structured event logging

the X-Files update

Robin Marx  rmarx@akamai.com
The QUIC-hat Wearing Man
Security and Privacy Considerations

- Old approach (IETF 113)
  - Extensive guidance:
    - Per-field sensitivity indicators
    - Anonymization strategies per data type

- BUT: this is a **deep rabbit hole**
  - Little existing guidance in IETF
  - Anonymization alone is not enough
  - Requirements depend on use case / deployment

- *Getting this right* would substantially delay qlog

Old discussion: [https://github.com/quicwg/qlog/issues/142](https://github.com/quicwg/qlog/issues/142)
Security and Privacy Considerations

- New approach (IETF 115+)
  - Only base guidance in qlog:
    - Highlight privacy risk
    - Provide (non-exhaustive?) examples of sensitive qlog
    - Touch upon tips and tricks for managing risk

- Start parallel effort for detailed recommendations
  - TBD
  - Broader than just qlog
  - Analogue to e.g., RFC6973 but for logs/captured data

Old discussion: https://github.com/quicwg/qlog/issues/142
New discussion: https://github.com/quicwg/qlog/issues/259
Applying Cunningham’s Law

“IPv6 is worse for privacy”

“use .json instead of .qlog to confuse hackers”

“qlogs MUST be stored on a blockchain”

Old discussion: https://github.com/quicwg/qlog/issues/142
New discussion: https://github.com/quicwg/qlog/issues/259

Extensibility

- Goal: add new qlog definitions later on in new documents
- Difficulty: CDDL definitions
  - Ideally: merge base + extension documents into 1 big CDDL schema

Main schema

```plaintext
Event = {
  time: float64
  name: text
  data: $ProtocolEventBody
}
```

HTTP/3 document

```plaintext
HTTPEvents = HTTPParametersSet / HTTPFrameCreated / HTTPPushResolved

$ProtocolEventBody $= HTTPEvents
```
Extensibility

- Goal: add new qlog definitions later on in new documents
- Difficulty: CDDL definitions
  - Ideally: merge base + extension documents into 1 big CDDL schema

- Test it out using QUIC and H3 **DATAGRAM** frames

```python
QUICDatagramFrame = {
    frame_type: "datagram"
    ? length: uint64
    ? raw: RawInfo
}
$QuicFrame /= QUICDatagramFrame
```

- Also need transport parameter
- Also needs H3 SETTING
- **Where does it end?**

The Truth Is (not?) Out There

ACK_MP Frame {
  Type (i) = TBD-00..TBD-01 (experiments use 0xbaba00..0xbaba01),
  Packet Number Space Identifier (i),
  Largest Acknowledged (i),
  ACK Delay (i),
  ACK Range Count (i),
  First ACK Range (i),
  ACK Range (..) ..., 
  [ECN Counts (..)],
}

- Make everything an extension point?
- Have new docs re-define/overwrite old stuff?
- Don’t have everything in CDDL?
- ...

https://github.com/rmarx/draft-marx-quic-qlog-datagram
https://github.com/quicwg/qlog/issues/261
- Plan to redefine events: Help appreciated!

InsertWithoutNameReferenceInstruction = {
    instruction_type: "insert_without_name_reference"
    huffman_encoded_name: bool
    ? name_length: uint32
    ? name: text
    huffman_encoded_value: bool
    ? value_length: uint32
    ? value: text
}

HeadOfLineBlocked = {
    stream_id: uint64
}
QPACK

- Thinking of splitting H3 and QPACK events into separate documents

https://github.com/quicwg/qlog/issues/262
Some remaining design issues

- Provision **something** for MultiPath?
  - Define path_id field, can also be useful for connection migration?

ECN events: https://github.com/quicwg/qlog/issues/212
Send blocking events: https://github.com/quicwg/qlog/issues/132
Multipath support: https://github.com/quicwg/qlog/issues/134
Connection Migration approach: https://github.com/quicwg/qlog/issues/79
ConnectionState definition: https://github.com/quicwg/qlog/issues/239
Merci!