: *tcpdump* capture filters...

What is *tcpdump*?

- Open-source text-based network trace facility
- Well-known, standard utility, in use for over ten years
- Originally developed at Lawrence Berkeley National Lab
- Uses the *libpcap* library to capture network traffic
- *tcpdump* and *libpcap* are actively maintained by **The tcpdump Group** (www.tcpdump.com)
- Advantages of *tcpdump*:
  - consumes minimal system resources (no X processing)
  - easy to use, yet supports complex filtering syntax (*libpcap*)
  - detail of output can be controlled, header to full dump
  - does respectable job decoding and formatting SMBs

...Getting started: *tcpdump* capture filters...

- *Ethereal* uses the *libpcap* packet-capture library of *tcpdump* (www.tcpdump.org), so *libpcap* filter syntax is used in *Ethereal*.

- The *libpcap* filter language allows for complex constructs. "This is explained in the *tcpdump* man page. If you can understand it, you are a better man than I..."
  –*Ethereal* User's Manual

- Basic syntax structure:

```
[not] primitive [and|or [not] primitive ...]
```

...Getting started: *tcpdump* capture filters...

tcpdump examples:

- Capture packets from host *A* to host *B* (*A* and *B* can be specified as hostnames or IP addresses):

```
$ tcpdump src A and dst B
```

- Capture all traffic between host *A* and host *B*:

```
$ tcpdump host A and host B
```

or between three hosts:

```
$ tcpdump \( host A and host B \) \
          or \( host B and host C \)  \
          or \( host C and host A \)
```

...Getting started: *tcpdump* capture filters...

More *tcpdump* examples:

- Capture all `telnet` traffic not from ip address 10.0.0.5:

```
$ tcpdump tcp port 23 and \
      not host 10.0.0.5
```

- Capture only SMBs:

```
$ tcpdump tcp[24:4] = 0xff534d42
```

- From the *tcpdump* manpage:
  To print the start and end packets (the SYN and FIN packets) of each TCP conversation that involves a non-local host:

```
$ tcpdump 'tcp[13] & 3 != 0 and \
      not src and dst net localnet'
```

...Getting started: *tcpdump* capture filters...

- For most purposes, **`host A [and host B [...]]`** is sufficient:

```
hpntc263
# tcpdump -q host hpntc825 and host hpntc263
tcpdump: listening on lan1
16:09:09.496741 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 88
16:09:09.518131 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 488 (DF)
16:09:09.537570 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 108
16:09:09.538296 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 105 (DF)
16:09:09.547261 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 112
16:09:09.547927 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 109 (DF)
16:09:09.558142 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 116
16:09:09.558815 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 114 (DF)
16:09:09.567772 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 108
16:09:09.568483 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 107 (DF)
16:09:09.601944 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 104
16:09:09.602796 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 104 (DF)
16:09:09.612554 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 104
16:09:09.613349 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 105 (DF)
16:09:09.622078 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 104
16:09:09.622812 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 105 (DF)
16:09:09.632253 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 104
16:09:09.632998 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 104 (DF)
16:09:09.642961 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 108
16:09:09.643692 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 106 (DF)
16:09:09.652259 hpntc263.cup.hp.com.63412 > hpntc825.cup.hp.com.netbios_ssn: tcp 104
16:09:09.653015 hpntc825.cup.hp.com.netbios_ssn > hpntc263.cup.hp.com.63412: tcp 102 (DF)
```

Notes: host representation = host.domain.port
tcp *x* = length of tcp segment
DF = do-not-fragment flag

...Getting started: *tcpdump* capture filters...

- *tcpdump* also does respectable job decoding SMBs (decoder written by Andrew Tridgell of Samba Team)

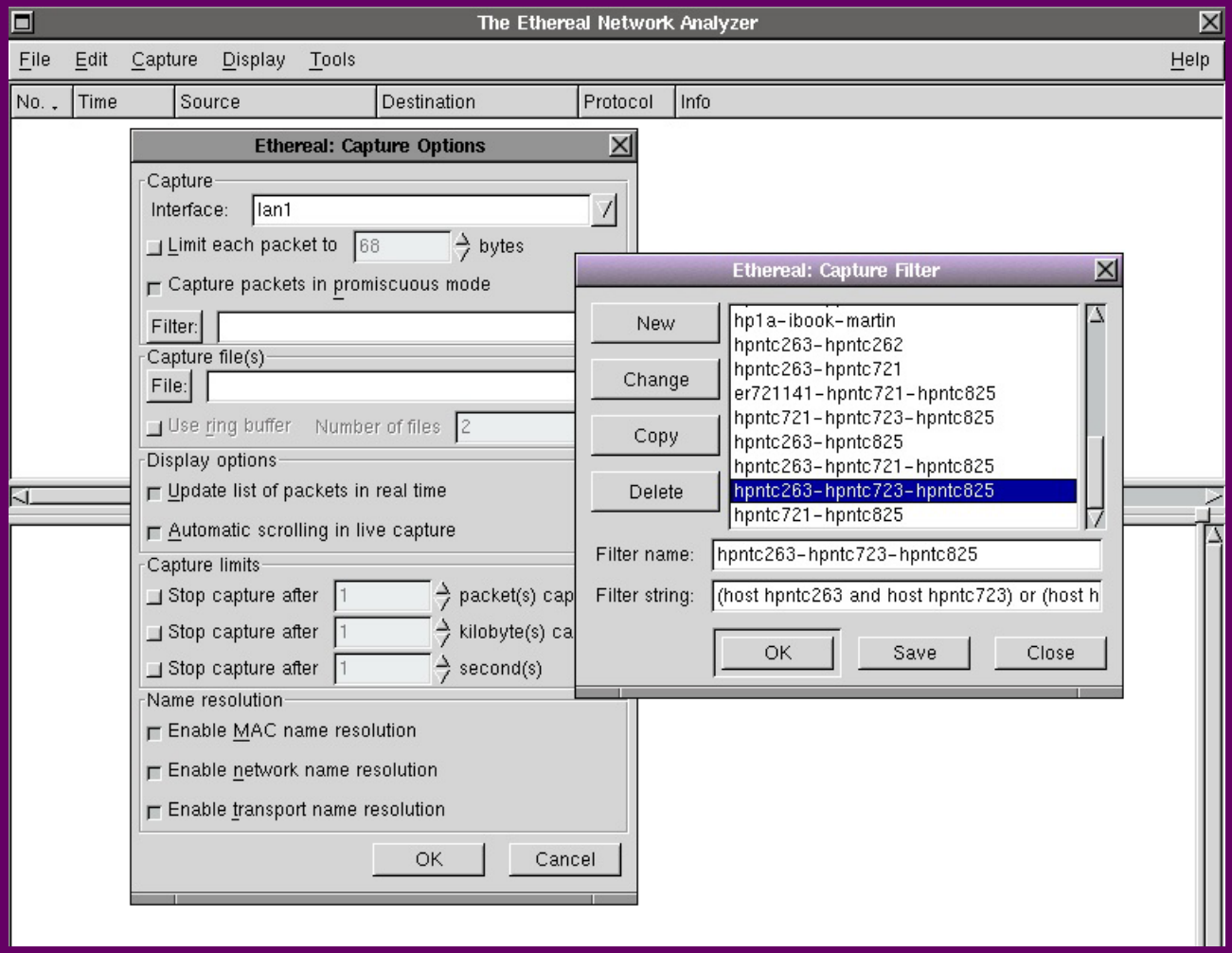
hpntc263

```
$ tcpdump -vvv host hpntc825 and host hpntc263
listening on lan0...
16:37:22.339266 hpntc263.cup.hp.com.64652 > hpntc825.cup.hp.com.netbios_ssn: P 3613323236:3613323324(88)
ack 4249154160 win 32768
>>> NBT Packet
NBT Session Packet
Flags=0x0
Length=84 (0x54)

SMB PACKET: SMBtrans2 (REQUEST)
SMB Command   =  0x32
Error class   =  0x0
Error code    =  0 (0x0)
Flags1        =  0x62
Flags2        =  0x3
Tree ID       =  2055 (0x807)
Proc ID       =  0 (0x0)
UID           =  2050 (0x802)
MID           =  104 (0x68)
Word Count    =  15 (0xf)
TRANSACT2_FINDFIRST param_length=15 data_length=0
TotParam=15 (0xf)
TotData=0 (0x0)
MaxParam=64 (0x40)
MaxData=32768 (0x8000)
MaxSetup=0 (0x0)
Flags=0x0
TimeOut=0 (0x0)
Res1=0x0
ParamCnt=15 (0xf)
ParamOff=68 (0x44)
DataCnt=0 (0x0)
DataOff=0 (0x0)
SetupCnt=1 (0x1)
TransactionName=SMB2
Attribute=HIDDEN SYSTEM DIR
SearchCount=63 (0x3f)
Flags=0x2
Level=260 (0x104)
File=*
```

...Getting started: *tcpdump* capture filters

- Starting a trace: Capture → Start → Filter



Getting started: active trace

- A trace in progress:

The Ethereal Network Analyzer

File Edit Capture Display Tools Help

Start... Ctl+K
Stop Ctl+E

No.	Time	Destination	Protocol	Info
13	0.1[illegible]	hpntc263.cup.hp.com	SMB	Tree Connect AndX Response
14	0.1[illegible]	hpntc825.cup.hp.com	SMB	Transaction2 Request FIND_FIRST2, Pattern: *
15	0.131787	hpntc263.cup.hp.com	SMB	Transaction2 Response FIND_FIRST2, Files: . .. lan_traces
16	0.184023	hpntc825.cup.hp.com	TCP	65285 > netbios_ssn [ACK] Seq=70080868 Ack=1091081291 Wir
17	20.070371	hpntc825.cup.hp.com	SMB	Tree Connect AndX Request, Path: \\HPNTC825.RKWIN2K-NATI\
18	20.070762	hpntc723.cup.hp.com	SMB	Tree Connect AndX Response
19	20.071290	hpntc825.cup.hp.com	SMB	Transaction2 Request GET_DFS_REFERRAL, File:
20	20.071788	hpntc723.cup.hp.com	SMB	Transaction2 Response GET_DFS_REFERRAL
21	20.080265	hpntc825.cup.hp.com	SMB	Transaction2 Request GET_DFS_REFERRAL, File: \RKWIN2K-NAT
22	20.080722	hpntc723.cup.hp.com	SMB	Transaction2 Response GET_DFS_REFERRAL
23	20.082607	hpntc825.cup.hp.com	SMB	Transaction2 Requ[illegible] KWIN2K-NAT
24	20.083007	hpntc723.cup.hp.com	SMB	Transaction2 Resp[illegible]
25	20.209903	hpntc825.cup.hp.com	TCP	1255 > 445 [ACK] [illegible] Win=16261
26	31.413004	hpntc825.cup.hp.com	SMB	Tree Disconnect [illegible]
27	31.413255	hpntc723.cup.hp.com	SMB	Tree Disconnect [illegible]
28	31.569126	hpntc825.cup.hp.com	TCP	1255 > 445 [ACK] [illegible] Win=16222

Frame 1 (62 bytes on wire, 62 bytes captured)
Ethernet II, Src: 00:10:83:03:9f:26, Dst: 08:00:09:cb:99:8a
Internet Protocol, Src Addr: hpntc263.cup.hp.com (15.13.114.212), Dst Addr: hpntc825.cu
Transmission Control Protocol, Src Port: 65285 (65285), Dst Port: netbios_ssn (139), Se

```
0000  08 00 09 cb 99 8a 00 10  83 03 9f 26 08 00 45 00   ........ ...&..E.
0010  00 30 99 46 00 00 40 06  dc db 0f 0d 72 d4 0f 0d   .0.F..@. ....r...
0020  73 b8 ff 05 00 8b 04 2d  53 23 00 00 00 00 70 02   s......- S#....p.
0030  80 00 a9 96 00 00 02 04  05 b4 03 03 00 01         ........ ......
```

Ethereal: Capture

Captured Frames

Total	28	(100.0%)
SCTP	0	(0.0%)
TCP	26	(92.9%)
UDP	2	(7.1%)
ICMP	0	(0.0%)
ARP	0	(0.0%)
OSPF	0	(0.0%)
GRE	0	(0.0%)
NetBIOS	0	(0.0%)
IPX	0	(0.0%)
VINES	0	(0.0%)
Other	0	(0.0%)

Running 00:00:47

Stop

Filter: Reset Apply <live capture in progress>

Extracting data…

Once network traffic is captured, how does one isolate the data of interest?

Ethereal provides multiple methods:

- Flexible C-style display filter syntax
- Colorizing display
- Edit → Find Frame

...Extracting data: display filters...

Ethereal display filter syntax, basic expression structure:

```
[!] E [rel-op val] [log-ops E [rel-op val]]...
```

where an element **E** is:

```
protocol[.field_1[.field_2]][substr]
```

the relational operators **rel-op** are:

```
==   !=   >   <   >=   <=
```

or

```
eq   ne   gt   lt   ge   le
```

and the logical operators **log-op** are:

```
and   or   not   xor
```

or

```
&&   ||   !   ^^
```

...Extracting data: display filters...

Ethereal display-filter examples:

- Display only the SMBs in a trace:

```
smb
```

- Display only SMB and Kerberos packets:

```
smb || kerberos
```

- Display only NetBIOS Session Service packets not containing SMBs:

```
nbss && !smb
```

...Extracting data: display filters...

More *Ethereal* display-filter examples:

- Display only packets from host *A* (ip address 1.2.3.4) to host *B* (ip address 5.6.7.8):

```
ip.src == A && ip.dst == B
```

or

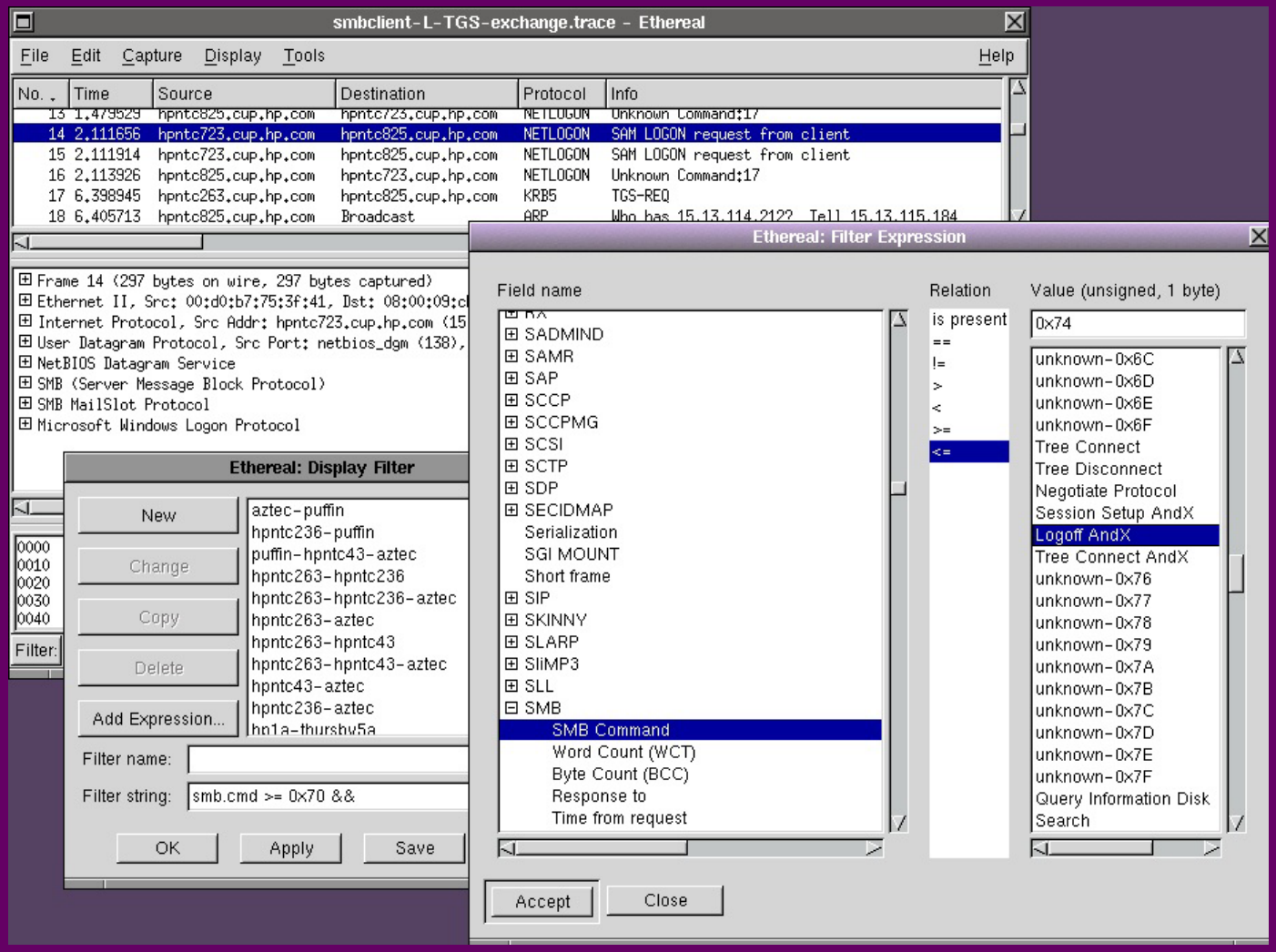
```
ip.src eq 1.2.3.4 && ip.dst eq 5.6.7.8
```

- Display only **CIFS_NEGOTIATE** replies with **CAP_UNIX** bit set:

```
smb.server_cap.unix == 1
```

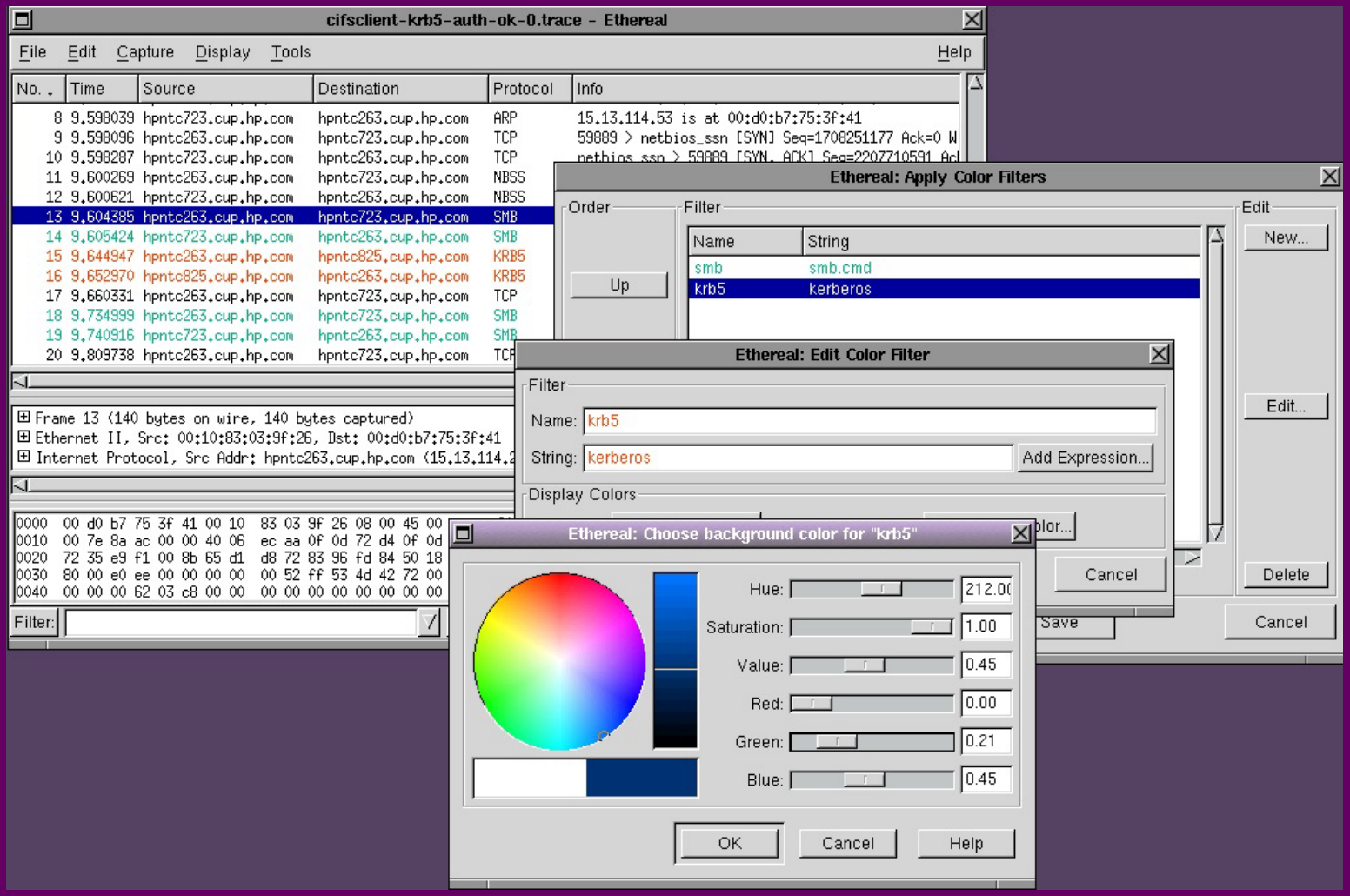
...Extracting data: display filters...

Ethereal interactive display-filter builder:



...Extracting data: colorizing the display...

Colorizing display:



...Extracting data: Find Frame

Edit → Find Frame

cifsclient-krb5-auth-ok-0.trace - Ethereal

File Edit Capture Display Tools Help

No.	Time	Source	Destination	Protocol	Info
12	9.600621	hpntc723.cup.hp.com	hpntc263.cup.hp.com	NBSS	Positive session response
13	9.604385	hpntc263.cup.hp.com	hpntc723.cup.hp.com	SMB	Negotiate Protocol Request
14	9.605424	hpntc723.cup.hp.com	hpntc263.cup.hp.com	SMB	Negotiate Protocol Response
15	9.644947	hpntc263.cup.hp.com	hpntc825.cup.hp.com	KRB5	

Ethereal: Find Frame

Filter: smb.flags2.esn

Forward Backward

OK Cancel

```
⊞ Flags: 0x62
⊟ Flags2: 0xc803
    1... .... .... .... = Unicode Strings: Strings are Uni
    .1.. .... .... .... = Error Code Type: Error codes are
    ..0. .... .... .... = Execute-only Reads: Don't permit
    ...0 .... .... .... = Dfs: Don't resolve pathnames wit
    .... 1... .... .... = Extended Security Negotiation: Extended security negotiation is supported
    .... .... .0.. .... = Long Names Used: Path names in request are not long file names
    .... .... .... .0.. = Security Signatures: Security signatures are not supported
    .... .... .... ..1. = Extended Attributes: Extended attributes are supported
    .... .... .... ...1 = Long Names Allowed: Long file names are allowed in the response
  Reserved: 0000000000000000000000000
```

```
0000  00 d0 b7 75 3f 41 00 10  83 03 9f 26 08 00 45 00   ...u?A.. ...&..E.
0010  00 7e 8a ac 00 00 40 06  ec aa 0f 0d 72 d4 0f 0d   .~....@. ....r...
0020  72 35 e9 f1 00 8b 65 d1  d8 72 83 96 fd 84 50 18   r5....e. .r....P.
0030  80 00 e0 ee 00 00 00 00  00 52 ff 53 4d 42 72 00   ........ .R.SMBr.
0040  00 00 00 62 03 c8 00 00  00 00 00 00 00 00 00 00   ...b.... ........
```

Filter: Reset Apply File: cifsclient-krb5-auth-ok-0.trace

Tracing for indefinite periods...

Problem: How to capture traffic for an indefinite period, while controlling disk consumption and size of trace files.

Solution: *tethereal* "ring buffers"

- *tethereal* is the terminal (non GUI) version of *Ethereal*
- ring buffers are capture files: when the last is full, the first is reused
- user specifies number of buffers (**`-b`** option), size in Kb or number of packets (**`-a`** option), and basename for output files (**`-w`** option)
- capture files are binary; they can be opened in *Ethereal* or displayed as text by *tethereal*

...Tracing for indefinite periods

Ring buffer example:

Run *tethereal* for an indefinite period, using four 1-Mb ring buffers:

```
$ tethereal -a filesize:1024 -b 4 -w eth.out
```

- terminate with **[Ctrl] [C]**, or from shell script with
  kill -s INT *tethereal_process_id*
- do not terminate with **kill -s KILL** (signal 9)
- output (note: file 1 reused—has most recent *mtime*):

```
$ ll -rt eth*
-rw------- 1 root sys 1024897 Mar 22 16:53 eth_00002_20050322165358.out
-rw------- 1 root sys 1025096 Mar 22 16:53 eth_00003_20050322165359.out
-rw------- 1 root sys 1025100 Mar 22 16:54 eth_00004_20050322165359.out
-rw------- 1 root sys  485822 Mar 22 16:54 eth_00001_20050322165400.out
```

Conversion to and from other formats

- *Ethereal* easily reads and writes *tcpdump* (*libpcap*), *nettl* and *Network Monitor* traces with no special action required of user. It even unpacks gzipped files on the fly, via *libz*. Simply do File → Open to read other formats directly.
- *editcap* can also perform conversions:

```
editcap [options] -F format infile outfile
```

For example, to convert a *nettl* trace to *Network Monitor v.1* format:

```
$ editcap -v -F netmon1 nettl.out.TRC0 \
    nettl-to-netmon.cap
```

Installation, dependencies…

Where to get *Ethereal* bundles:

- Source code, documentation, etc.:

  http://www.ethereal.com

- SD depots for HP-UX:

  http://software.hp.com
  (from “Internet Express” bundle—search for “ethereal”)

...Installation, dependencies...

On Unix and Linux, *Ethereal* depends on the following open-source software:

- gettext
- glib
- gtk+
- libiconv
- libpcap
- snmp
- zlib

These are available on most Linux distributions, but on HP-UX they may have to be installed in order to compile or run *Ethereal*...

…Installation, dependencies…

SD depots for *Ethereal*'s dependencies are available at the HP-UX Porting and Archive Centre:

http://hpux.cs.utah.edu/

NOTE: *Ethereal's* dependencies sometimes change with new versions.

…Installation, dependencies

On Windows, *Ethereal* depends only on the Win32 port of *libpcap*, known as *WinPcap*. This consists of two dynamic link libraries: **`packet.dll`** and **`wpcap.dll`**, both released under a "BSD-style" license, and available at:

http://winpcap.polito.it/

Resources, Mailing Lists

The *Ethereal* website, www.ethereal.com, contains a wealth of information, including man pages and a 454-page user manual.

Under the “Resources” section are links to:

- various mailing lists: announce, users, dev, doc, cvs
- sample captures
- useful links: lots of information on protocols
- etc.

There is a wish list; you can add your request!

Questions

Questions?

Thank you

Thank you, and happy