NetFlow: Information loss or win?

Robin Sommer and Anja Feldmann
{sommer,anja}@in.tum.de

Technical University of Munich, Germany

Data sources for network monitoring

- SNMP
  - Coarse-grained, low-volume
  - e.g. Byte counts per router interface
- Packet Traces
  - Fine-grained, high-volume
  - e.g. TCP connection summaries
- Flow-level data
  - Medium-grained, medium-volume
  - Where does it fit in?

Cisco's NetFlow

- FLOW: Uni-directional stream of packets
- Router aggregates packets matching on $(src_{ip}, src_{port}, dst_{ip}, dst_{port}, \ldots)$
- NetFlow records contain among others
  - Source and destination
  - Start and end time
  - Number of bytes
  - TCP flags (cumulative OR)
- Problem: When does a flow end?

Packet traces vs. NetFlow

- TCP connection summaries from packet traces
  - 5355.558 0.346 6346 1283 152 222 A.B.C.D E.F.G.H SF
- NetFlows are
  - uni-directional
  - already aggregated
  - time-constraints only informaly defined
- Can we construct summaries based on NetFlow?

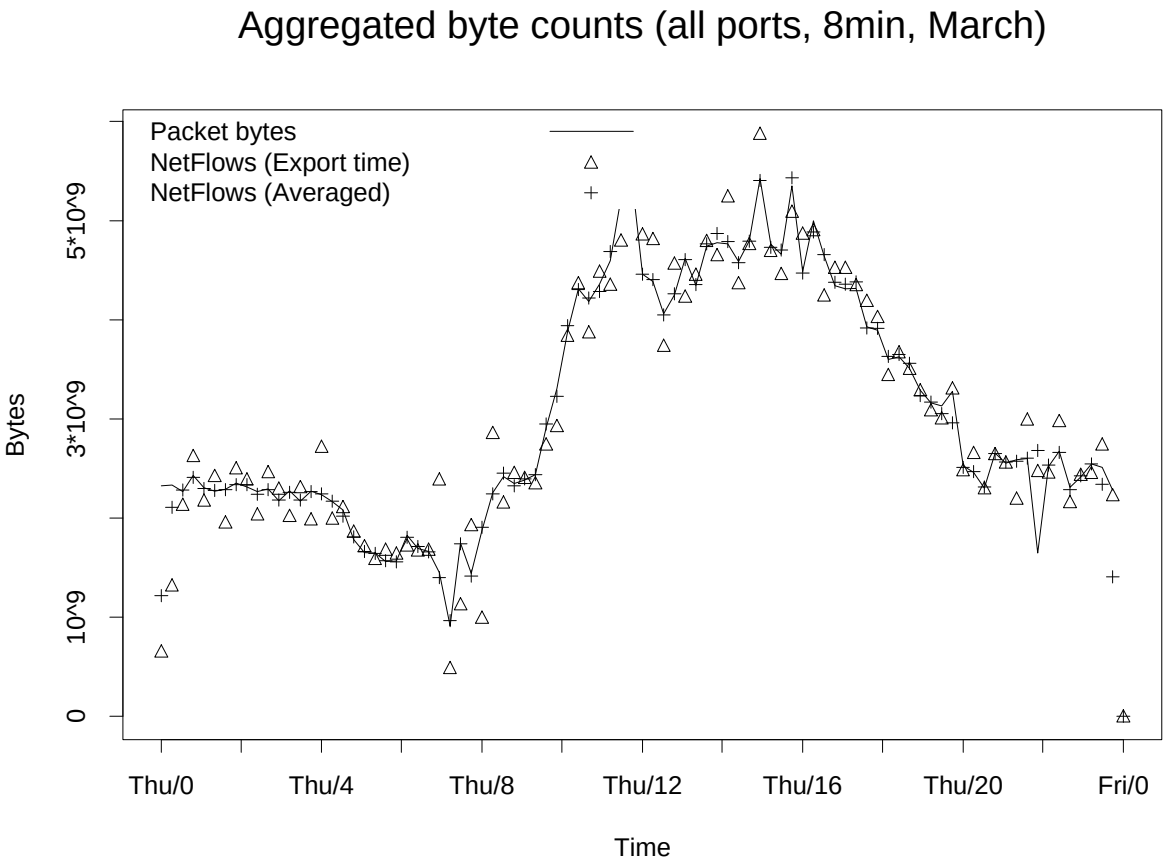
Flow-Reduce

- Identifies NetFlows belonging to the same connection
- Derives originator, duration and TCP state
- Evaluation by comparing to packet-based summaries
- Performance
  - Very well for over 90% of the connections
  - Best for Web traffic
  - Worst for Gnutella due to its connection setup

SNMP vs. NetFlow

- Accounting via SNMP
  - Transfered volume per interface
  - Queries byte counts every n minutes
- Accounting via NetFlow
  - Transfered volume per subnet, AS, protocol, etc.
  - How do we distribute NetFlow bytes over time slots?
  - Usual answer: Put all bytes into slot of export time
  - Better answer: Average bytes over slots in $[start, end]$

Example: Aggregated byte counts



Summary

- NetFlow is suitable for approximating
  - TCP connection summaries
  - SNMP byte counts, but in more detail
- Care has to be taken nevertheless
- Future work
  - Use test-bed router to understand NetFlow generation
  - Extend work to other flow models