# [qlog]

# structured event logging
# for (encrypted) protocols

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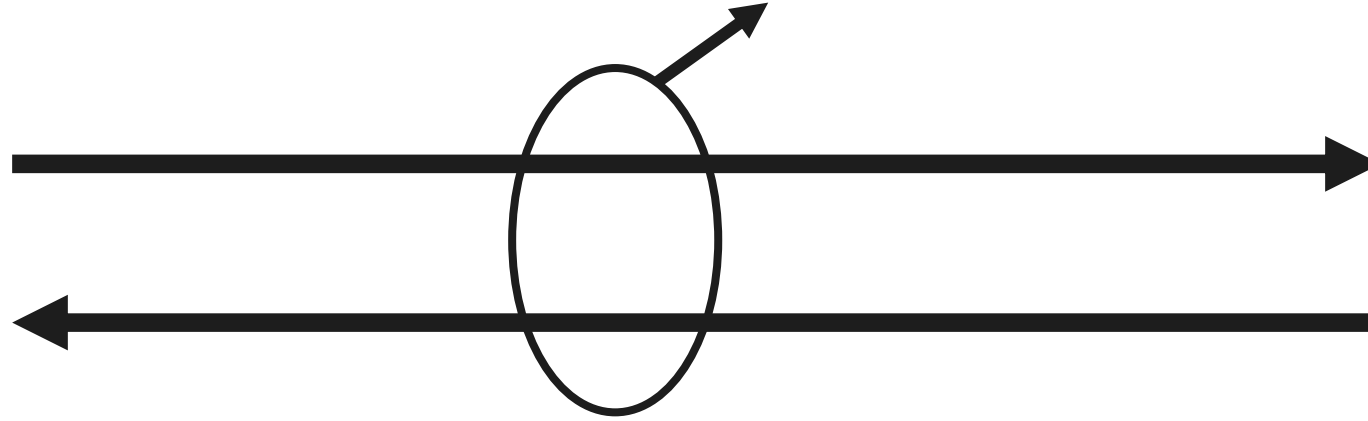
# What's in a name?

[qlog] = QUIC Logging

## QUIC and HTTP/3 are complex

- Will need good debugging and analysis **tools**
- Tools need **data** to ingest

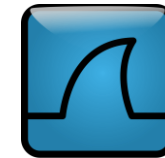# Typical network logging

get raw wire image
from one location



wireshark

# 1. QUIC is almost entirely encrypted

**TCP**

**Encrypted**

| Src Port | Dest Port | Seq No | ACK No | Flags | Windows | Options |

**Payload**

**UDP**

| Src Port | Dest Port |

**QUIC (open)**

| Flags | Connection ID |

**QUIC (encrypted)**

| Packet No | Frame | ACK | Window | Options | Payload |
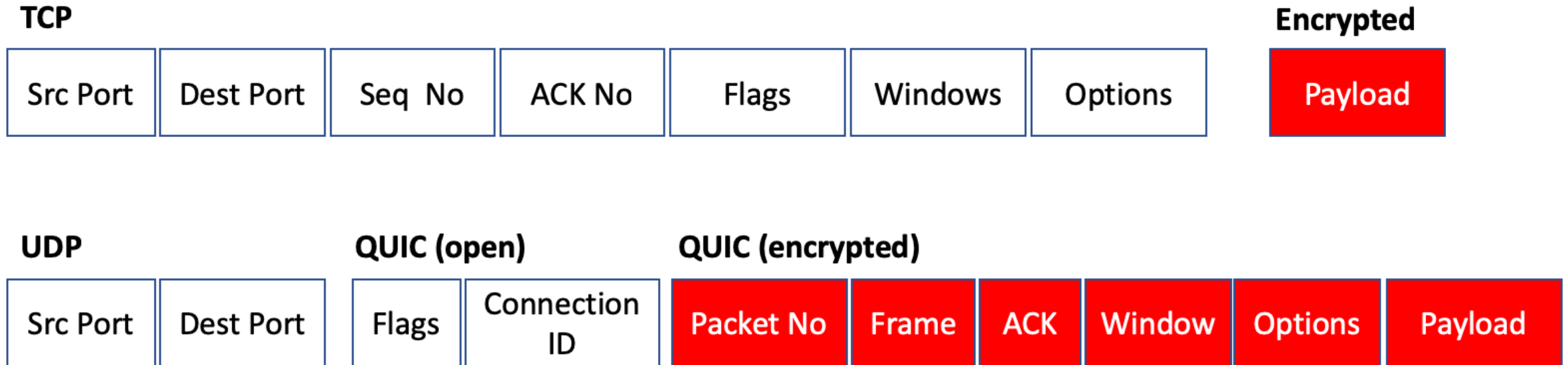
## Storing full packet captures and TLS secrets is bad for:
- scalability
- privacy

# 1. QUIC is almost entirely encrypted



**TCP**

| Src Port | Dest Port | Seq No | ACK No | Flags | Windows | Options | | **Encrypted** Payload |

**UDP** | **QUIC (open)** | **QUIC (encrypted)**

| Src Port | Dest Port | | Flags | Connection ID | | Packet No | Frame | ACK | Window | Options | Payload |
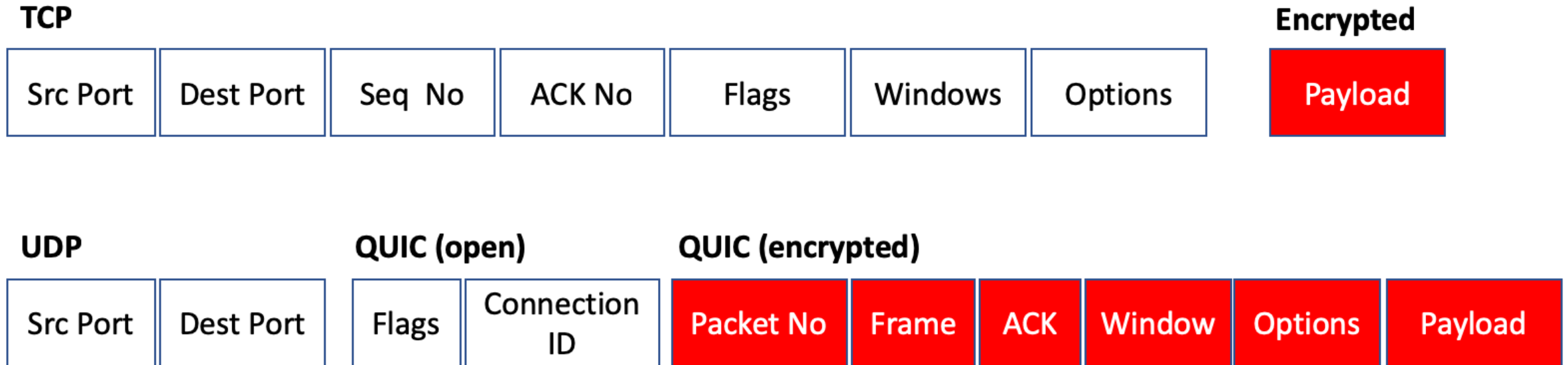
# 2. not everything is sent on the wire
### congestion control, decision making, internal errors, …

# [qlog] structured endpoint logging



[qlog]          [qlog]

**get data from (both) implementations directly**

# Event examples

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

```
{
"time": 15001,
"name": "recovery:metrics_updated",
"data": {
  "min_rtt": 25,
  "smoothed_rtt": 30,
  "latest_rtt": 25,

  "congestion_window": 60,
  "bytes_in_flight": 77000,
}
```

6

# QUIC and HTTP/3 tools

https://qvis.quictools.info

https://github.com/quiclog/qvis

"TCPtrace" for QUIC

https://qvis.quictools.info

8

https://github.com/quiclog/qvis
https://blog.cloudflare.com/cubic-and-hystart-support-in-quiche
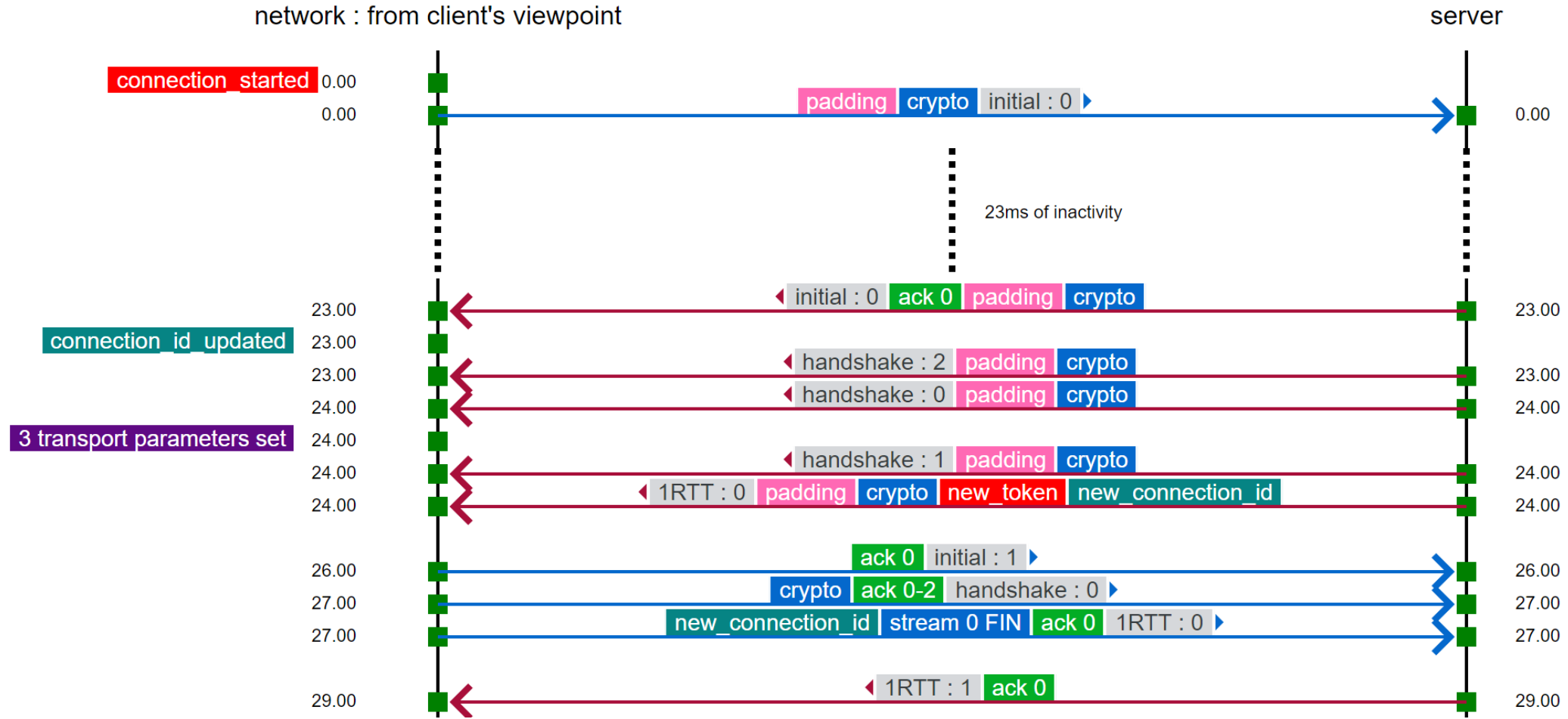
# [qlog] support

> **75%** of QUIC/H3 stacks support direct qlog output:

- mvfst
- ngtcp2
- quiche
- quic-go
- aioquic
- quicly / H2O
- neqo
- picoquic     - ...

**mjoras** 10:35 PM
@rmarx we currently have qlog enabled in prod with similar amounts of events being recorded a day as I quoted before (dozens of billions).

9

https://qlog.edm.uhasselt.be/anrw
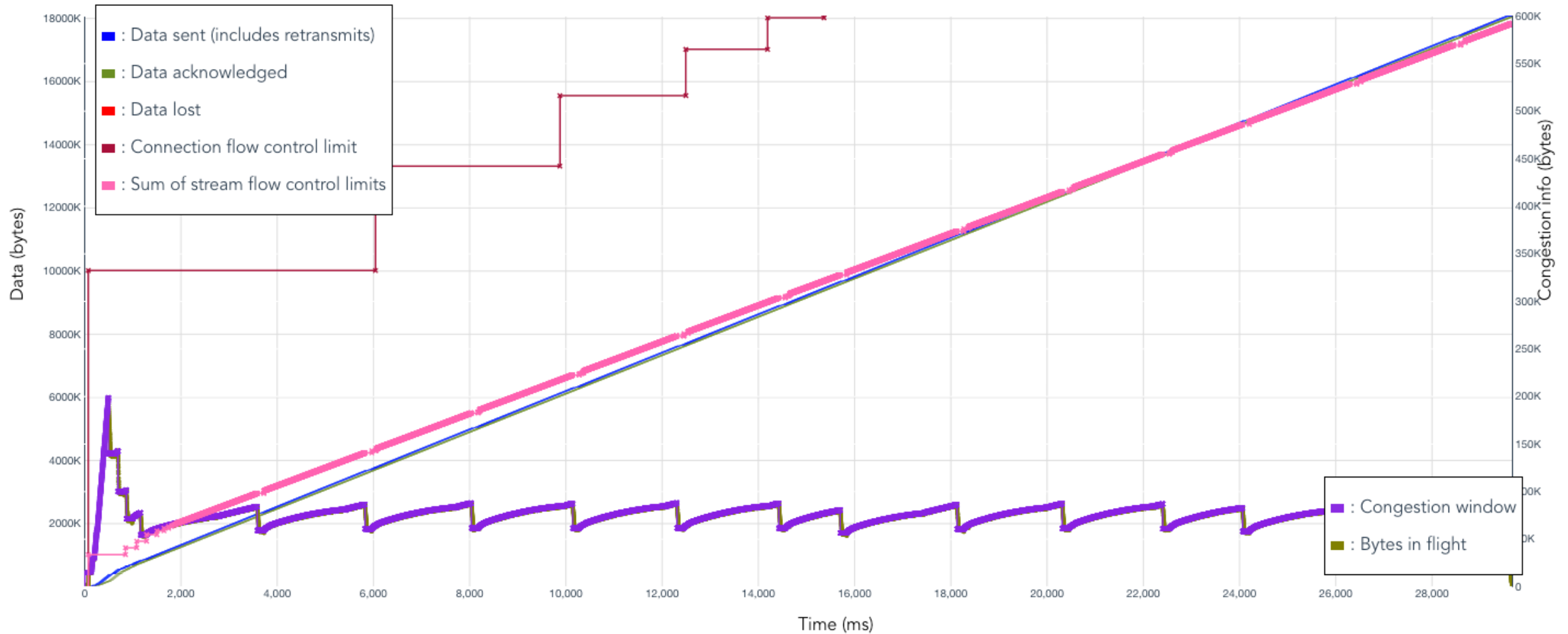
# [qlog] adoption

## 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 + HTTP/x
    - DNS, BGP, WebTransport
    - Multipath TCP and QUIC, MASQUE
    - Adaptive BitRate (ABR) video streaming logic
    - ...

https://tools.ietf.org/html/draft-marx-qlog-main-schema-02
https://tools.ietf.org/html/draft-marx-qlog-event-definitions-quic-h3-02
https://research.edm.uhasselt.be/~mwijnants/pdf/herbotsCONEXT2020.pdf

# [qlog] drafts

## Main
**Protocol-agnostic**

- Container / metadata
- Format (JSON)
- *Best practices / guidelines*

## QUIC

- Connectivity
- Transport
- Recovery

## HTTP/3

- HTTP/3
- QPACK

…
Hopefully more to come

https://tools.ietf.org/html/draft-marx-qlog-main-schema-02
https://tools.ietf.org/html/draft-marx-qlog-event-definitions-quic-h3-02

11
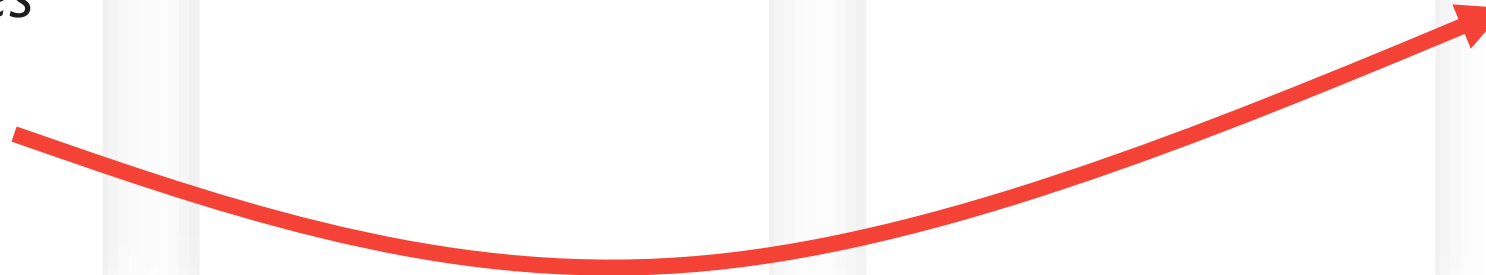
# [qlog] for more than QUIC/H3

## Plenty of challenges

- Event definitions
- Formats and datatypes
- Privacy aspects

<br>

- Operational aspects
- Cross-protocol tooling
- Protocol overlaps (e.g., TCP and QUIC, HTTP/3 vs HTTP/2 and 1, DoX, …)
- …

# Event definitions

QUIC wire image

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

# Event definitions

## QUIC wire image

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

**and / or?**

## Implementation behaviour

```
{
"time": 15000,
"name": "transport:packets_acked",
"data": {
    "packets": [
        19,20
    ]
}
```

```
{
"time": 15000,
"name": "transport:packets_lost",
"data": {
    "packets": [
        16
    ]
}
```

# Event definitions

TCP wire image

```json
{
"time": 15000,
"name": "transport:packet_received",
"data": {
    "header": {
        "seq_number": 25,
        "options": [
        {
            "type": "sack",
            "acked_ranges": [
                [10,15],
                [17,20]
            ]
        }]
    }
}}
```

and /
or?

Implementation behaviour

```json
{
"time": 15000,
"name": "transport:packets_acked",
"data": {
    "packets": [
        19,20
    ]
}
```

```json
{
"time": 15000,
"name": "transport:packets_lost",
"data": {
    "packets": [
        16
    ]
}
```

# [qlog] serialization format

## qlog is currently JSON-based

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

## Format agnostic
- Define datatypes and schema
- Can be mapped to multiple serialization formats
    - Which one(s) should we focus on?
    - Automated generation from text?

## Stream vs file-based
- Typical ingestion/storage/analysis pipelines

```
class StreamFrame{
    frame_type:string = "stream";

    stream_id:uint64;

    offset:uint64;
    length:uint64;

    fin?:boolean;

    raw?:bytes;
}
```

https://github.com/quiclog/internet-drafts/issues/30

# [qlog] privacy

## Lots of sensitive data

- IP addresses / Connection IDs
- HTTP payloads, SNIs
- Timestamps?

## "Sanitization levels"

- From loose to strict
- Concrete guidelines and rules
- Tagging of individual fields

## Log sharing

- Could aid with QUIC manageability, research, troubleshooting
- Operational aspects (APIs, endpoints, …)

https://huitema.wordpress.com/2020/07/21/scrubbing-quic-logs-for-privacy

# Next steps

## Eventually:

- Separate qlog wg for main aspects?
- Individual (protocol) wg's define new qlog documents?

## First step:

- 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!