



The *Barbie* Update

IETF 117 – San Francisco – 2023-07

**Robin Marx**, Luca Niccolini, Marten Seemann, Lucas Pardue

# Since IETF 116

- Published 1 set of 3 new drafts
  - Extensive editorial updates
  - Bumped qlog\_version from 0.3 to 0.4
  - Added support for extension RFCs
- Received early SECDIR review of privacy + security considerations
  - "The [current] security considerations seem appropriate given the generic nature of the schema."
  - Minor remarks and nits resolved in new drafts
  - Thanks to Dan Harkins!



# Bumped qlog\_version 0.3 → 0.4 [#300](#), [#302](#)

qlog events have a “category”

- e.g., **transport:packet\_sent**

Categories were too high-level:

- Renamed **http** to **h3**

`http:frame_parsed` → `h3:frame_parsed`

- Renamed **transport** to **quic**

`transport:packet_sent` → `quic:packet_sent`

Spoilers: the plot will thicken

- Recommend holding off on implementing changes



# Extension RFC support [#310](#), [#311](#), [#312](#), [#325](#)

RFC9287 (GREASE bit)

RFC9220 (Extended CONNECT for WebSockets)

RFC9218 (Extensible prioritization)

- h3:priority\_updated event added



# Extension RFC support [#310](#), [#311](#), [#312](#), [#325](#)

RFC9287 (GREASE bit)

RFC9220 (Extended CONNECT for WebSockets)

RFC9218 (Extensible prioritization)

- h3:priority\_updated event added

RFC9221 (QUIC DATAGRAM) + RFC9297 (H3 DATAGRAM)

- NO capsule protocol
- Used to be [draft-marx-quick-log-datagram](#)
- quic:data\_moved → quic:stream\_data\_moved + quic:datagram\_data\_moved
- BUT... *what is a datagram?*



# How to log UDP-level info? [#323](#)

## Previous:

- **transport**:datagrams\_sent

## Current:

- **quic**:datagrams\_sent
- BUT also: QUICDatagramFrame inside quic:packet\_sent
- What *exactly* is a “datagram”?

## Proposal:

- **udp**:datagrams\_sent
- *main/only downside: another breaking change*



# Versioning of qlog and additional schema [#293](#)

Old:

```
{
  "qlog_version": "0.3",
  "qlog_format": "JSON",
  "additional_schema": [
    "draft-ietf-quicklog-quicklog-events-04",
    "draft-ietf-quicklog-quicklog-h3-events-04"
  ],
  "summary": {...},
  "traces": [...]
}
```

New (and improved?):

```
{
  "qlog_version": "0.4",
  "qlog_format": "JSON",
  "additional_schema": [
    "urn:ietf:params:qlog:quicklog-events-06",
    "urn:ietf:params:qlog:h3-events-06"
  ],
  "summary": {...},
  "traces": [...]
}
```

Use **IANA** to register URI/URN for extension RFCs

# Removal of .well-known endpoint [#313](#)

After generation, qlog implementers **MAY** make available generated logs and traces on an endpoint (typically the server) via the following .well-known URI:

`.well-known/qlog/IDENTIFIER.extension`

Proposal: remove from qlog docs (MERGED)

Maybe: move to a new draft and find a home for it





# Multipath/Migration support [#134](#), [#318](#)



## Current:

YES: PATH\_CHALLENGE + PATH\_RESPONSE, preferred\_address, ...

NO: no support for IP addresses, “PathMigrationStarted/Stopped” events, ...

## Plan:

Add opaque “**path\_id**” top-level field (can be logged for each event)

- Nice and flexible, but is it really enough?
- Do we have a clearer view on this with the progressing multipath interop?

# The cliffhanger (PR not ready)

- **QPACK event reworking proposal ([#199](#))**



## The end (for now)

- Introduce udp category
- versioning / additional schema
- Removed .well-known
- “path\_id” as panacea
- QPACK events

