Strip Plots: A Simple Automated Time-Series Visualization

Sid Faber
sfaber@cert.org

Overview

Motivation & Goals

Sample Output

The Basics

Special Features

NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

Motivation and Goals

Caveat

This is analyst code, not engineering code

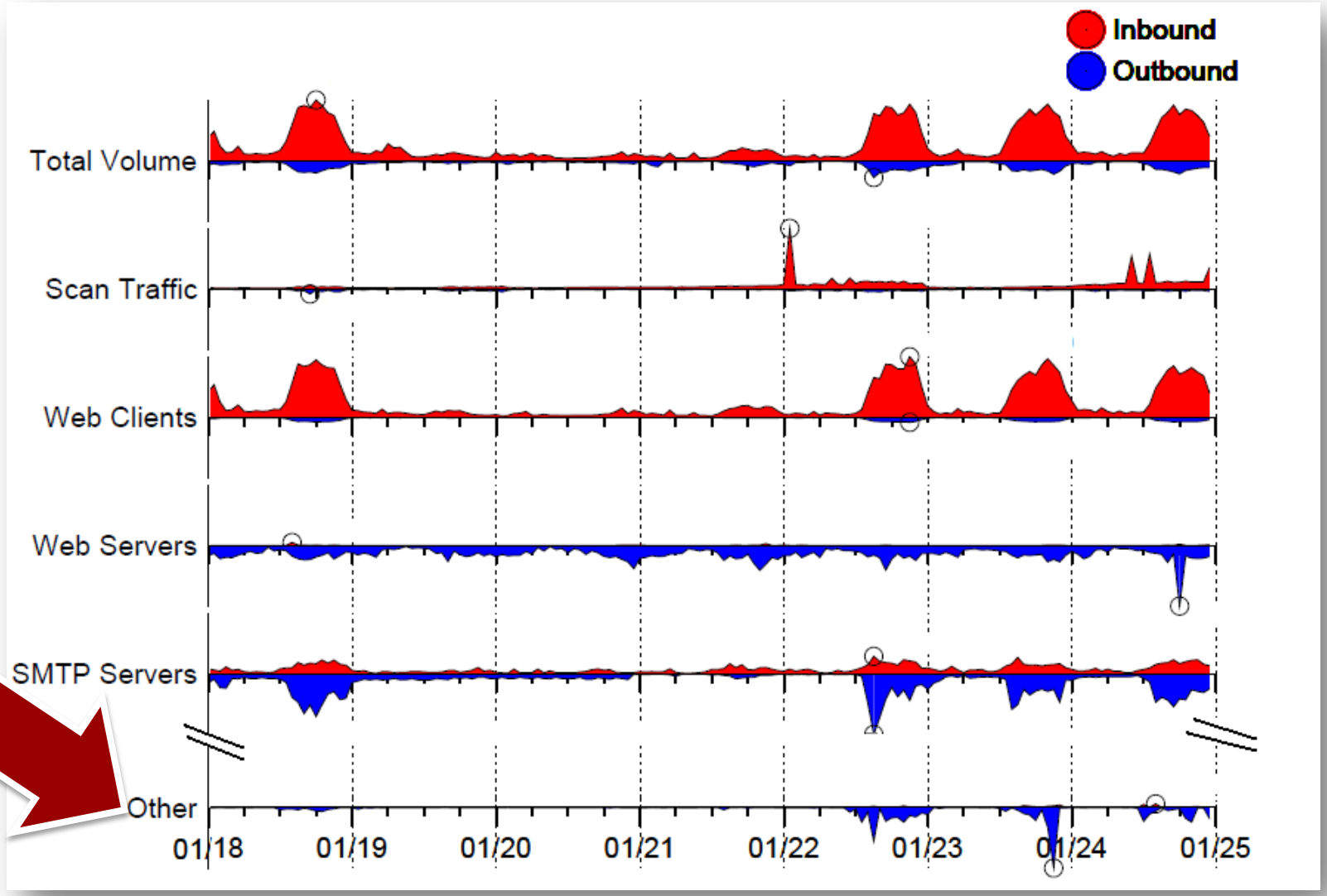
- Your mileage may vary

Motivation

Support network profiling for Situational Awareness

- I know most of what's on my network
  - Based exclusively on past observations
- I can filter / categorize out routine traffic
- What can I do with "leftover" traffic?
  - Is this something new to add to my profile?
  - Is something odd happening on my network?

Example

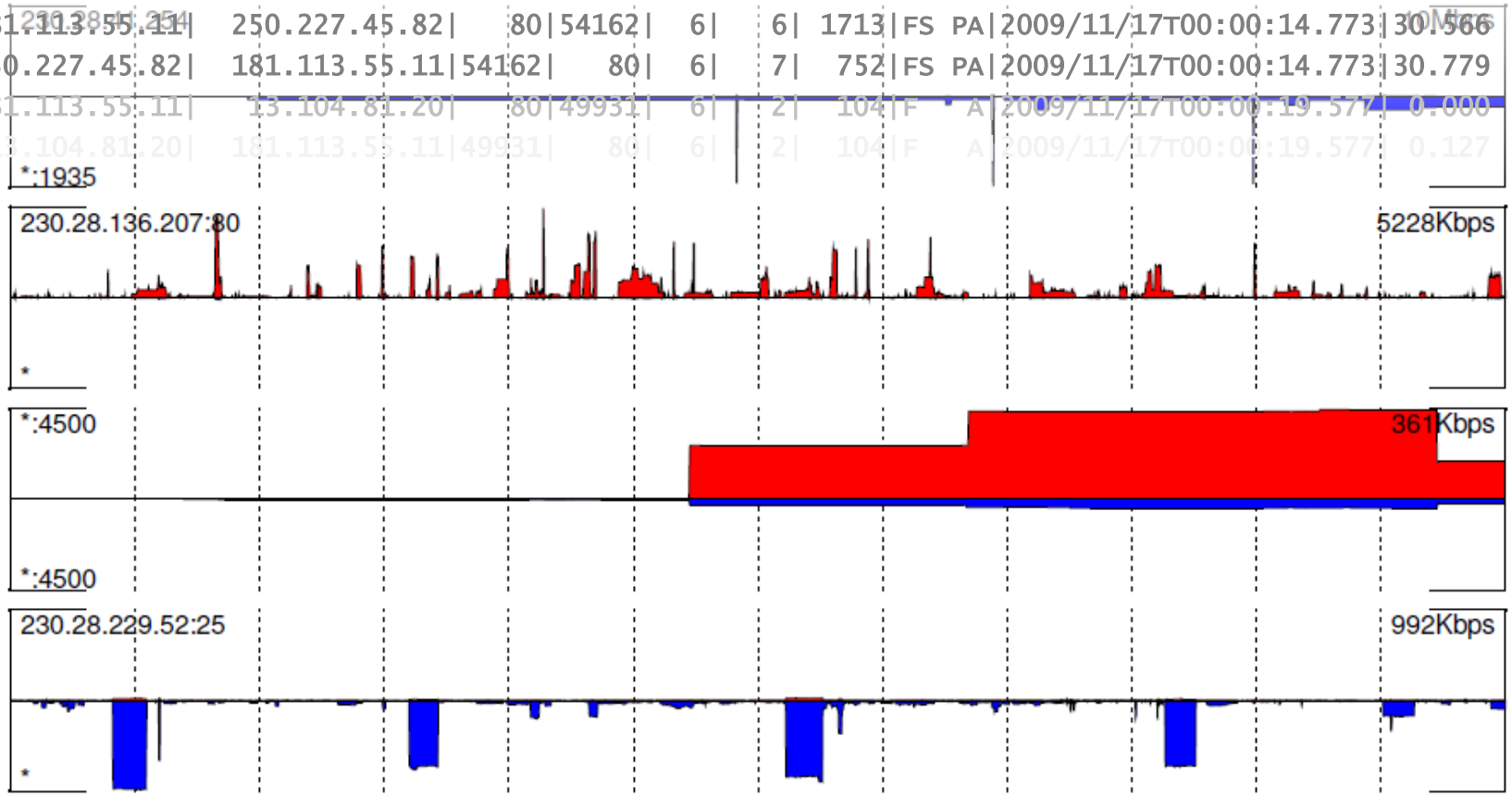


Goals

Nearly self-maintaining network profile

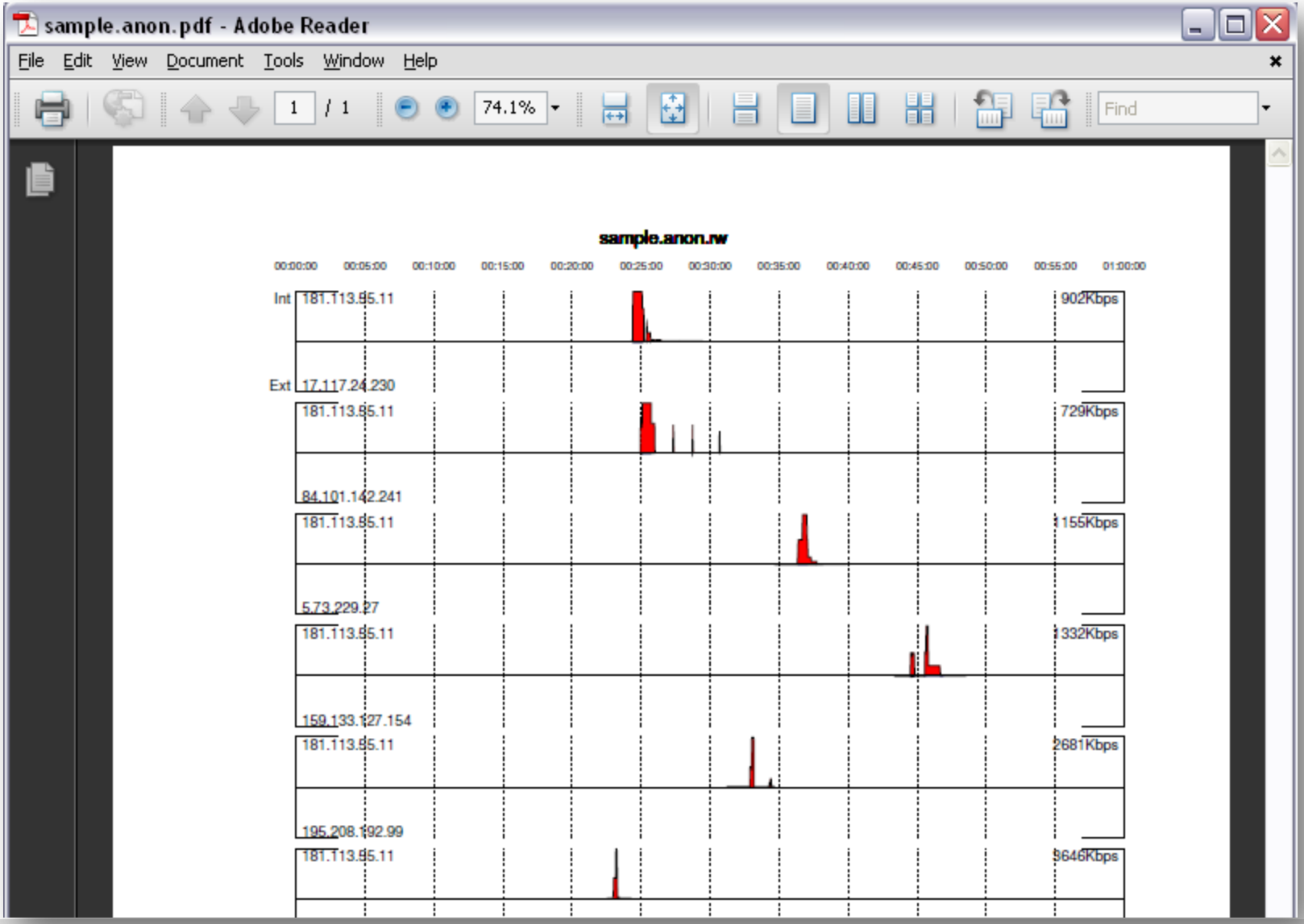
- Batch processing
- Email delivery
- ***Quick*** triage of “leftovers”
  - Add to my profile?
  - Something odd?
  - <5 minutes per report
- Self-sufficient description: ***No Additional Explanation Necessary***

sIP	dIP	sPort	dPort	pro	pkts	bytes	flags	sTime	dur
0.33.6.97	181.113.55.11	63220	80	6	2	104	F A	2009/11/17T00:00:01.236	0.071
181.113.55.11	0.33.6.97	80	63220	6	2	104	F A	2009/11/17T00:00:01.236	0.000
181.113.55.11	94.145.72.124	80	50151	6	8	9016	S PA	2009/11/17T00:00:07.233	0.232
92.4.182.32	181.113.55.11	1067	80	6	20	1822	PA	2009/11/17T00:00:09.827	2.125
181.113.55.11	92.4.182.32	80	1067	6	25	32910	PA	2009/11/17T00:00:09.840	1.713
92.4.182.32	181.113.55.11	1056	80	6	20	1868	PA	2009/11/17T00:00:09.841	2.684
181.113.55.11	92.4.182.32	80	1056	6	24	30819	PA	2009/11/17T00:00:09.842	2.355
139.173.53.191	181.113.55.11	9429	80	6	1	40	R A	2009/11/17T00:00:12.299	0.000
181.113.55.11	184.199.238.201	80	33559	6	33	41996	FS PA	2009/11/17T00:00:12.866	1.222
184.199.238.201	181.113.55.11	33559	80	6	28	1878	FS PA	2009/11/17T00:00:12.866	1.222
181.113.55.11	250.227.45.82	80	54162	6	6	1713	FS PA	2009/11/17T00:00:14.773	30.566
250.227.45.82	181.113.55.11	54162	80	6	7	752	FS PA	2009/11/17T00:00:14.773	30.779
181.113.55.11	13.104.81.20	80	49931	6	2	104	F A	2009/11/17T00:00:19.577	0.000
13.104.81.20	181.113.55.11	49931	80	6	2	104	F A	2009/11/17T00:00:19.577	0.127

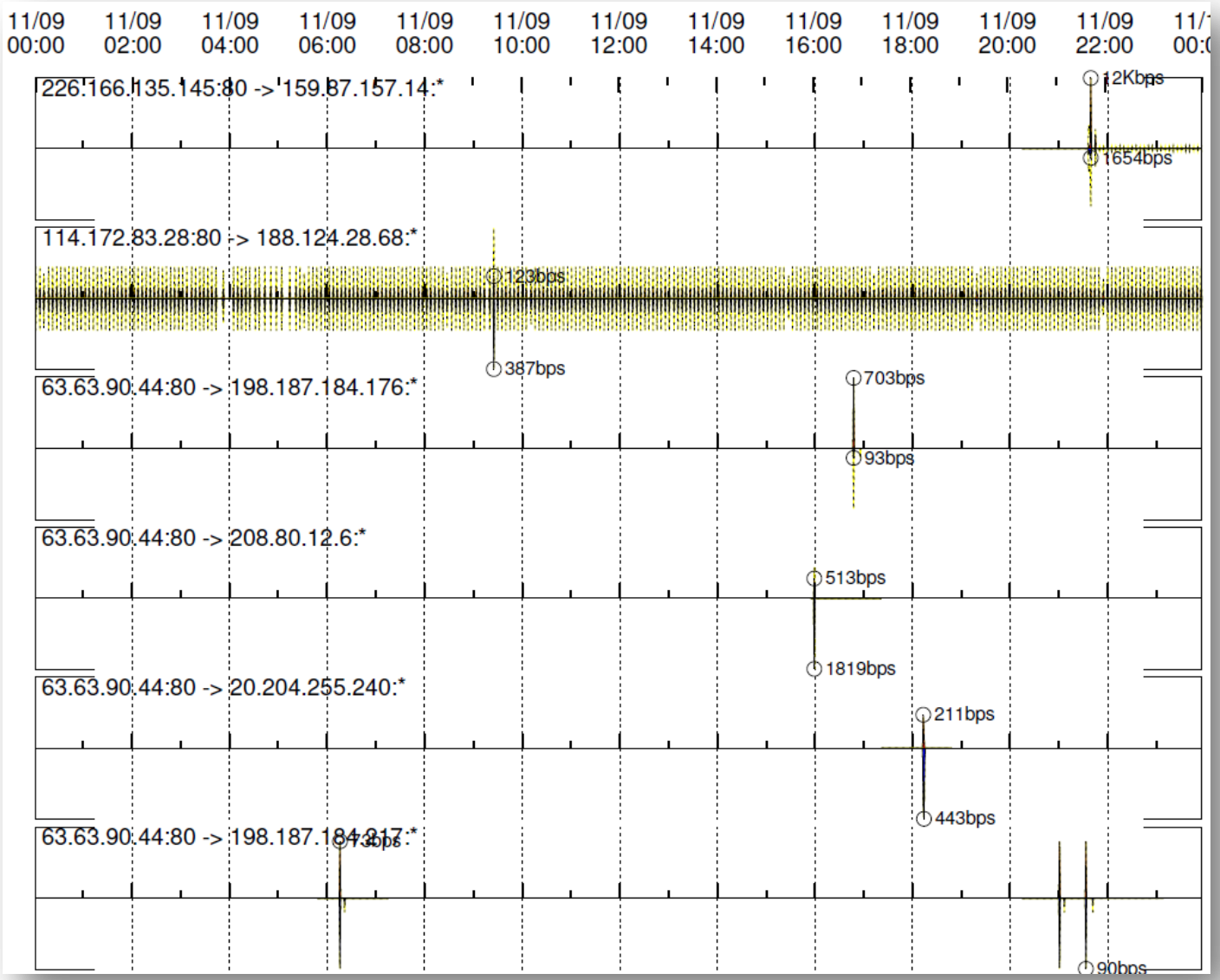


Sample Output

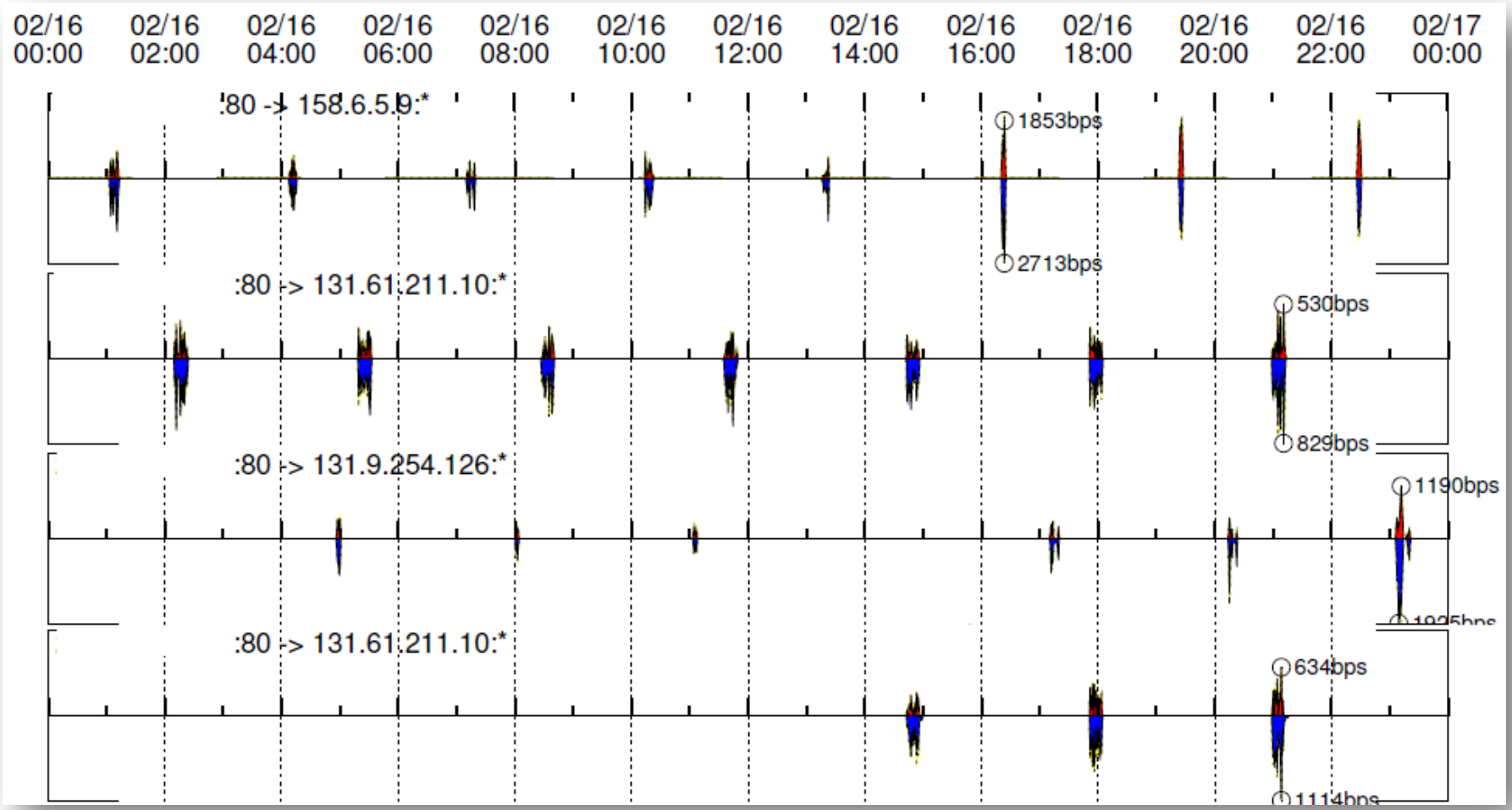
One Hour of Standard HTTP Traffic



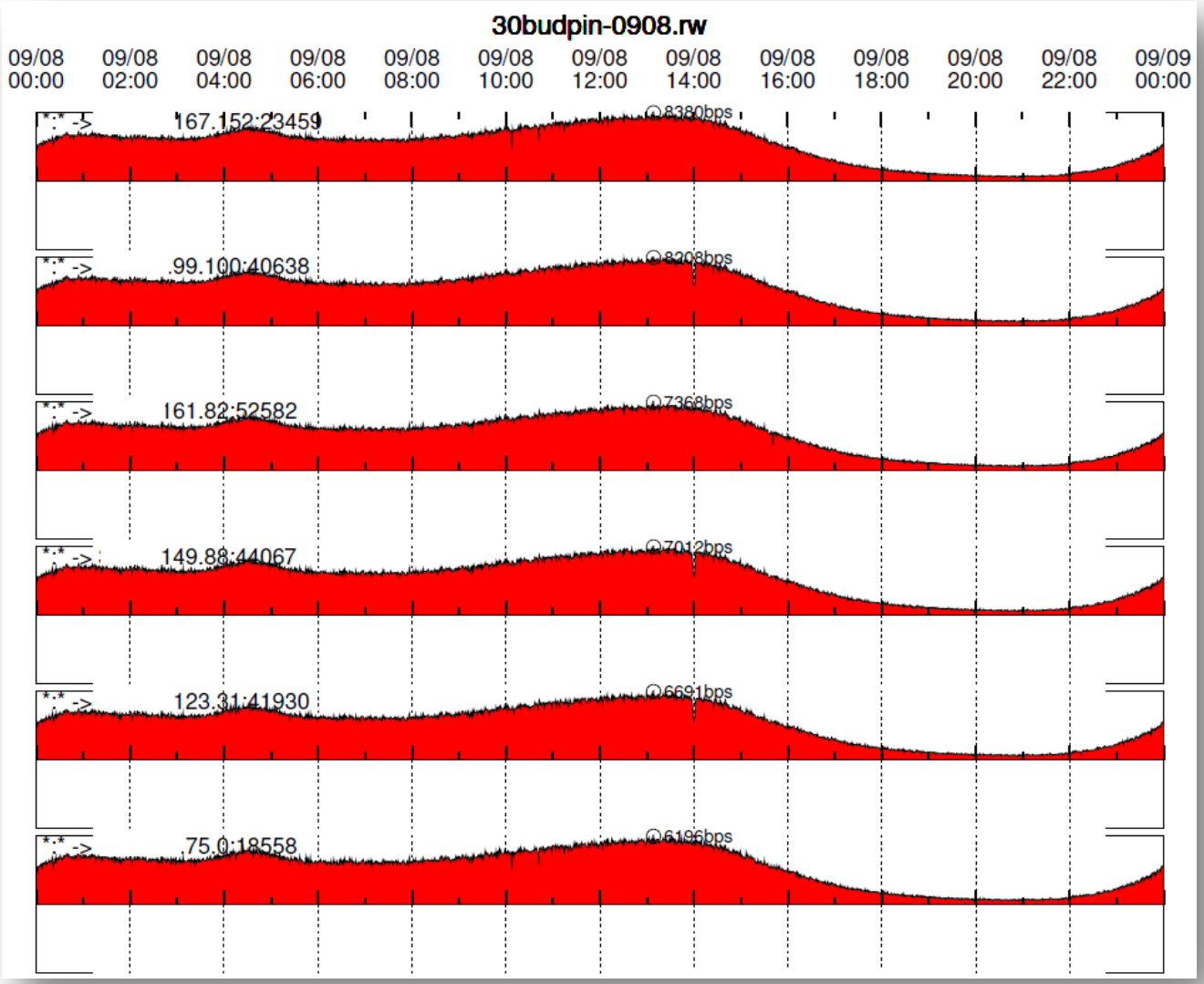
More http: can you spot the beacon?



Conficker

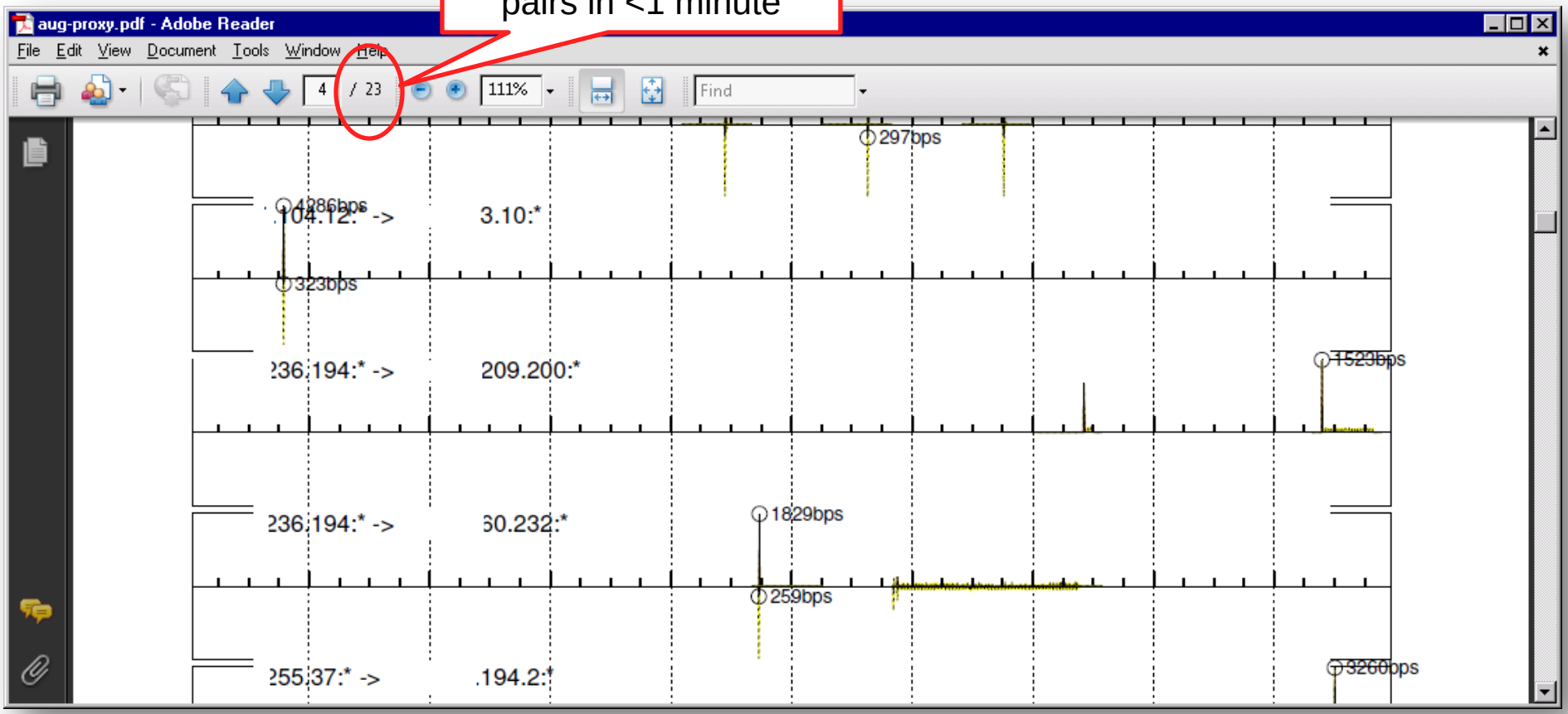


30-byte UDP packets



More web…

23 pages: Review traffic for 230 sip/dip pairs in <1 minute



The Basics

Basic script workflow

Find the top talkers

Count in & out traffic for each talker

Plot traffic for each

Compile plots into a single .pdf

Dependencies

SiLK toolset

gnuplot

ghostscript (gs)

Python (but not pySiLK)

./stripplot.py

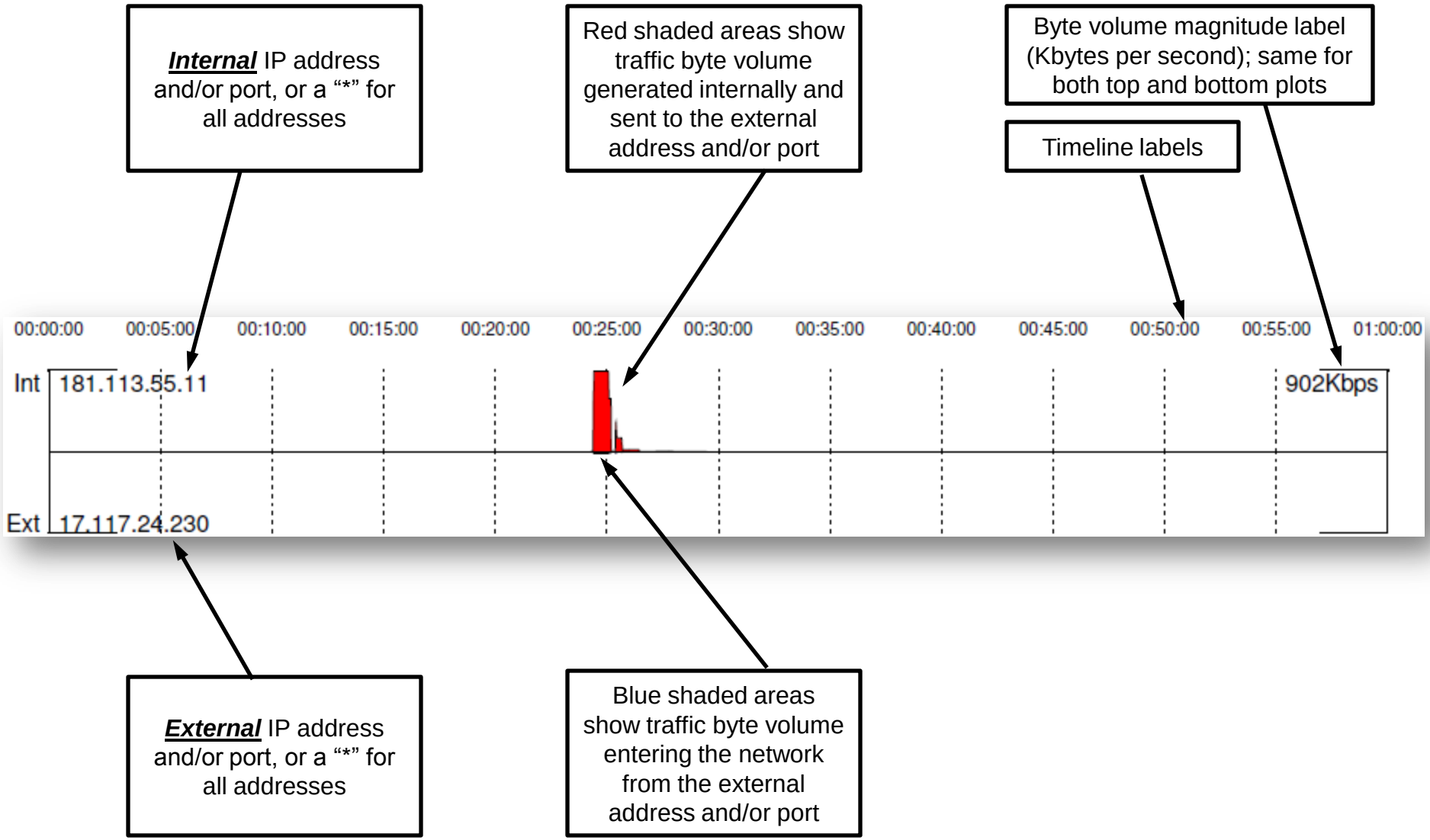
```
 $ stripplot.py
Usage: stripplot.py [options] FILE
Creates a strip-plot of the most significant traffic contributors
within a raw SiLK data file

  FILE   (required) The binary SiLK file to analyze

  Options:
  --binsize   Default counting bin size (seconds) (def=auto)
  --bottomleft Bottom left tag; allows %(substitution)s; use '-' for none (def
              =-)
  --bottommiddle Words on the bottom right of the page, allows %(substitution)
              s; use '-' for none (def=FOUO)
  --bottomright Words on the bottom middle of the page, allows %(substitution)
              s; use '-' for none (def=Page %(page)i of %(pagecount)s)
  --count     Number of plots to output (def=5)
  --endtime   Plot end time, YYYY/MM/DDTHH:MM:SS (def=auto)
  --fields    rwuniq-style list of fields to group IN traffic on, or '*' for a
              utomatic (NOT ALL FIELDS WORK) (def=sip,dip)
  --flags     Include this option to add a plot of TCP flags to the strips (de
              f=0)
  --help      Print this output (def=)
  --types     Inbound and outbound types; these are used to make sure the IN a
              ddress is on the top plot; must be in the form [in-type/out-type
              ]; unspecified types work fine but either address may end up on
              the top. (def=in/out,inweb/outweb,inicmp/outicmp)
  --pdffilepath PDF final output file (def=tmp.pdf)
  --plotfile  Temporary gnuplot script file to create (def=tmp.plot)
  --plotsperpage Number of plots per page (def=10)
  --prefilter rwfilter expression to apply to flow file before selecting what
              to plot.  NOTE:  this filter is NOT applied to the trends themse
              lves, only to the selection routine (def=--proto=0-)
  --psfilepath Post-script file to generate (def=tmp.ps)
  --selectionval Choose the top [count] combinations to plot based on this val
              ue; must be either 'bytes', 'packets', 'flows' or 'none'); if 'n
              one' then output is in rwuniq (random) order (def=bytes)
  --topleft   Words on the upper left of the page, allows %(substitution)s; us
              e '-' for none (def=-)
  --topright  Words on the upper right of the page, allows %(substitution)s; u
              se '-' for none (def=-)
  --trendline Highlighted and dotted trendline to add to plot; f for flows, p
              for packets (def=b)
  --starttime Plot start time in YYYY/MM/DDTHH:MM:SS format (def=auto)
  --maintitle Title for this plot, allows %(substitution)s; use '-' for none (
              def=auto)
  --verbose   Print out debugging info, use twice for more info and to print d
              ebugging info on the plot itself (def=0)

Fields with string substitution support the following:
  %(page)i    Current page number
  %(date)s    Date the report was printed
  %(time)s    Time the report was printed
  %(pagecount)i Total number of pages in the report
  %([setting])s Any of the report configuration settings
              (run with -v option to see settings
```

An individual strip

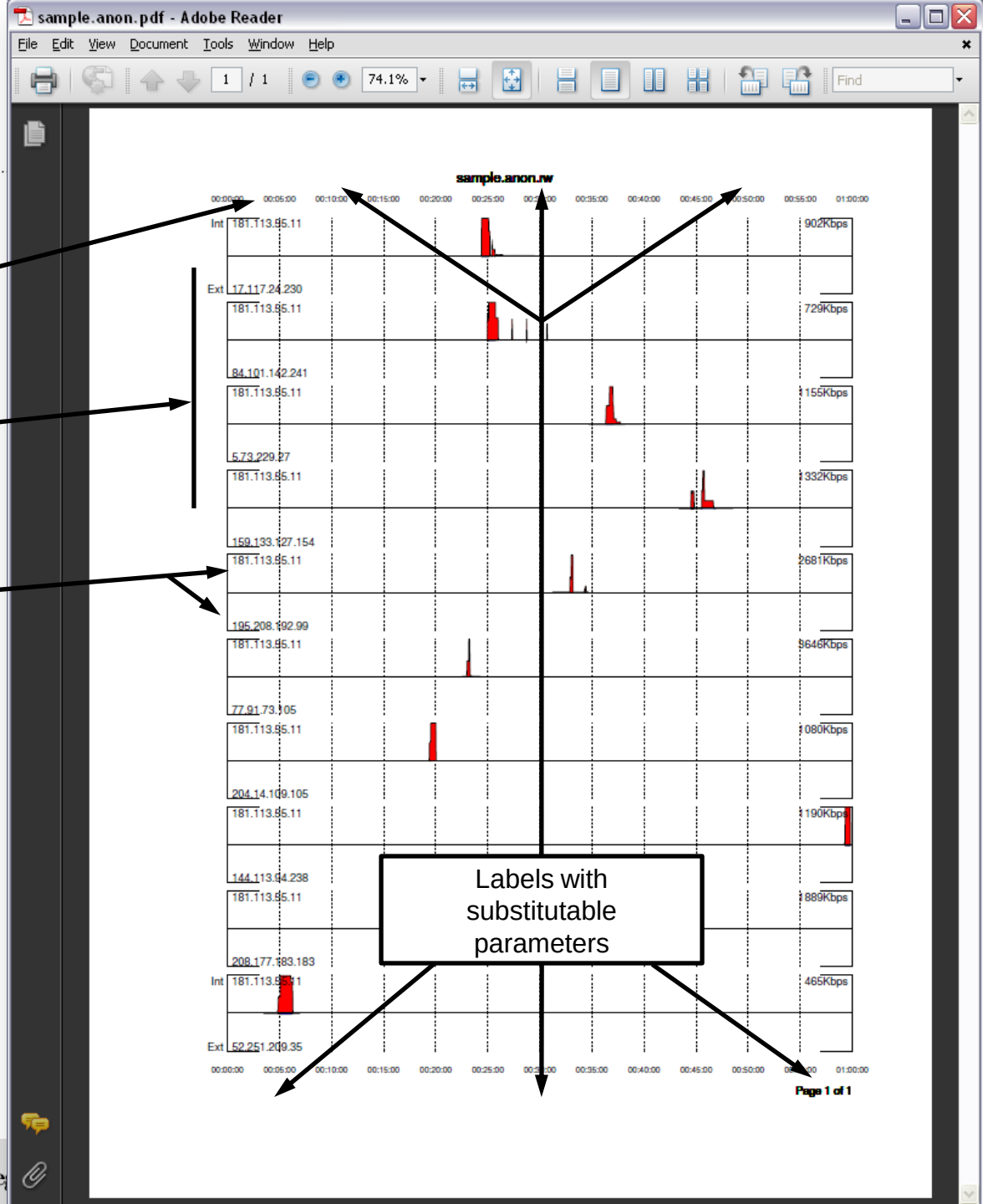


The Full Page

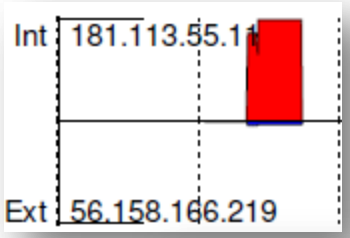
Time scale/formatting courtesy of gnuplot

Consistent timeline throughout

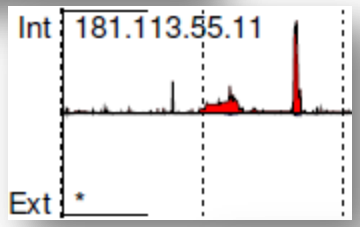
Each strip has an independent magnitude, but top and bottom are the same magnitude



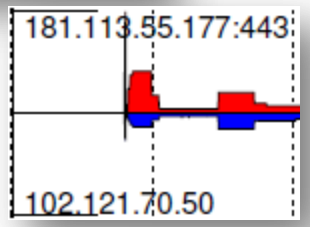
--fields=[rwuniq field set] defines groupings



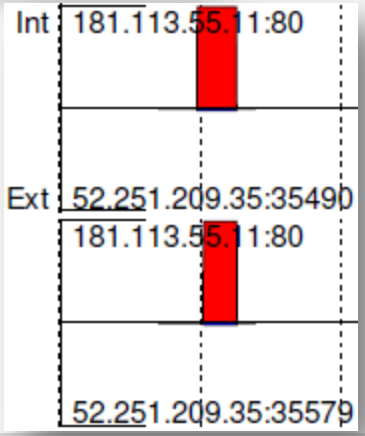
--fields=sip/dip (default): group on client/server pairs



--fields=dip: group on dest (my) address



--fields=sip,dip,dport: group on external services



--fields=sip,dip,sport,dport: group on TCP sessions

Q: How does it choose what strips to plot?

A: It uses rwuniq to group records, then chooses the largest by byte volume:

```
rwuniq [file] \
   --fields=[field list] \
   | sort -nr -k [bytes column]
```

Tweaking the selection criteria

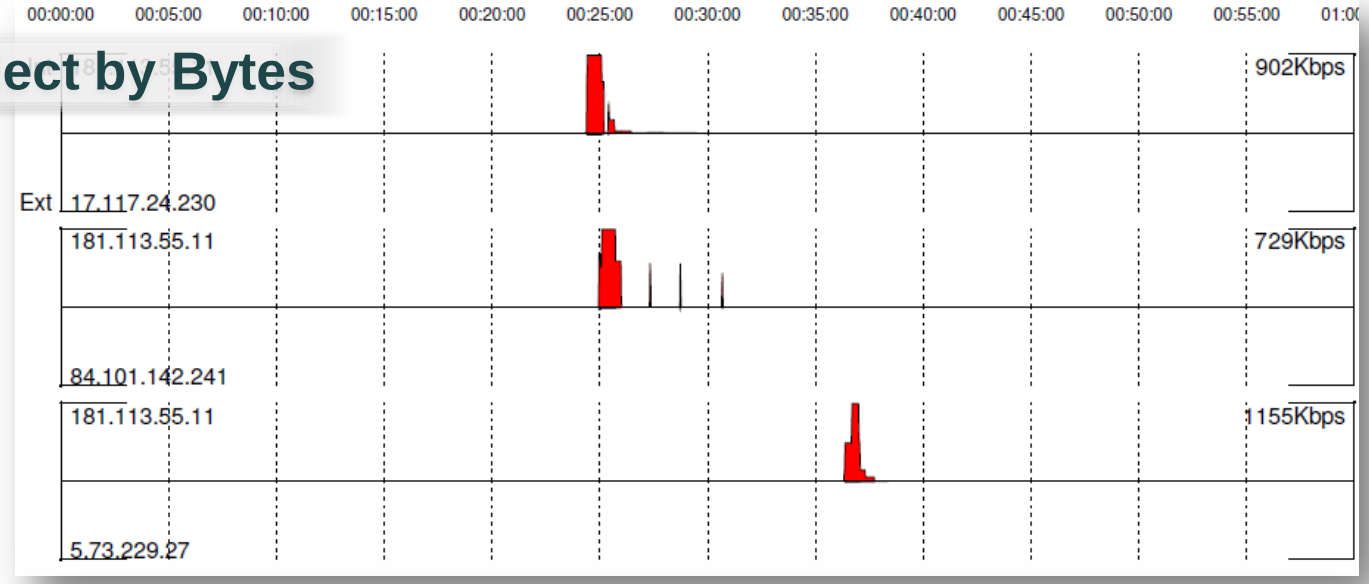
Sometimes you want to show the top packets or flows

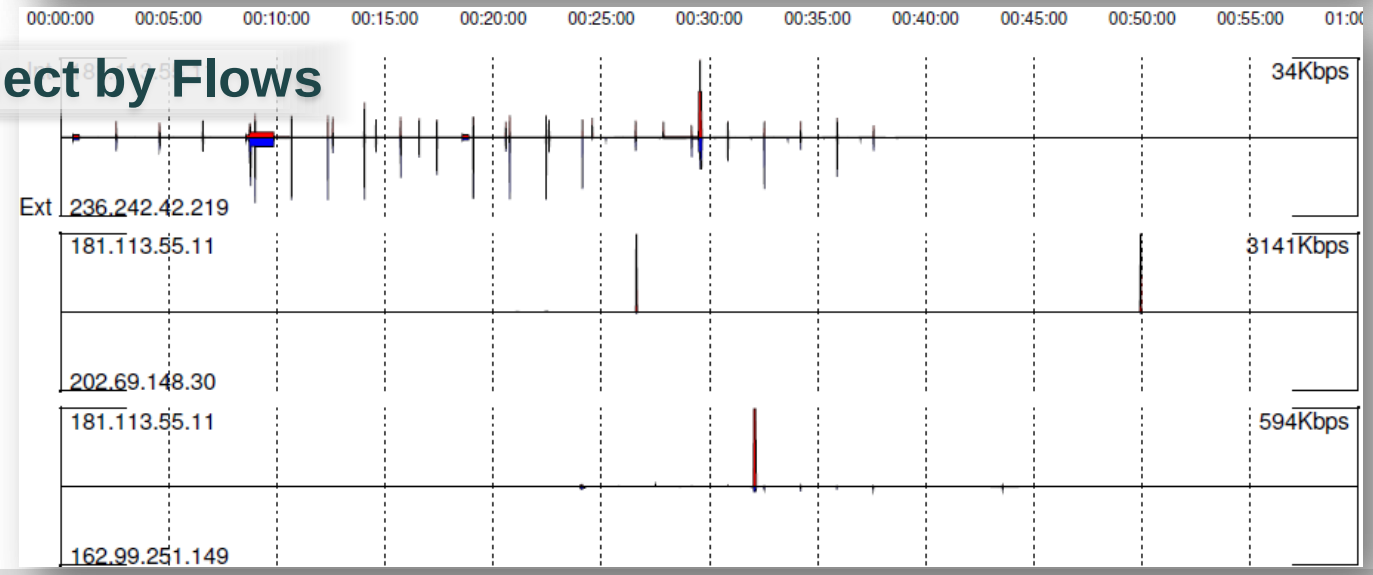
- Repeated failed connection attempts
- Bot phone-home

`--selectionval=flows|packets`

- Sort by top flow count or packet count

selectionval example





Other Common Options

--count

- Defines how many strips to plot, defaults to 5

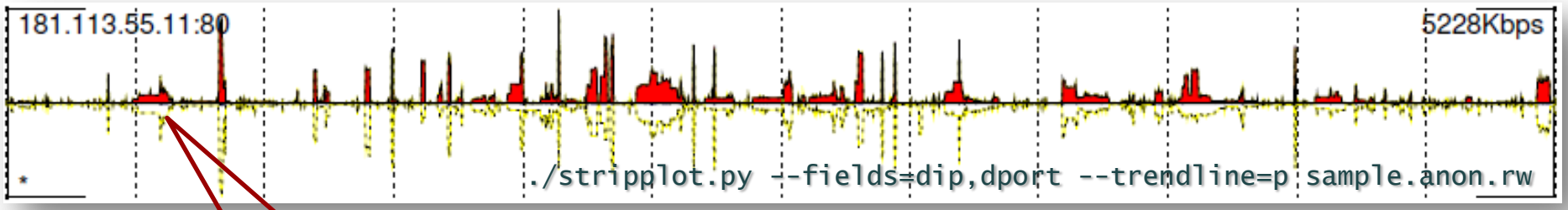
--prefilter

- Filter out the .rw file before plotting it

Trend lines

--trendline=f [or p]

- Adds a highlighted dotted line for flows (or packets)
- No labeling, no agnitudes, not symmetric
- Good for drawing out low-volume data

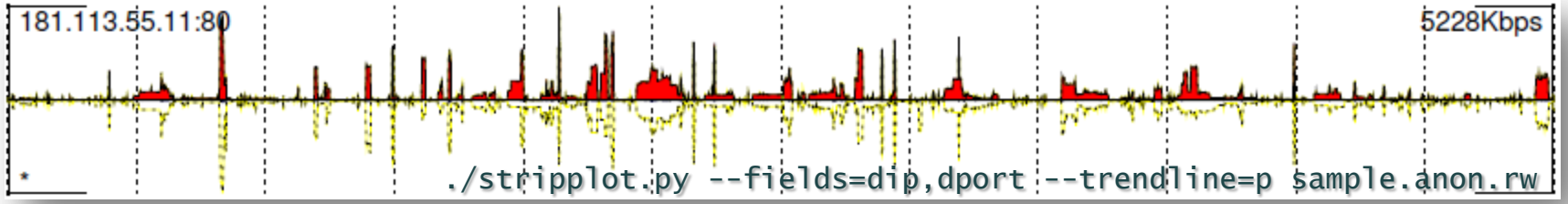


The trend line show ACK packet counts generally match data packets

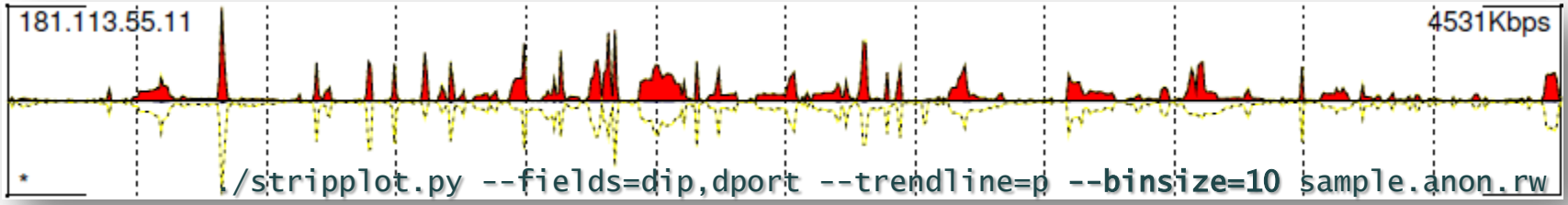
Bin size

Points on the time axis:

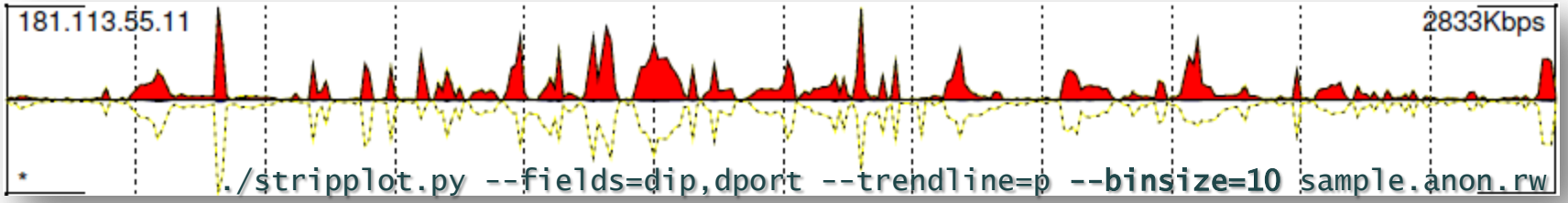
One hour defaults to 1 second bins (3600 points):



5 second bins makes little difference:

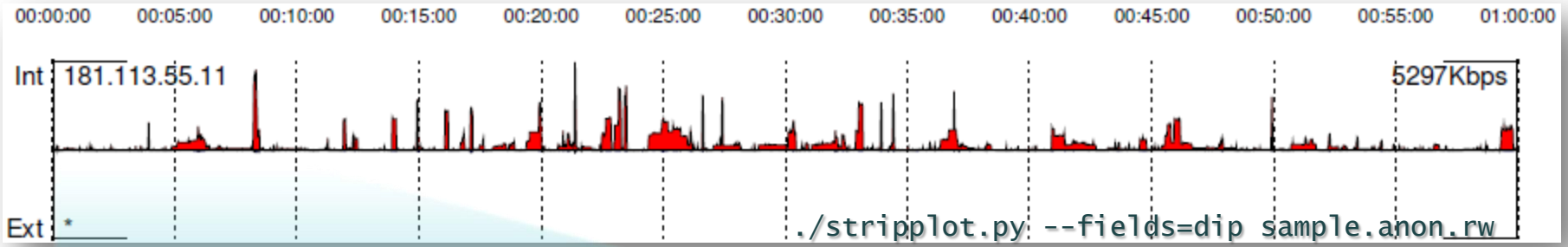


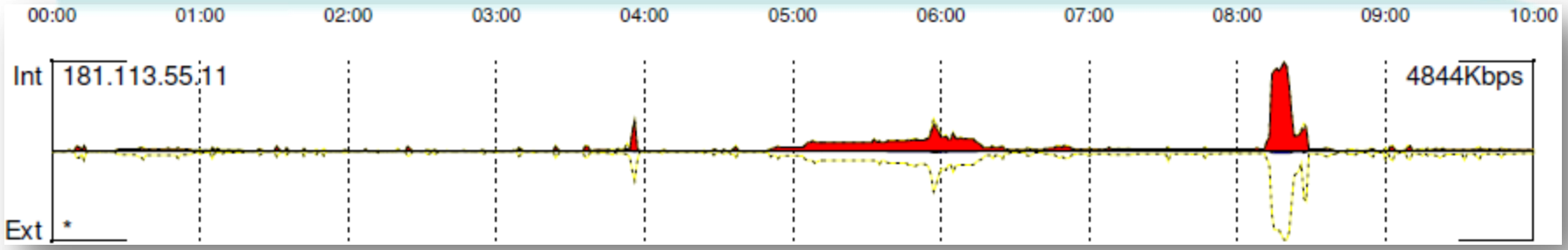
At 10 seconds (360 bins), the picture gets fuzzy:



Starttime, endtime

Zoom in on a particular time frame:






```
./stripplot.py --fields=dip --starttime=2009/11/17T00:00:00 --endtime=2009/11/17T00:10:00 \
                --trendline=p sample.anon.rw
```

Special Features

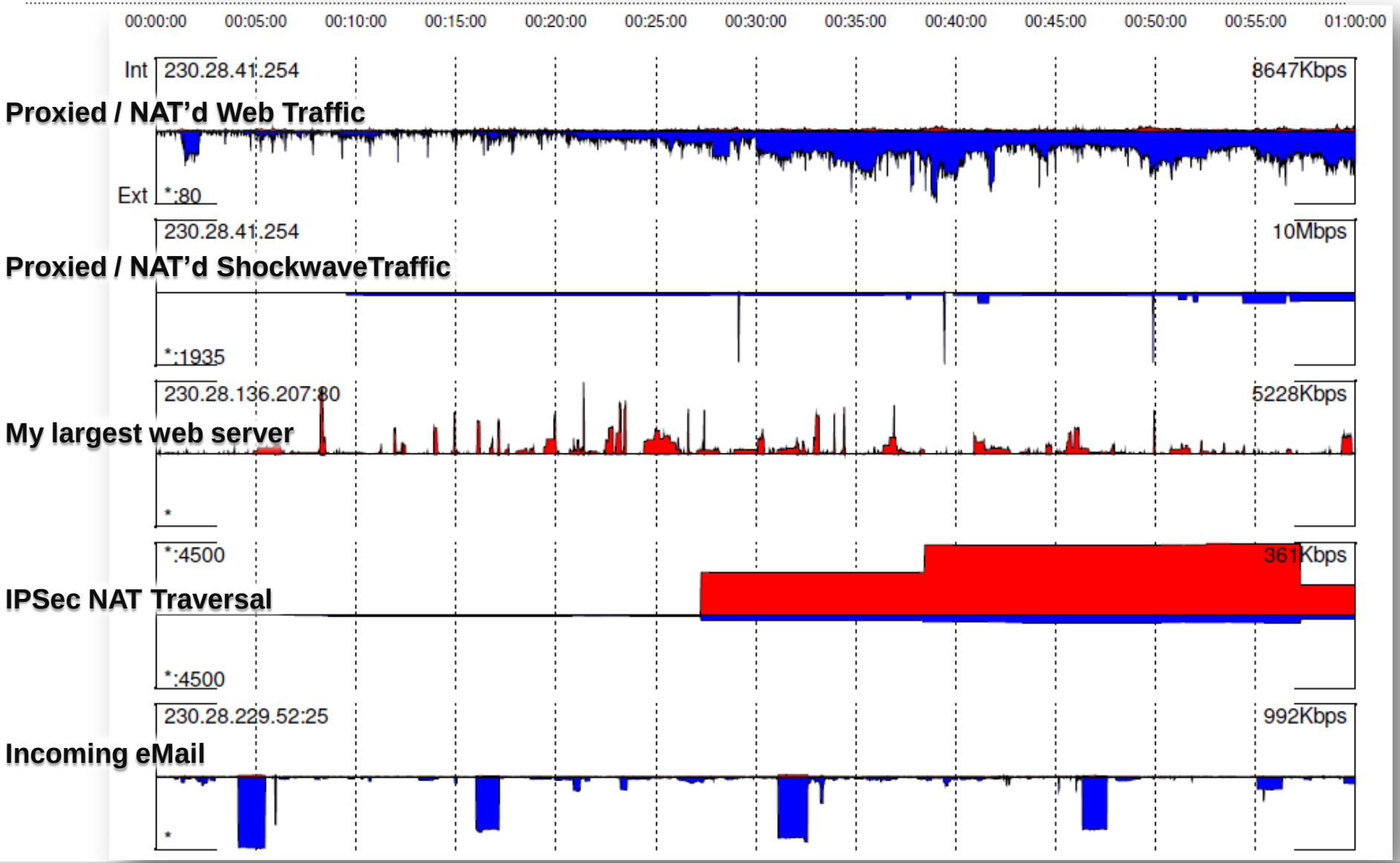
Automatically selecting fields

--fields=*

- Selects the best combination for 2, 3, 4 or 5 of sip,dip,sport,dport,proto
- For example, looking at all traffic for one network:

```
*:80 -> 230.28.41.254:*
*:1935 -> 230.28.41.254:*
*:* -> 230.28.136.207:80
*:4500 -> *:4500
*:* -> 230.28.229.52:25
45.178.111.132:61296 -> 230.28.41.74:56066
239.213.117.254:8080 -> 230.28.41.254:53474
*:* -> 230.28.229.75:25
45.178.111.132:50769 -> 230.28.41.74:55722
*:443 -> 230.28.41.254:*
```

Auto-select Example



Auto-select: how does that work?

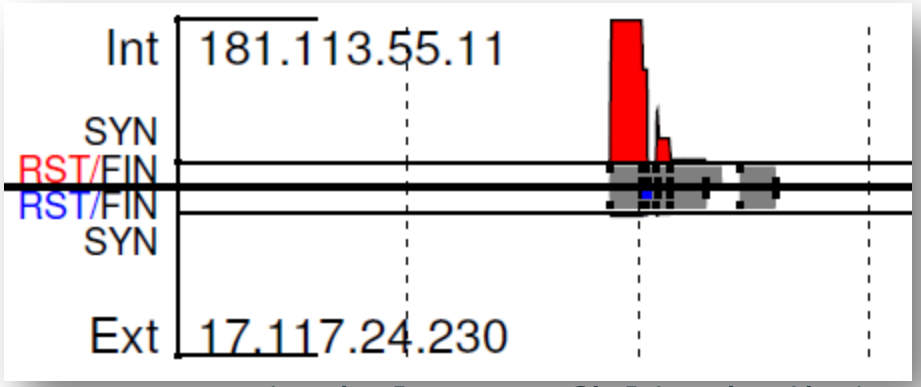
Simple, but slow. Find the top value for rwuniq on:

- sip,sport,dip,dport
- sip,sport,dip
- sip,sport,dport
- sip,sport
- sip,dip,dport
- sip,dip
- sip,dport
- sport,dip,dport
- sport,dip
- sport,dport
- dip,dport

Plotting TCP Flags

--flags

- Display points for TCP flags
- Works fine, just can't find a good generic use case



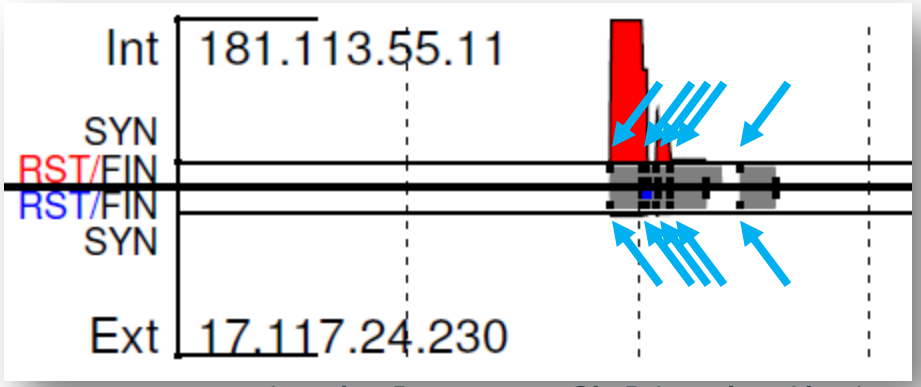

```
./stripplot.py --fields=sip,dip \
               --flags sample.anon.rw
```

Activity: Grey bar

Plotting TCP Flags

--flags

- Display points for TCP flags
- Works fine, just can't find a good generic use case



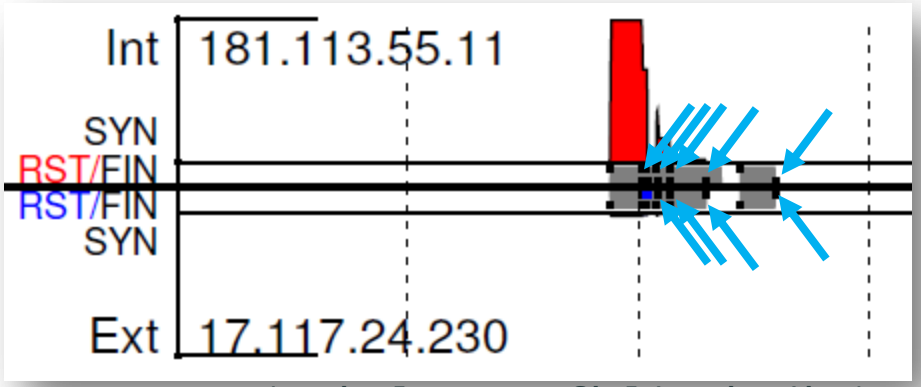

```
./stripplot.py --fields=sip,dip \
               --flags sample.anon.rw
```

SYN Packets (outside, black)

Plotting TCP Flags

--flags

- Display points for TCP flags
- Works fine, just can't find a good generic use case



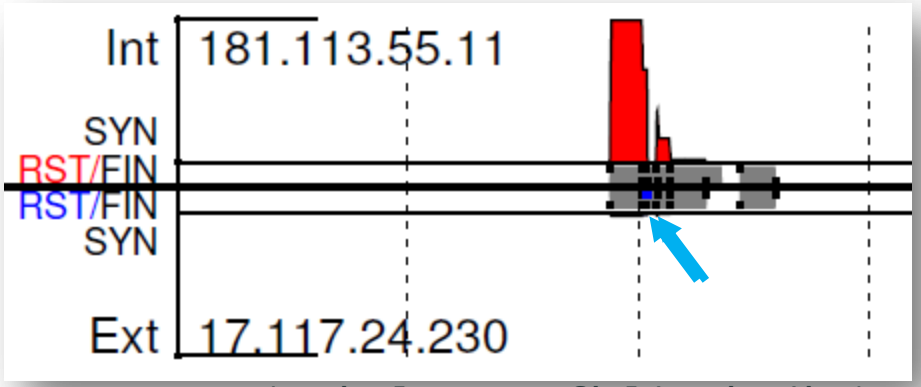

```
./stripplot.py --fields=sip,dip \
               --flags sample.anon.rw
```

FIN Packets (middle, black)

Plotting TCP Flags

--flags

- Display points for TCP flags
- Works fine, just can't find a good generic use case




```
./stripplot.py --fields=sip,dip \
               --flags sample.anon.rw
```

RST Packets (middle, blue or red)

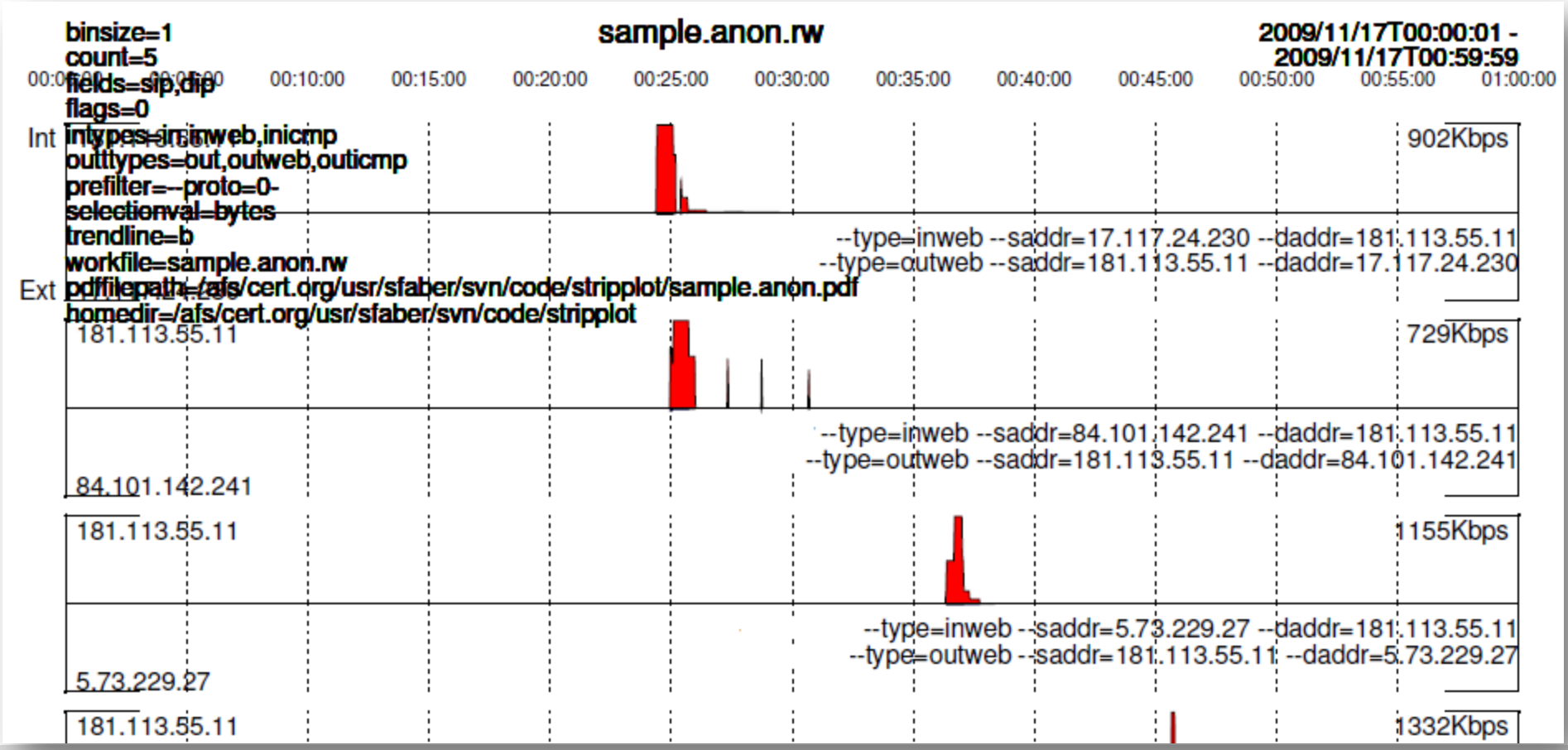
Verbose output, -v

```
$ ./stripplot.py --fields=sip,dip --flags -v sample.anon.rw
Found 17.117.24.230:* -> 181.113.55.11:*
Found 84.101.142.241:* -> 181.113.55.11:*
Found 5.73.229.27:* -> 181.113.55.11:*
Found 159.133.127.154:* -> 181.113.55.11:*
Found 195.208.192.99:* -> 181.113.55.11:*
#  Settings:
#  binsize    1
#  bottomMargin 0.08
...
#  usableWidth 0.9
#  verbose    1
#  workfile   sample.anon.rw
(000): Page 001, Plot 000
(001): Page 001, Plot 001
(002): Page 001, Plot 002
(003): Page 001, Plot 003
(004): Page 001, Plot 004
```

Very Verbose Output, -vv

Echos all rw* commands

Adds lots of info to the .pdf output



Overriding default types

--types=in/out,inweb/outweb

- It's OK if the type doesn't actually exist (ie, multiple installations)
- Has to match rwcut type field

Open Issues

Better error checking

- Most inputs are passed directly to the rw* tools
- Occasional errors trying to plot empty data sets

Doesn't work well for transit traffic

- Assumes "in" and "out" traffic

Problems if you have outbound traffic only

- Trend selection doesn't seem to work well

TCP flags

- Tech is there, but visualization needs lots of improvement

Strip Plots: A Simple Automated Time-Series Visualization

Sid Faber
sfaber@cert.org