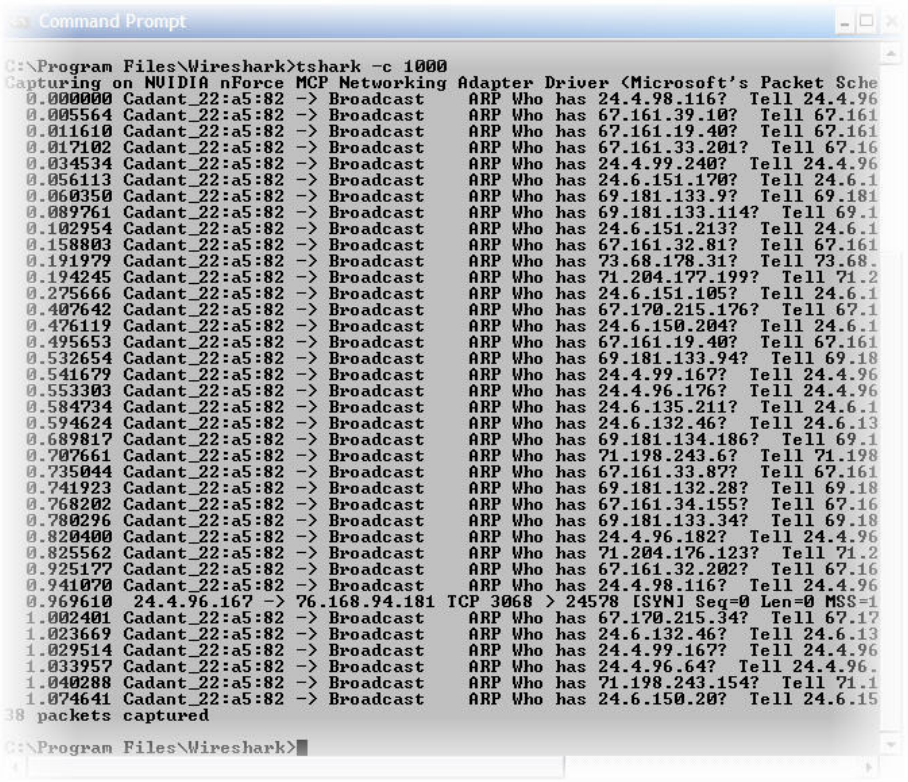
## WSU03: Wireshark Troubleshooting
## Network Performance

# Appendix E: Command-Line Tool Reference



## *Tshark*

```
TShark 0.99.4 (SVN Rev 19757)
Dump and analyze network traffic.
See http://www.wireshark.org for more information.

Copyright 1998-2006 Gerald Combs <gerald@wireshark.org> and contributors.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

Usage: tshark [options] ...

Capture interface:
  -i <interface>           name or idx of interface (def: first non-loopback)
  -f <capture filter>      packet filter in libpcap filter syntax
  -s <snaplen>             packet snapshot length (def: 65535)
  -p                       don't capture in promiscuous mode
  -B <buffer size>         size of kernel buffer (def: 1MB)
  -y <link type>           link layer type (def: first appropriate)
  -D                       print list of interfaces and exit
  -L                       print list of link-layer types of iface and exit

Capture stop conditions:
  -c <packet count>        stop after n packets (def: infinite)
  -a <autostop cond.> ...  duration:NUM - stop after NUM seconds
                           filesize:NUM - stop this file after NUM KB
                              files:NUM - stop after NUM files
Capture output:
  -b <ringbuffer opt.> ... duration:NUM - switch to next file after NUM secs
                           filesize:NUM - switch to next file after NUM KB
                              files:NUM - ringbuffer: replace after NUM files
Input file:
  -r <infile>              set the filename to read from (no pipes or stdin!)

Processing:
  -R <read filter>         packet filter in Wireshark display filter syntax
  -n                       disable all name resolutions (def: all enabled)
  -N <name resolve flags>  enable specific name resolution(s): "mntC"
  -d <layer_type>==<selector>,<decode_as_protocol> ...
                           "Decode As", see the man page for details
                           Example: tcp.port==8888,http
Output:
  -w <outfile|->           set the output filename (or '-' for stdout)
  -F <output file type>    set the output file type, default is libpcap
                           an empty "-F" option will list the file types
  -V                       add output of packet tree        (Packet Details)
  -x                       add output of hex and ASCII dump (Packet Bytes)
  -T pdml|ps|psml|text     output format of text output (def: text)
  -t ad|a|r|d              output format of time stamps (def: r: rel. to first)
  -l                       flush output after each packet
  -q                       be more quiet on stdout (e.g. when using statistics)
  -X <key>:<value>         eXtension options, see the man page for details
  -z <statistics>          various statistics, see the man page for details

Miscellaneous:
  -h                       display this help and exit
  -v                       display version info and exit
  -o <name>:<value> ...    override preference setting
```

## *Editcap*

```
Editcap 0.99.4 (SVN Rev 19757)
Edit and/or translate the format of capture files.
See http://www.wireshark.org for more information.

Usage: editcap [options] ... <infile> <outfile> [ <packet#>[-<packet#>] ... ]

A single packet or a range of packets can be selected.

Packets:
  -C <choplen>           chop each packet at the end by <choplen> bytes
  -d                     remove duplicate packets
  -E <error probability> set the probability (between 0.0 and 1.0 incl.)
                         that a particular packet byte will be randomly changed
  -r                     keep the selected packets, default is to delete them
  -s <snaplen>           truncate packets to max. <snaplen> bytes of data
  -t <time adjustment>   adjust the timestamp of selected packets,
                         <time adjustment> is in relative seconds (e.g. -0.5)
  -A <start time>        don't output packets whose timestamp is before the
                         given time (format as YYYY-MM-DD hh:mm:ss)
  -B <stop time>         don't output packets whose timestamp is after the
                         given time (format as YYYY-MM-DD hh:mm:ss)

Output File(s):
  -c <packets per file>  split the packet output to different files,
                         with a maximum of <packets per file> each
  -F <capture type>      set the output file type, default is libpcap
                         an empty "-F" option will list the file types
  -T <encap type>        set the output file encapsulation type,
                         default is the same as the input file
                         an empty "-T" option will list the encapsulation types

Miscellaneous:
  -h                     display this help and exit
  -v                     verbose output
```

## *Capinfos*

```
Capinfos 0.99.4
Prints information about capture files.
See http://www.wireshark.org for more information.

Usage: capinfos [options] <infile> ...

General:
  -t display the capture file type

Size:
  -c display the number of packets
  -s display the size of the file (in bytes)
  -d display the total length of all packets (in bytes)

Time:
  -u display the capture duration (in seconds)
  -a display the capture start time
  -e display the capture end time

Statistic:
  -y display average data rate (in bytes/s)
  -i display average data rate (in bits/s)
  -z display average packet size (in bytes)

Miscellaneous:
  -h display this help and exit
```

## *Dumpcap*

```
Dumpcap 0.99.4 (SVN Rev 19757)
Capture network packets and dump them into a libpcap file.
See http://www.wireshark.org for more information.

Usage: dumpcap [options] ...

Capture interface:
  -i <interface>           name or idx of interface (def: first none loopback)
  -f <capture filter>      packet filter in libpcap filter syntax
  -s <snaplen>             packet snapshot length (def: 65535)
  -p                       don't capture in promiscuous mode
  -B <buffer size>         size of kernel buffer (def: 1MB)
  -y <link type>           link layer type (def: first appropriate)
  -D                       print list of interfaces and exit
  -L                       print list of link-layer types of iface and exit

Stop conditions:
  -c <packet count>        stop after n packets (def: infinite)
  -a <autostop cond.> ...  duration:NUM - stop after NUM seconds
                           filesize:NUM - stop this file after NUM KB
                              files:NUM - stop after NUM files
Output (files):
  -w <filename>            name of file to save (def: tempfile)
  -b <ringbuffer opt.> ... duration:NUM - switch to next file after NUM secs
                           filesize:NUM - switch to next file after NUM KB
                              files:NUM - ringbuffer: replace after NUM files
Miscellaneous:
  -v                       print version information and exit
  -h                       display this help and exit

Example: dumpcap -i eth0 -a duration:60 -w output.pcap
"Capture network packets from interface eth0 until 60s passed into output.pcap"

Use Ctrl-C to stop capturing at any time.
```

## *Mergecap*

```
Mergecap 0.99.4 (SVN Rev 19757)
Merge two or more capture files into one.
See http://www.wireshark.org for more information.

Usage: mergecap [options] -w <outfile|-> <infile> ...

Output:
  -a                 files should be concatenated, not merged
                     Default merges based on frame timestamps
  -s <snaplen>       truncate packets to <snaplen> bytes of data
  -w <outfile|->     set the output filename to <outfile> or '-' for stdout
  -F <capture type> set the output file type, default is libpcap
                     an empty "-F" option will list the file types
  -T <encap type>    set the output file encapsulation type,
                     default is the same as the first input file
                     an empty "-T" option will list the encapsulation types

Miscellaneous:
  -h                 display this help and exit
  -v                 verbose output
```

## *Text2pcap*

```
Text2pcap 0.99.4
Generate a capture file from an ASCII hexdump of packets.
See http://www.wireshark.org for more information.

Usage: text2pcap [options] <input-filename> <output-filename>

where  <input-filename> specifies input  filename (use - for standard input)
      <output-filename> specifies output filename (use - for standard output)

Input:
  -o hex|oct              parse offsets as (h)ex or (o)ctal, default is hex
  -t <timefmt>            treats the text before the packet as a date/time code;
                          the specified argument is a format string of the sort
                          supported by strptime.
                          Example: The time "10:15:14.5476" has the format code
                          "%H:%M:%S."
                          NOTE: The subsecond component delimiter must be given
                           (.) but no pattern is required; the remaining number
                           is assumed to be fractions of a second.

Output:
  -l <typenum>            link-layer type number. Default is 1 (Ethernet).
                          See the file net/bpf.h for list of numbers.
  -m <max-packet>         max packet length in output, default is 64000

Prepend dummy header:
  -e <l3pid>              prepend dummy Ethernet II header with specified L3PID
                          (in HEX)
                          Example: -e 0x800
  -i <proto>              prepend dummy IP header with specified IP protocol
                          (in DECIMAL).
                          Automatically prepends Ethernet header as well.
                          Example: -i 46
  -u <srcp>,<destp>       prepend dummy UDP header with specified
                          dest and source ports (in DECIMAL).
                          Automatically prepends Ethernet & IP headers as well
                          Example: -u 30,40
  -T <srcp>,<destp>       prepend dummy TCP header with specified
                          dest and source ports (in DECIMAL).
                          Automatically prepends Ethernet & IP headers as well
                          Example: -T 50,60
  -s <srcp>,<dstp>,<tag>  prepend dummy SCTP header with specified
                          dest/source ports and verification tag (in DECIMAL).
                          Automatically prepends Ethernet & IP headers as well
                          Example: -s 30,40,34
  -S <srcp>,<dstp>,<ppi>  prepend dummy SCTP header with specified
                          dest/source ports and verification tag 0.
                          It also prepends a dummy SCTP DATA
                          chunk header with payload protocol identifier ppi.
                          Example: -S 30,40,34

Miscellaneous:
  -h                      display this help and exit
  -d                      detailed debug of parser states
  -q                      generate no output at all (automatically turns off -d)
```