Trumpet
Timely and Precise Triggers in Data Centers
by Masoud Moshref, Minlan Yu, Ramesh Govindan (USC) and Amin Vahdat (Google)

Alexander Hedges 16.3.2018
Why Monitoring?

- Know how much to bill the customer
- Detect attacks and failures
- Detect unbalanced loads
- Determine causes of congestion
- If done on short timescales can be even more useful
Main ideas of the paper

• Count every packet
  ◦ This can be achieved by moving the monitoring to the end host
  ◦ The endhost has to process the packet anyway

• Automated response on short timescales

• The two are coupled:
  ◦ Making the collection period short decreases memory requirements, counting every packet makes the automatic triggers more useful
• Packet counting is done at the end host before they enter the vm

• The controller configures the end hosts and is notified when the events take place
Events

• An event is defined by a packet filter and a predicate

• The packet filter specifies the type of packets (flows) that count towards the event
  ◦ Can be specified in a variety of ways (5-tuple, DstIp/24 etc.)

• The predicate specifies the condition at which an action should be triggered
  ◦ Allows operations such as min, max, etc.
Trumpet Packet Monitor (on the End Host)

• Needs to match the packet and evaluate the predicate at the end of the time interval

• The naive implementations of either matching the packet on arrival or when evaluating the predicate don’t perform sufficiently well

• For this reason the TPM first matches each packet to its flow and in the second phase collects the relevant packets and computes the predicate
TPM - Optimizations

• Maintains an extra index to map from 5-tuples to triggers

• Uses buffering to be able to interleave accepting packets with doing the gathering sweeps

• Uses prefixing to increase cache coherence

• And a few more

• To not collapse under DoS attacks, very small flows are ignored when unfinished sweeps are detected
Trumpet Event Manager (Controller)

• It installs the necessary triggers on the TPMs
• Upon receiving a satisfaction message from the TPM it queries other hosts to check if the event really occurred
• Some triggers can also be installed conditionally as responses to events
Evaluation

• Running simulated network traffic on a 40G NIC:
  ◦ No packet was dropped *
    * The minimum packet size for 40G was 650 bytes and the minimum time granularity was 10ms

• Most optimizations were vital to not dropping packets

• Resource usage is proportional to the traffic rate and the monitored traffic rate
Evaluation (continued)

- The packet processing time increases linearly with the amount of patterns to match

- The authors determined a feasibility region for their configuration:

  **Feasibility region** The different lines denote time intervals (in ms). The tests are for 300 (first image) 600 (second image) flows.

- They also tested trumpet under attack loads (see next slides)
Evaluation - Pictures

**Sweep and quiescent time**
Sweep time linearly increases with percentage of matched packages irrespective of load. CPU idle time decreases with increased packet throughput.

**Trigger matching performance**
Matching is almost independent of trigger amount but depends on the amount of patterns. The packet matching times are proportional to the amount of patterns.

**DoS threshold**
Blue is a syn attack (attacks phase 1). Green is a threshold attack (attacks phase 2). Trumpet is fairly good at estimating its own performance.
Conclusion and Outlook

• The paper shows the realization of per packet network monitoring

• It demonstrates the use of automated responses to network events

• To scale beyond 40G or very small packets TPMs can be run on different cores and there can be "local" TEMs to bundle requests

• The paper also proposes sharding to increase TEM performance
My 2 cents

• Very readable paper
• Paper strikes nice balance between new ideas and optimization
• The results don’t make any comparisons to other systems
• One should note that this solution is specific to data centers where the hosts are under the control of the data center owner
Thank you