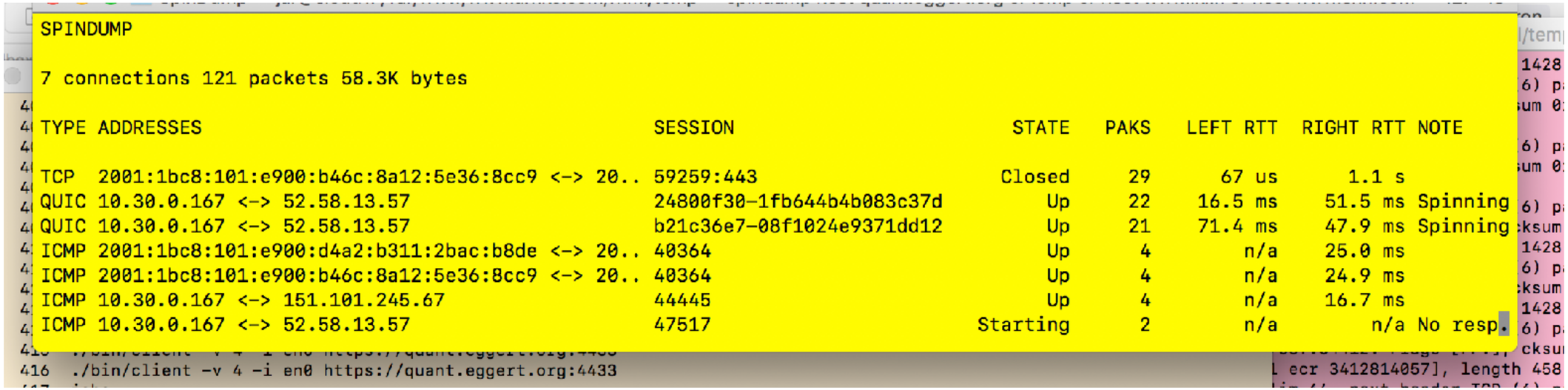
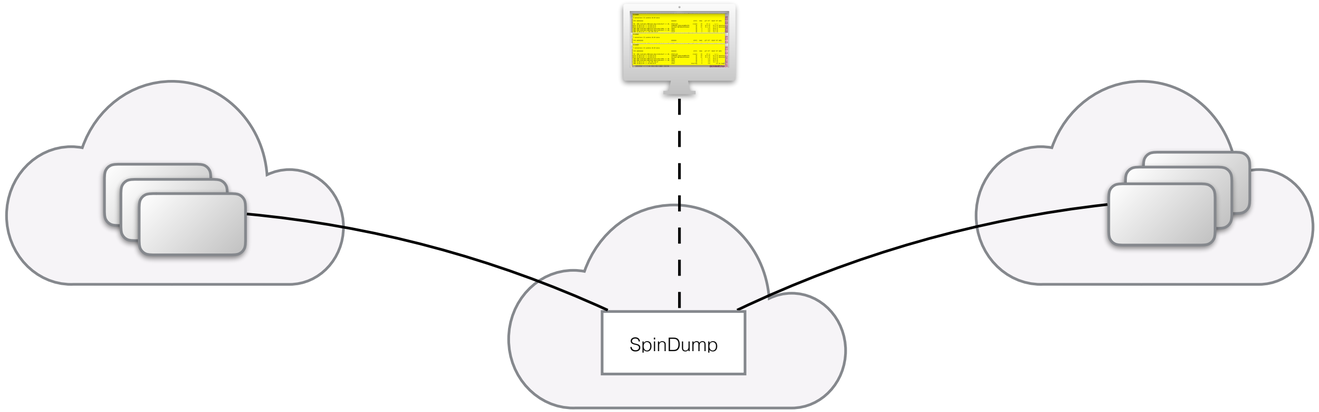
QUIC Logging
The In-Network View



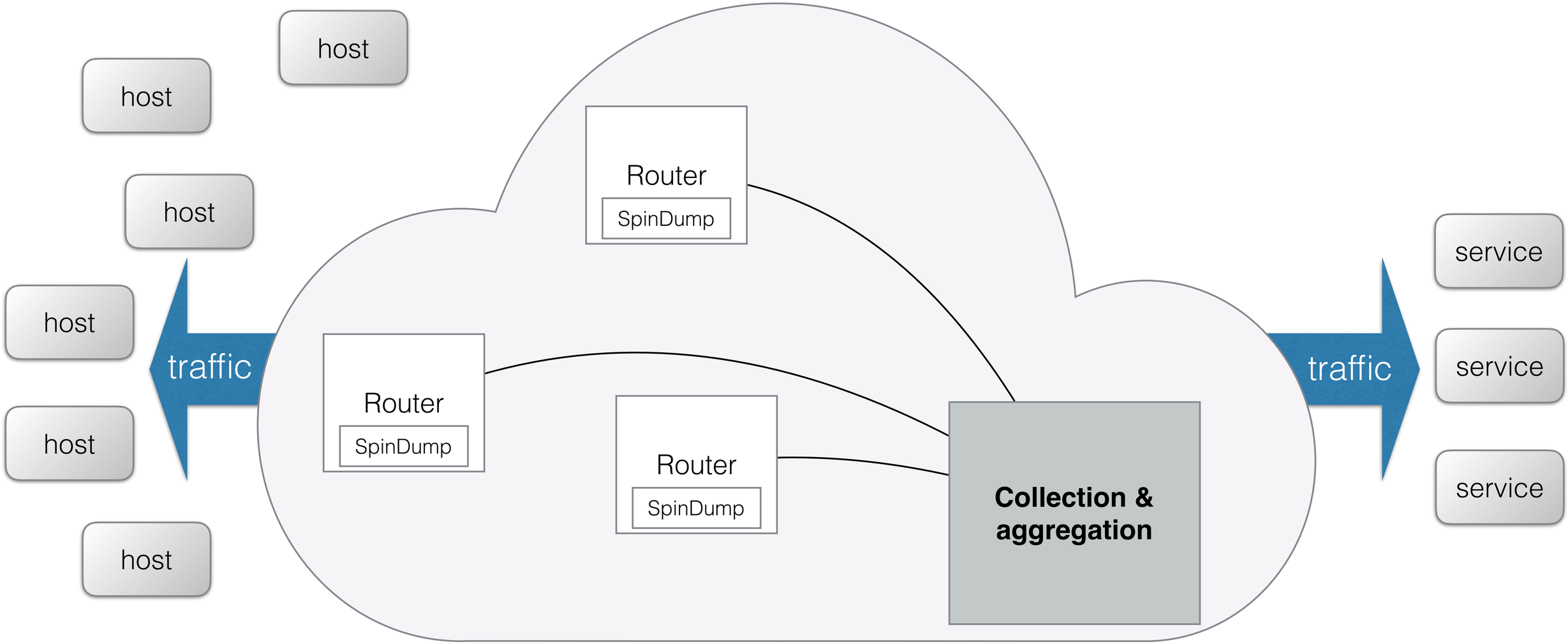
Jari Arkko and Marcus Ihlar
Ericsson

Background 1: Spindump

- An open source in-network monitoring tool: https://github.com/EricssonResearch/spindump
- Observes traffic going by, measuring end-to-end RTT; supports measurements for TCP, QUIC, ICMP, DNS, COAP, … and aggregates
- Use cases
  - Network debugging
  - Operations, e.g., alarms
  - Research
  - <Add your use case here>



Background 2: Distributed data collection



Background 3: Spindump Data Formats

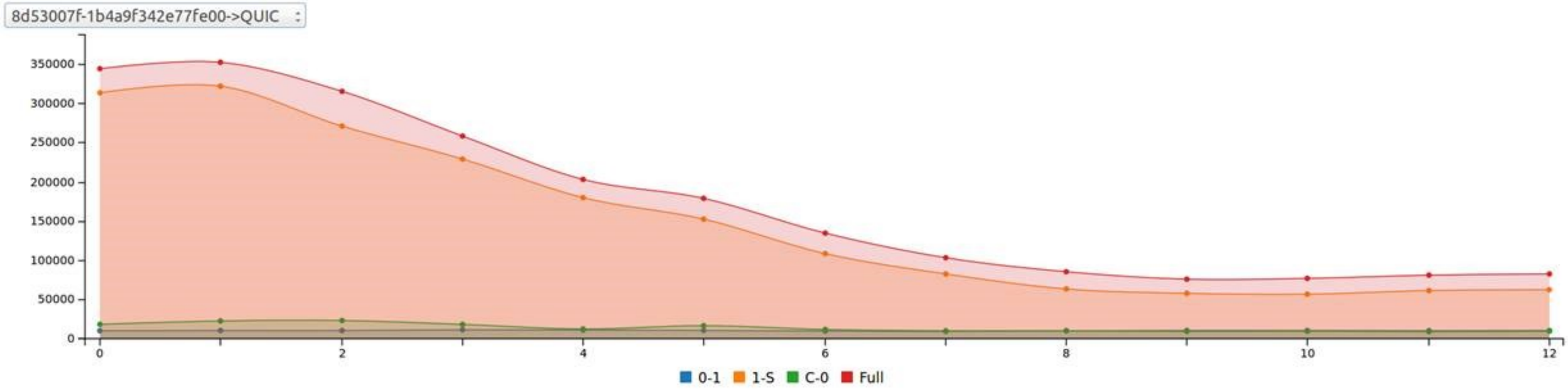
- Could be screen or file output, or delivery to a server (over https)
  - POST to example.com/data/id
- Formats
  - Human readable text
  - JSON
  - A binary format in the works

```
{
   "Event": "measurement",
   "Type": "QUIC",
   "Addrs": [“31.133.149.35","52.58.13.57"],
   "Session": “be8f318b-241534d71c3049ac",
   "Ts": "155355225957434O",
   "Full_rtt_responder": 14763,
   "Packets": 150,
   "Bytes": 136994
}
```

- Data could be either per event, buffered set of events, or aggregated over time & connections

```
“Avg_rtt": 12000,
 "Packets": +1000000,
```

Example


8d53007f-1b4a9f342e77fe00->QUIC
350000
300000
250000
200000
150000
100000
50000
0
0
2
4
6
8
10
12
0-1
1-S
C-0
Full

Observations

- Some unification in logging approaches and formats would be useful
- Consider different uses from debugging to research
- The ability to combine information from multiple sources would be very valuable, e.g., break down delay component to different network path parts
- “Semantic compression” should have more priority than looking at purely formatting issues (e.g., aggregation benefits are bigger than format changes)
- Anonymization would be very helpful for research and network adjustment use cases (maybe not so much for debugging)