ETH zürich

Inline Data Integrity Signals for Passive Measurement

Brian Trammell, David Gugelmann (CSG ETH Zürich), and
Nevil Brownlee (Univ. Auckland)

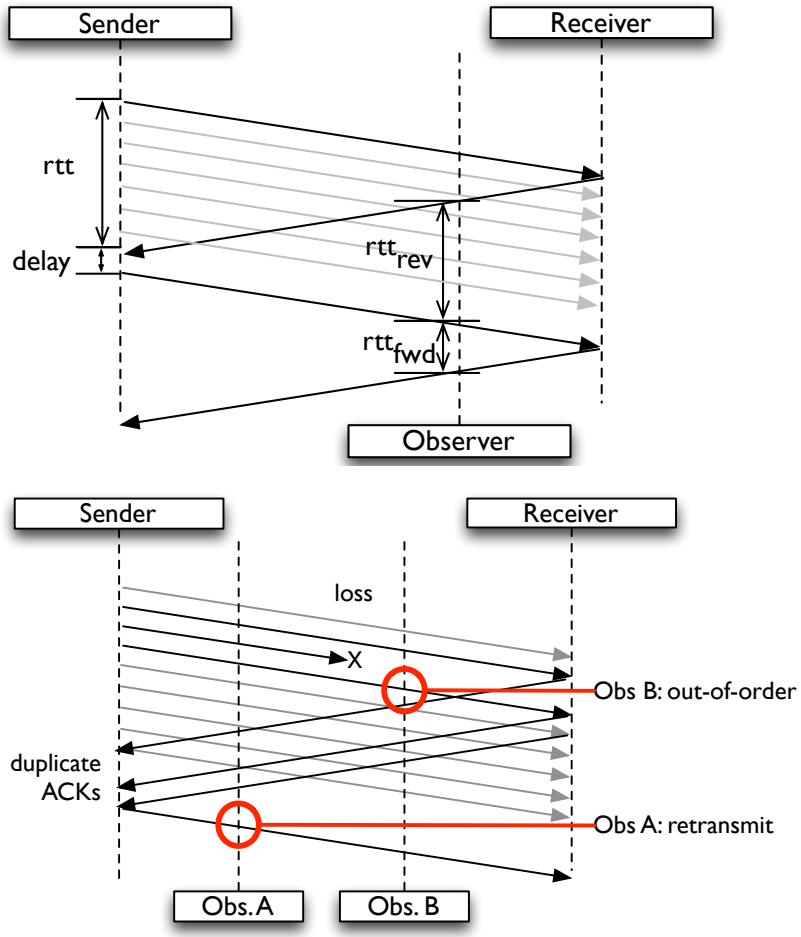
At TMA 2014, 14 April 2014, London, England

mPlane

…or, never make the mistake of thinking you're measuring what you think you're measuring!

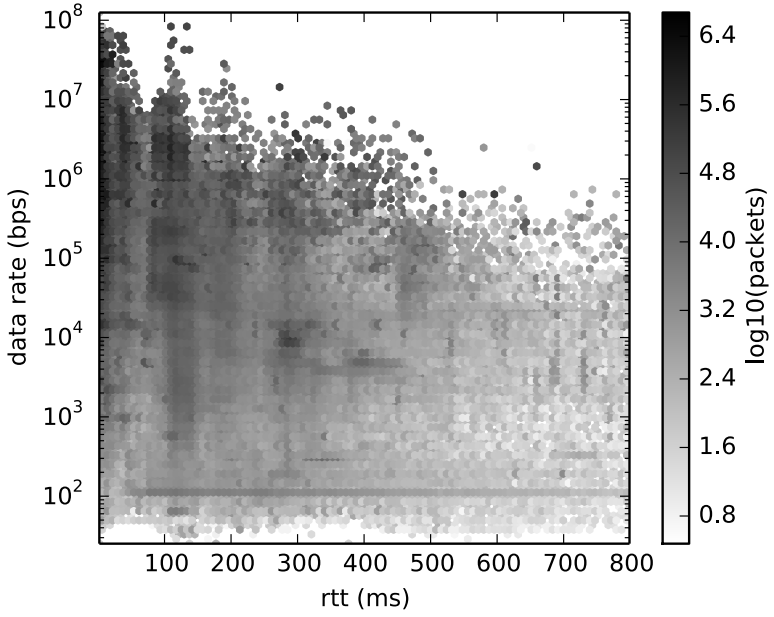
- The history of discovery of erroneous assumptions in network measurement is long and varied.
  - Augustin et al (SIGCOMM 2006): traceroute is broken.
  - Cunha et al (PAM 2009): flow durations often wrong.
  - Trammell et al (PAM 2011): timing broken in NetFlow *by design*.
  - Hofstede et al (PAM 2013): actually, all flow export is suspect.
  - Pelsser et al (IMC 2013): and even ping doesn't work. ☹
- Are there features of the observed data in passive measurement we can use to check our assumptions?
- Assumption #1: we see all the packets.

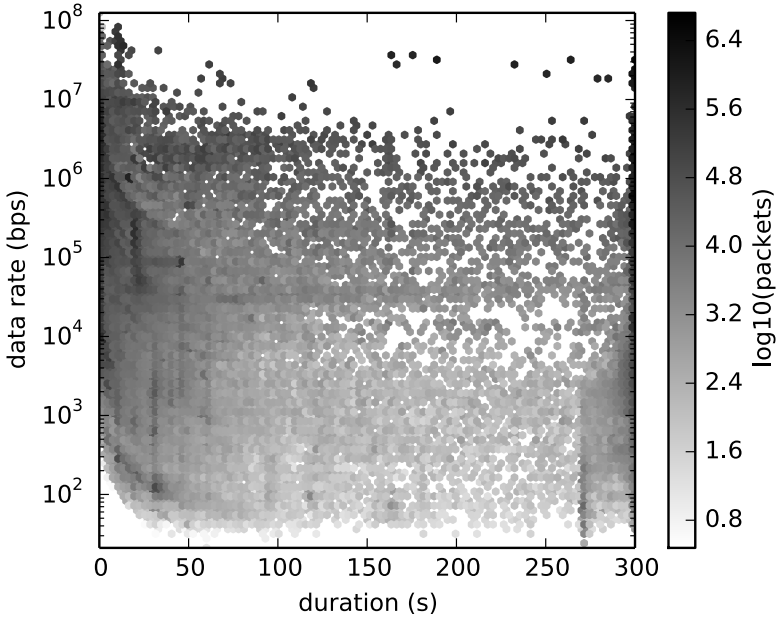
QoF ("Quality of Flow"), a passive transport performance meter



- Passive TCP performance monitor and IPFIX exporter
  - Estimated sender-observed RTT
  - Midpoint loss event detection
- **GPL, github.com/britram/qof**
- Designed for performance and observation point independence
- RTT via biflow SEQ/ACK and TSval/TSecr measurement
- Transport loss via RTX and sequence number jump
- *Needs to see every packet*

But what is it good for?

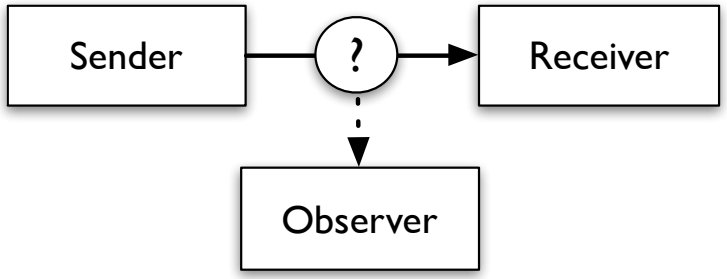




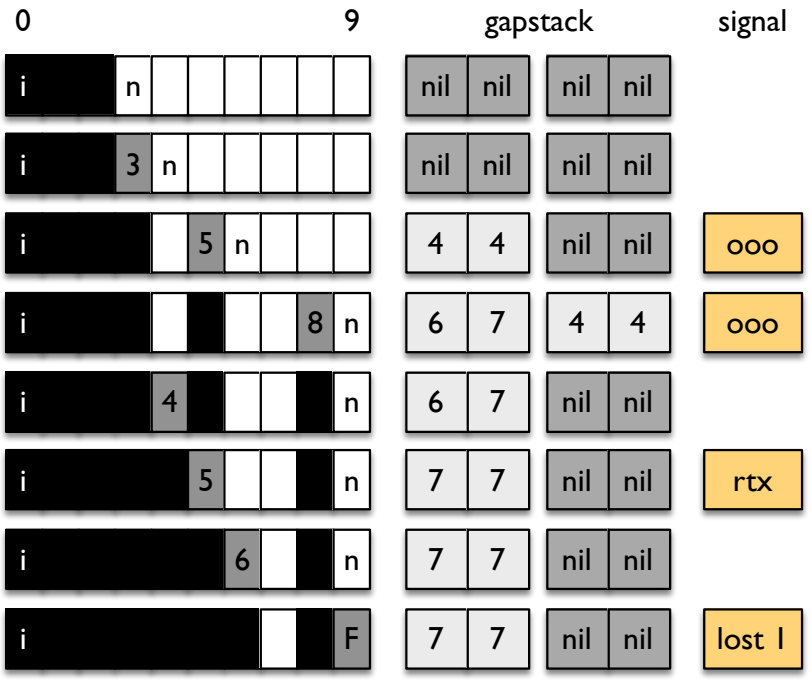
Generated from 3h of MAWI trace data (30 Mar 2013)…
… in 3m (+ 5m of postprocessing)

Observation Loss in Passive Measurement

- Packets that made it from the source to the destination, but not to the observation point:
  - Capture device buffering
  - Optical tap errors/packet drop
  - Span port queue drop
- Currently: per interface count
  - Only for loss at OP device
- In TCP, the destination behavior depends on whether it received a given packet or not.
- We can use this to deduce loss, *on a per-flow basis.*



Measuring Observation Loss in QoF



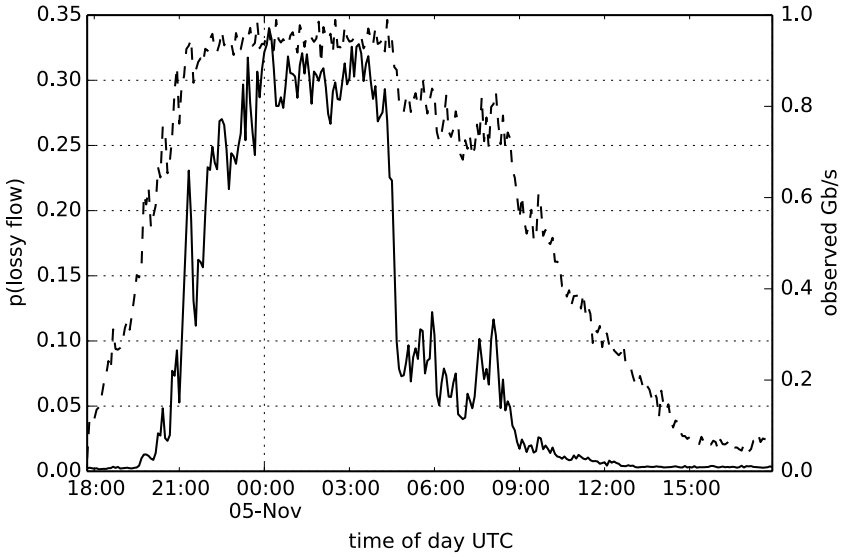
- Sequence numbers tracked using a *gap stack*: list of ranges of unseen sequence numbers
- Sequence numbers seen before expected → jump
- Sequence numbers seen more than once → retransmission
- (These two together indicate transport loss)
- Sequence numbers that remain unseen → observation loss

Properties of per-flow observation loss estimation

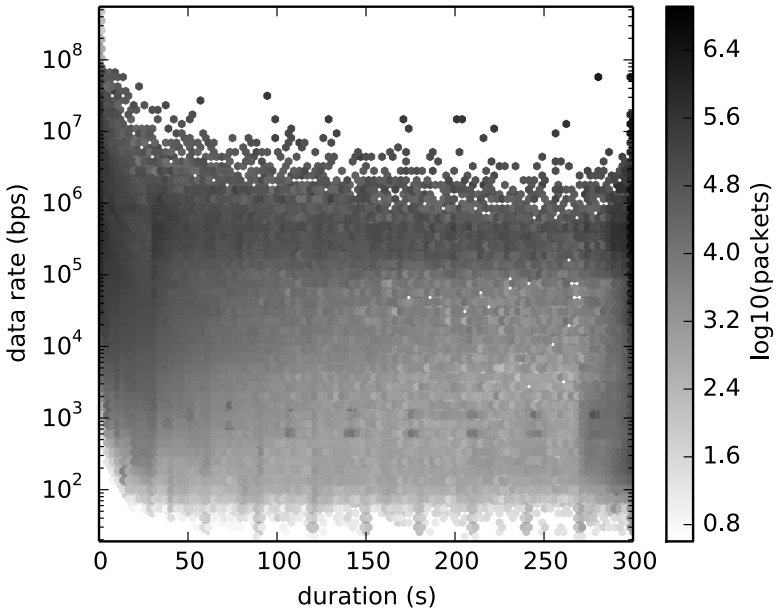
- Evaluated against induced loss on MAWI traces
  - QoF has a built-in leaky bucket (`--detune` options)
  - Sensitive to very low loss rates (<1:1M)
- Observation loss also leads to unobserved flows
  - Flow counts are dominated by short flows
  - In MAWI, 1:4 lossy:unobserved ratio
- Observation loss measured in TCP flows will indicate observation loss in non-TCP flows with the same property
  - May allow localization to paths/upstream packet treatment
  - ...assuming observation loss is protocol independent

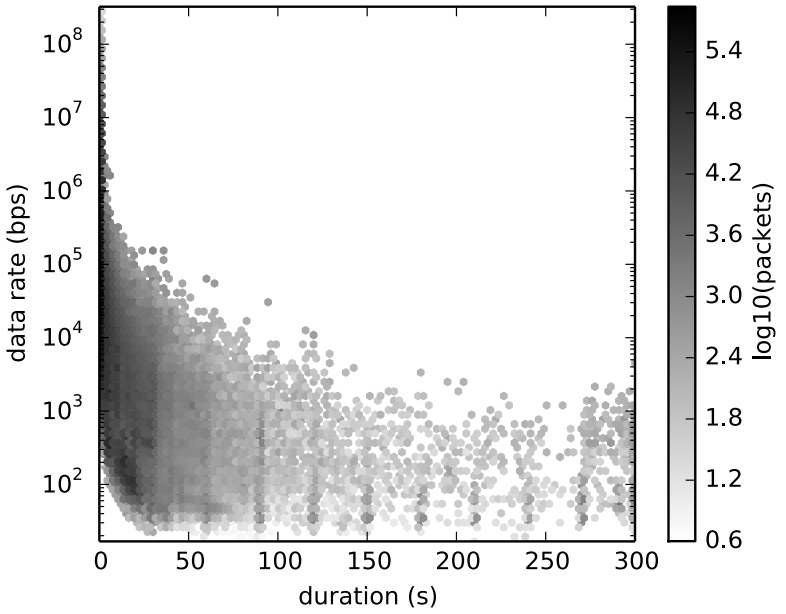
At the University of Auckland

- Collection from campus border upstream and downstream links pushed over span port
- Originally: 150M up, 150M down, 1G span
- 2013: 1G up, 1G down (700M peak ea. direction), 1G span
- Result: >30% observation loss
- (In progress: dedicated measurement taps)



Lossy vs. lossless flows, packet density by data rate and duration, Auckland, 5 November 2013





Evaluation of Horizon Extender

- Data leakage forensics system[1] evaluated on 1.2TB full packet trace data.
  - Like QoF, requires all packets: did we get them?
  - Applied QoF to generate *just* observation loss statistics.
- Result: close enough. 0.02% of flows overall with at least one observation loss, load dependent.
- Meta-evaluation against Bro stream reconstruction engine
  - ~ 0.001% of flows (up to 20x overcounting in QoF if Bro correct)
  - …but took 25 times as long to verify (730s/day vs. 29s/day)
  - …and on further review, appears to undercount observation loss.[2]

[1] Gugelmann, D., Schatzmann, D., Lenders, V.: "*Horizon Extender: Long-term Preservation of Data Leakage Evidence in Web Traffic.*" In *Proceedings of the 8th ACM SIGSAC symposium on Information, Computer and Communications Security*, Hangzhou, China (2013)
[2] http://mailman.icsi.berkeley.edu/pipermail/bro/2014-January/007275.html

Principles for further research

- Inline observation loss measurement for TCP works
- Less-than-100% source data is the *rule*, not the exception
  - So let's stop hiding information about this in the debug logs.
- Data quality is essential metadata
- Inline export makes it useful
  - TCP: observation loss
  - IP: per-interface loss/error at high frequency?
  - other layers: other hacks
- Look for opportunities to hack your tools to give you more information about the quality of your data sources!