[qlog]

structured event logging for (encrypted) protocols

Robin Marx    robin.marx@kuleuven.be
What’s in a name?

[qlog] = QUIC Logging

QUIC and HTTP/3 are complex
- Will need good debugging and analysis tools
- Tools need data to ingest

Typical network logging

get raw wire image from one location

wireshark
1. QUIC is almost entirely encrypted

Storing full packet captures and TLS secrets is bad for:
- scalability
- privacy

TCP:
- Src Port
- Dest Port
- Seq No
- ACK No
- Flags
- Windows
- Options
- Encrypted: Payload

UDP:
- Src Port
- Dest Port
- Flags
- Connection ID

QUIC (open):
- Flags
- Connection ID

QUIC (encrypted):
- Packet No
- Frame
- ACK
- Window
- Options
- Payload

[Image source: https://labs.apnic.net/?p=1207]
1. QUIC is almost entirely encrypted

2. not everything is sent on the wire
   congestion control, decision making, internal errors, ...
structured endpoint logging

get data from (both) implementations directly
Event examples

```json
{
    "time": 15000,
    "name": "transport:packet_received",
    "data": {
        "header": {
            "packet_type": "1rtt",
            "packet_number": 25
        },
        "frames": [
            {
                "frame_type": "ack",
                "acked_ranges": [
                    [10,15],
                    [17,20]
                ]
            }
        ]
    }
}
```

```json
{
    "time": 15001,
    "name": "recovery:metrics_updated",
    "data": {
        "min_rtt": 25,
        "smoothed_rtt": 30,
        "latest_rtt": 25,
        "congestion_window": 60,
        "bytes_in_flight": 77000,
    }
}
```
QUIC and HTTP/3 tools

https://github.com/quiclog/qvis

https://qvis.quictools.info
“TCPtrace” for QUIC

https://qvis.quictools.info

https://github.com/quiclog/qvis
“TCPtrace” for QUIC

https://qvis.quictools.info

https://github.com/quiclog/qvis
BBR
- 0-RTT bandwidth estimate too low, because no previously ACKed packets
- Cross-Atlantic growth too slow, due to bad RTT estimates
- Implicit ACKs during handshake caused huge RTT swings, preventing BW growth

Flow control
- FC for reset streams was not released, leading to FC-limited transfer

➔ Tooling is crucial for non-experts, still useful for experts

https://github.com/facebookincubator/mvfst/commit/39d1a991450d02b2a618bb2fa915c5cf462ee45
https://github.com/facebookincubator/mvfst/commit/a5ed2117f36e875f47422ff103cd2026754f3545
75% of QUIC/H3 stacks support direct qlog output:
- mvfst
- ngtcp2
- quiche
- quic-go
- aioquic
- quicly / H2O
- neqo
- picoquic
qlog draft adoption in QUIC wg
- Expected before or during IETF 111
- Part of recharter

Goals
- Flesh out schema’s for QUIC and HTTP/3
- Prepare qlog for broader use with other protocols / applications
  - TCP
  - Multipath TCP and QUIC, MASQUE
  - TLS, HTTP, DNS, WebTransport
  - ...

https://qlog.edm.uhasselt.be/anrw/
Use eBPF and kprobes to get info from kernel
- Combine with pcaps to get full info

![qlog] for TCP

Figure 29: CWND increasing during one congestion avoidance phase in NS-3 (10ms, 5Mbps)
Log path ID with each event, split out later
qlog is currently JSON-based

- 500 MB transfer → 300 MB qlog
- With compression: 18 MB

- Tests on Gbps networks
  - difficult to scale
  - Binary formats or different tools can help there

https://github.com/nibanks/quic-perf
https://github.com/quiclog/internet-drafts/issues/30
https://github.com/microsoft/msquic/tree/main/src/plugins/wpa
How do you actually work your magic?

Configuration

```
cmd="/sysctl -q net.ipv4.tcp_congestion_control
machine_cwnd="machine_cwnd"
echo 40 "net.ipv4.tcp_congestion_control" net.ipv4.tcp_window_scaling
```
How do you actually work your magic?
Next steps

Not standardizing tools
- But hoping to make creation of re-usable tools and test suites easier
- Working backwards from tools to see what we need in qlog is a possible approach

Come join us soon!
- Drafts adoption in the QUIC wg (part of recharter)
- Expected before or during IETF 111

In the mean time
- Join us on github.com/quiclog/internet-drafts
- Join the qlog IETF mailing list ietf.org/mailman/listinfo/qlog

Give feedback now!