Red Hat Enterprise Linux Release 9.2 Manual Pages on 'tcpslice.8' command

$ man tcpslice.8

```
TCPSLICE(8)              System Manager's Manual              TCPSLICE(8)

NAME

       tcpslice - extract pieces of and/or merge together pcap files

SYNOPSIS

       tcpslice [ -DdlhRrtv ] [ -w file ]

              [ -s types [ -e seconds ] [ -f format ] ]

              [ start-time [ end-time ] ] file ...

DESCRIPTION

       Tcpslice  is  a  program  for extracting portions of packet-trace files

       generated using tcpdump(1)'s -w flag.  It can also be used to merge to?

       gether several such files, as discussed below.

       The  basic  operation of tcpslice is to copy to stdout all packets from

       its input file(s) whose timestamps fall  within  a  given  range.   The

       starting  and ending times of the range may be specified on the command

       line.  All ranges are inclusive.  The starting  time  defaults  to  the

       earliest  time  of  the first packet in any of the input files; we call

       this the first time.  The ending time defaults to ten years  after  the

       starting  time.   Thus,  the  command tcpslice trace-file simply copies

       trace-file to stdout (assuming the file does not include more than  ten

       years' worth of data).

       There  are  a number of ways to specify times.  The first is using Unix

       timestamps of the form sssssssss.uuuuuu (this is the  format  specified

       by  tcpdump's -tt flag).  For example, 654321098.7654 specifies 38 sec?
```

onds and 765,400 microseconds after 8:51PM PDT, Sept. 25, 1990.

All examples in this manual are given for PDT times, but when display?
ing times and interpreting times symbolically as discussed below, tcp?
slice uses the local timezone, regardless of the timezone in which the
pcap file was generated. The daylight-savings setting used is that
which is appropriate for the local timezone at the date in question.
For example, times associated with summer months will usually include
daylight-savings effects, and those with winter months will not.

Times may also be specified relative to either the first time (when
specifying a starting time) or the starting time (when specifying an
ending time) by preceding a numeric value in seconds with a `+'. For
example, a starting time of +200 indicates 200 seconds after the first
time, and the two arguments +200 +300 indicate from 200 seconds after
the first time through 500 seconds after the first time.

Times may also be specified in terms of years (y), months (m), days
(d), hours (h), minutes (m), seconds (s), and microseconds(u). For ex?
ample, the Unix timestamp 654321098.7654 discussed above could also be
expressed as 1990y9m25d20h51m38s765400u. 2 or 4 digit years may be
used; 2 digits can specify years from 1970 to 2069.

When specifying times using this style, fields that are omitted default
as follows. If the omitted field is a unit greater than that of the
first specified field, then its value defaults to the corresponding
value taken from either first time (if the starting time is being spec?
ified) or the starting time (if the ending time is being specified).
If the omitted field is a unit less than that of the first specified
field, then it defaults to zero (1 for days). For example, suppose
that the input file has a first time of the Unix timestamp mentioned
above, i.e., 38 seconds and 765,400 microseconds after 8:51PM PDT,
Sept. 25, 1990. To specify 9:36PM PDT (exactly) on the same date we
could use 21h36m. To specify a range from 9:36PM PDT through 1:54AM
PDT the next day we could use 21h36m 26d1h54m.

Relative times can also be specified when using the ymdhmsu format.
Omitted fields then default to 0 if the unit of the field is greater

than that of the first specified field, and to the corresponding value
taken from either the first time or the starting time if the omitted
field's unit is less than that of the first specified field.   Given a
first time of the Unix timestamp mentioned above, 22h +1h10m specifies
a range from 10:00PM PDT on that date through 11:10PM PDT, and +1h
+1h10m specifies a range from 38.7654 seconds after 9:51PM PDT through
38.7654 seconds after 11:01PM PDT. The first hour of the file could be
extracted using +0 +1h.

Note that with the ymdhmsu format there is an ambiguity between using m
for `month' or for `minute'. The ambiguity is resolved as follows: if
an m field is followed by a d field then it is interpreted as specify?
ing months; otherwise it specifies minutes.

If more than one input file is specified then tcpslice merges the pack?
ets from the various input files into the single output file. Nor?
mally, this merge is done based on the value of the time stamps in the
packets in the individual files. (Tcpslice assumes that within each
input file, packets are in time stamp order.)   If the -l option is
used, the value used for ordering is the time stamp of a given packet
minus the time stamp of the first packet in the input file in which the
given packet occurs.

When merging files, by default tcpslice will discard any duplicate
packet it finds in more than one file. A duplicate is a packet that
has an identical timestamp (either relative or absolute) and identical
packet contents (for as much as was captured) as another packet previ?
ously seen in a different file. Note that it is possible for the net?
work to generate true replicates of packets, and for systems that can
return the same timestamp for multiple packets, these can be mistaken
for duplicates and discarded. Accordingly, tcpslice will not discard
duplicates in the same trace file. In addition, you can use the -D op?
tion to suppress any discarding of duplicates.

OPTIONS

If any of -R, -r or -t are specified then tcpslice reports the time?
stamps of the first and last packets in each input file and exits.

Only one of these three options may be specified.

-D Do not discard duplicate packets seen when merging multiple trace files.

-d Dump the start and end times specified by the given range and exit. This option is useful for checking that the given range actually specifies the times you think it does. If one of -R, -r or -t has been specified then the times are dumped in the corresponding format; otherwise, raw format (-R) is used.

-e Specify a number of seconds to wait after the last packet was seen before considering a session to be expired (default: 0 = do not expire inactive sessions). This is only effective when the -s option is used to track sessions.

-f Specify the name format of PCAP files to which each session will be extracted (default: NULL = do not extract sessions to sepa?rate files). This is only effective when the -s option is used to track sessions.

-h Print the tcpslice and libpcap version strings, print a usage message, and exit.

-l When merging more than one file, merge on the basis of relative time, rather than absolute time. Normally, when merging files is done, packets are merged based on absolute time stamps. With -l packets are merged based on the relative time between the start of the file in which the packet is found and the time stamp of the packet itself. The time stamp of packets in the output file is calculated as the relative time for the packet within its file plus first time.

-R Dump the timestamps of the first and last packets in each input file as raw timestamps (i.e., in the form sssssssss.uuuuuu).

-r Same as -R except the timestamps are dumped in human-readable format, similar to that used by date(1).

-s Enable session tracking for the specified types which is a comma-separated list of the following:

tcp track all TCP connections

sip track SIP-based VoIP calls, which may enable tracking of TCP connections but only the ones that are related to SIP calls. This feature is only available if tcpslice was linked against Aymeric Moizard's GNU oSIP library; if not, install the latest version of libosip2 from https://www.gnu.org/software/osip/ and recompile tcp?slice.

h323 track H.323-based VoIP calls, which may enable tracking of TCP connections but only the ones that are related to H.323 calls. This feature is only available if tcpslice was linked against Objective Systems' Open H.323 library for C; if not, install the latest version of libooh323c from https://sourceforge.net/projects/ooh323c/ and recom?pile tcpslice.

Session tracking altogether is only available if tcpslice was linked against a recent version (>1.20) of Rafal Wojtczuk's Net?work Intrusion Detection System library; if not, install the latest version of libnids from http://libnids.sourceforge.net/ and recompile tcpslice.

-t Same as -R except the timestamps are dumped in tcpslice format, i.e., in the ymdhmsu format discussed above.

-v Turn on verbose mode. Currently this only affects session track?ing (-s) messages: if specified at least once, sessions openings and closings are displayed regardless of the time (by default the closings are only displayed past end-time); if specified at least twice, subsessions (sessions initiated by other sessions) openings and closings are also displayed.

-w Direct the output to file rather than stdout.

SEE ALSO

tcpdump(1)

AUTHORS

The original author was:

Vern Paxson, of Lawrence Berkeley Laboratory, University of California, Berkeley, CA.

It is currently being maintained by The Tcpdump Group.

The current version is available at:

https://github.com/the-tcpdump-group/tcpslice

The original distribution is available via anonymous ftp:

ftp://ftp.ee.lbl.gov/tcpslice-1.2a3.tar.gz

BUGS

Please send problems, bugs, questions, desirable enhancements, etc. to:

tcpdump-workers@lists.tcpdump.org

Please send source code contributions as git pull requests through the project page above.

An input filename that exactly matches the sssssssss.uuuuuu or the ymdhmsu format discussed above can be confused with a start/end time (regardless if the date and the time are valid in the latter case). Such filenames can be specified with a leading \`./'; for example, spec? ify the file \`1976y07m04d' as \`./1976y07m04d' and \`00000123' as \`./00000123'. Alternatively, renaming the files to \`1976y07m04d.pcap' and \`00000123.pcap' respectively would resolve this ambiguity.

tcpslice cannot read its input from stdin, since it uses random-access to rummage through its input files.

tcpslice refuses to write to its output if it is a terminal (as indi? cated by isatty(3)). This is not a bug but a feature, to prevent it from spraying binary data to the user's terminal. Note that this means you must either redirect stdout or specify an output file via -w.

tcpslice will not work properly on pcap files spanning more than one year; with files containing portions of packets whose original length was more than 65,535 bytes; nor with files containing fewer than two packets. Such files result in the error message: \`couldn't find final packet in file'. These problems are due to the interpolation scheme used by tcpslice to greatly speed up its processing when dealing with large trace files. Note that tcpslice can efficiently extract slices from the middle of trace files of any size, and can also work with truncated trace files (i.e., the final packet in the file is only par?

tially present, typically due to tcpdump being ungracefully killed).

Adding -l has broken some compatibility with older versions, since tcp? slice now merges its input files, rather than (approximately) concate? nating them together as it did previously.

It would sometimes be convenient if you could specify a clock offset to use with the -l option.

It would be nice if tcpslice supported more general editing of trace files.