

QUIC Logging

The In-Network View

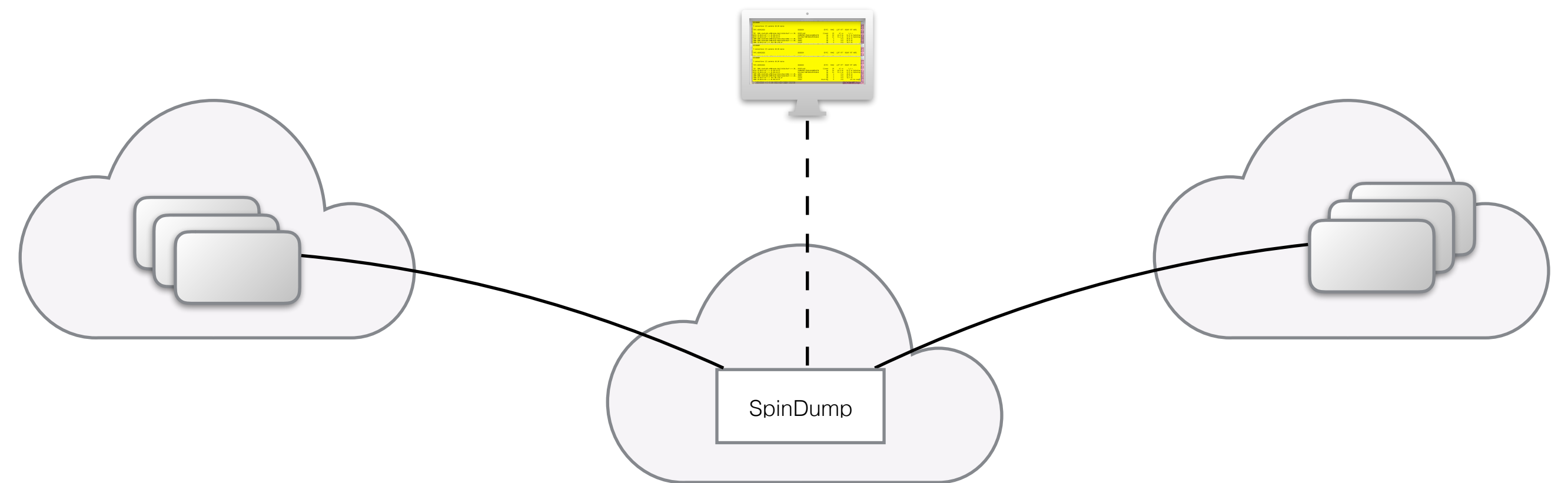
```
SPINDUMP
7 connections 121 packets 58.3K bytes

TYPE ADDRESSES                               SESSION                               STATE  PAKS  LEFT RTT  RIGHT RTT  NOTE
TCP 2001:1bc8:101:e900:b46c:8a12:5e36:8cc9 <-> 20.. 59259:443 Closed 29 67 us 1.1 s
QUIC 10.30.0.167 <-> 52.58.13.57 24800f30-1fb644b4b083c37d Up 22 16.5 ms 51.5 ms Spinning
QUIC 10.30.0.167 <-> 52.58.13.57 b21c36e7-08f1024e9371dd12 Up 21 71.4 ms 47.9 ms Spinning
ICMP 2001:1bc8:101:e900:d4a2:b311:2bac:b8de <-> 20.. 40364 Up 4 n/a 25.0 ms
ICMP 2001:1bc8:101:e900:b46c:8a12:5e36:8cc9 <-> 20.. 40364 Up 4 n/a 24.9 ms
ICMP 10.30.0.167 <-> 151.101.245.67 44445 Up 4 n/a 16.7 ms
ICMP 10.30.0.167 <-> 52.58.13.57 47517 Starting 2 n/a n/a No resp.
```

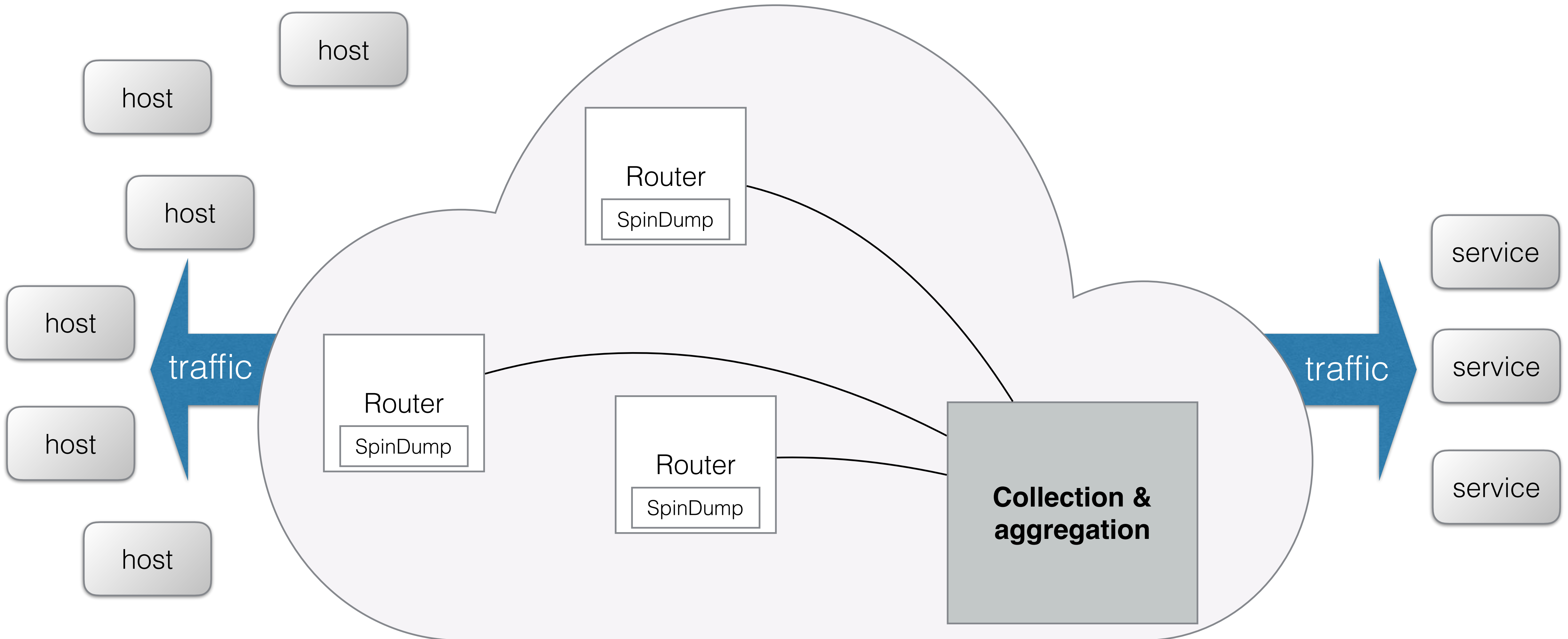
Jari Arkko and Marcus Ihlar
Ericsson

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": "1553552259574340",  
  "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,
```

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)

Example

(to be supplied ... an example graph of what parts of network path contributed to the RTT of a connection ... graph produced by the aggregation server built in the IETF-104 hackathon)