QUIC

-transport-02
COMPLAINTS

https://github.com/quicwg/base-drafts/issues
LOTS of changes

- Defined short and long packet headers (#40, #148, #361)
- Defined a versioning scheme and stable fields (#51, #361)
- Define reserved version values for “greasing” negotiation (#112, #278)
- The initial packet number is randomized (#35, #283)
- Narrow the packet number encoding range requirement (#67, #286, #299, #323, #356)
- Defined client address validation (#52, #118, #120, #275)
- Define transport parameters as a TLS extension (#49, #122)
- SCUP and COPT parameters are no longer valid (#116, #117)
- Transport parameters for 0-RTT are either remembered from before, or assume default values (#126)
- The server chooses connection IDs in its final flight (#119, #349, #361)
- The server echoes the Connection ID and packet number fields when sending a Version Negotiation packet (#133, #295, #244)
- Defined a minimum packet size for the initial handshake packet from the client (#69, #136, #139, #164)
- Path MTU Discovery (#64, #106)
- The initial handshake packet from the client needs to fit in a single packet (#338)
- Forbid acknowledgment of packets containing only ACK and PADDING (#291)
- Removed the STOP_WAITING frame (#66)
- Don’t require retransmission of old timestamps for lost ACK frames (#308)
- Clarified that frames are not retransmitted, but the information in them can be (#157, #298)
- Error handling definitions (#335)
- Split error codes into four sections (#74)
- Forbid the use of Public Reset where CONNECTION_CLOSE is possible (#289)
- Define packet protection rules (#336)
- Require that stream be entirely delivered or reset, including acknowledgment of all STREAM frames or the RST_STREAM, before it closes (#381)
- Remove stream reservation from state machine (#174, #280)
- Only stream 1 does not contribute to connection-level flow control (#204)
- Stream 1 counts towards the maximum concurrent stream limit (#201, #282)
- Remove connection-level flow control exclusion for some streams (except 1) (#246)
- RST_STREAM affects connection-level flow control (#162, #163)
- Flow control accounting uses the maximum data offset on each stream, rather than bytes received (#378)
- Moved length-determining fields to the start of STREAM and ACK (#168, #277)
- Added the ability to pad between frames (#158, #276)
- Remove error code and reason phrase from GOAWAY (#352, #355)
- GOAWAY includes a final stream number for both directions (#347)
- Error codes for RST_STREAM and CONNECTION_CLOSE are now at a consistent offset (#249)
- Defined priority as the responsibility of the application protocol (#104, #303)
QUIC header format (in -01)

Regular Packets

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID (opt)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Version (opt)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Packet Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Encrypted Payload (Frames)</th>
</tr>
</thead>
</table>

ACK

WINDOW_UPDATE

STREAM

Version Negotiation Packet (Unencrypted)

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Supported Version 1

Supported Version 2

Supported Version 3

Public Reset Packet (Unencrypted)

<table>
<thead>
<tr>
<th>Flags</th>
<th>Connection ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Public Reset fields (TBD)
QUIC header format (proposed in -02)

Long Header Packets

<table>
<thead>
<tr>
<th>Type (7)</th>
<th>Connection ID (64)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Packet Number (32)</td>
</tr>
<tr>
<td></td>
<td>Version (32)</td>
</tr>
<tr>
<td></td>
<td>Payload</td>
</tr>
<tr>
<td></td>
<td>Type-dependent</td>
</tr>
<tr>
<td></td>
<td>Not always encrypted</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Version Negotiation</td>
</tr>
<tr>
<td>02</td>
<td>Client Cleartext</td>
</tr>
<tr>
<td>03</td>
<td>Non-Final Server Cleartext</td>
</tr>
<tr>
<td>04</td>
<td>Final Server Cleartext</td>
</tr>
<tr>
<td>05</td>
<td>0-RTT Encrypted</td>
</tr>
<tr>
<td>06</td>
<td>1-RTT Encrypted (key phase 0)</td>
</tr>
<tr>
<td>07</td>
<td>1-RTT Encrypted (key phase 1)</td>
</tr>
<tr>
<td>08</td>
<td>Public Reset</td>
</tr>
</tbody>
</table>
QUIC header format (proposed in -02)

Short Header Packets
(optimized for packets encrypted with TLS 1-RTT key)

<table>
<thead>
<tr>
<th>Type</th>
<th>Packet Number Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>1 octet</td>
</tr>
<tr>
<td>02</td>
<td>2 octets</td>
</tr>
<tr>
<td>03</td>
<td>4 octets</td>
</tr>
</tbody>
</table>
Server-chosen Connection ID

Client chooses random 64-bit connection ID in early packets

Server replaces with its chosen one in Final Cleartext Packet (end of handshake)

All Client Cleartext packets, 0-RTT Encrypted packets, and Non-Final Server Cleartext packets MUST use the client's randomly-generated initial Connection ID. Final Server Cleartext packets, 1-RTT Encrypted packets, and all short-header packets MUST use the final Connection ID.
Fields that will remain fixed across versions

Long header:
   - Header form bit
   - Connection ID
   - Version field
   - Packet Number field

Short header:
   - Header form bit
   - Connection ID bit
   - Connection ID
Versioning

Version 1 will be the RFC

Drafts will be identified using 0xff000000 + draft version (the integer value)

Registry: https://github.com/quicwg/base-drafts/wiki/QUIC-Versions

Request that suggestions re: IANA be taken to a GitHub issue

Version negotiation can be “greased” (0x?a?a?a?a reserved for this)

Server echoes client connection ID and packet number
Transport Parameter Negotiation

TLS extension carries QUIC transport properties

- flow control offsets
- idle timer (see later discussion)
- concurrent stream limit
- allow omission of connection ID (see later discussion)

Authenticates QUIC version negotiation

Proprietary SCUP/COPT removed

Added defaults for 0-RTT, clients can remember as well
Added section on path MTU discovery (thanks @martinduke)

Includes a requirement to set initial client packet to 1280 octets or higher

  Important: QUIC doesn’t work with a smaller path MTU

  The initial cryptographic handshake message has to fit in this packet
Packet Loss &c

STOP_WAITING is gone

If a packet contains only ACK/PADDING, don’t acknowledge it.

If you acknowledge a packet, you have processed it.

For STREAM frames, processing just means adding data to receive queue.

Streams aren’t closed until all data is sent and acknowledged.

Frames aren’t retransmitted, instead:

*New frames containing the relevant data are created*

  e.g., STREAM data, ACK ranges (not timestamps though), flow control offsets
Frame Layout

GOAWAY doesn’t include an error code or reason

GOAWAY includes max stream number in both directions (see later discussion)

Length-determination fields in STREAM and ACK moved up

Error codes in CONNECTION_CLOSE/RST_STREAM moved to same place

PAD no longer consumes remainder of packet (allows interstitial padding)
Other

Packet number encoding sizes are better defined
Initial packet number randomized
TLS provides client address validation
Flow control clarifications and improvements
Error code rationalization between all drafts
Priority is an application protocol responsibility (no protocol change)
Simplified stream state machine