Using Reliable Stream Resets in HTTP/3

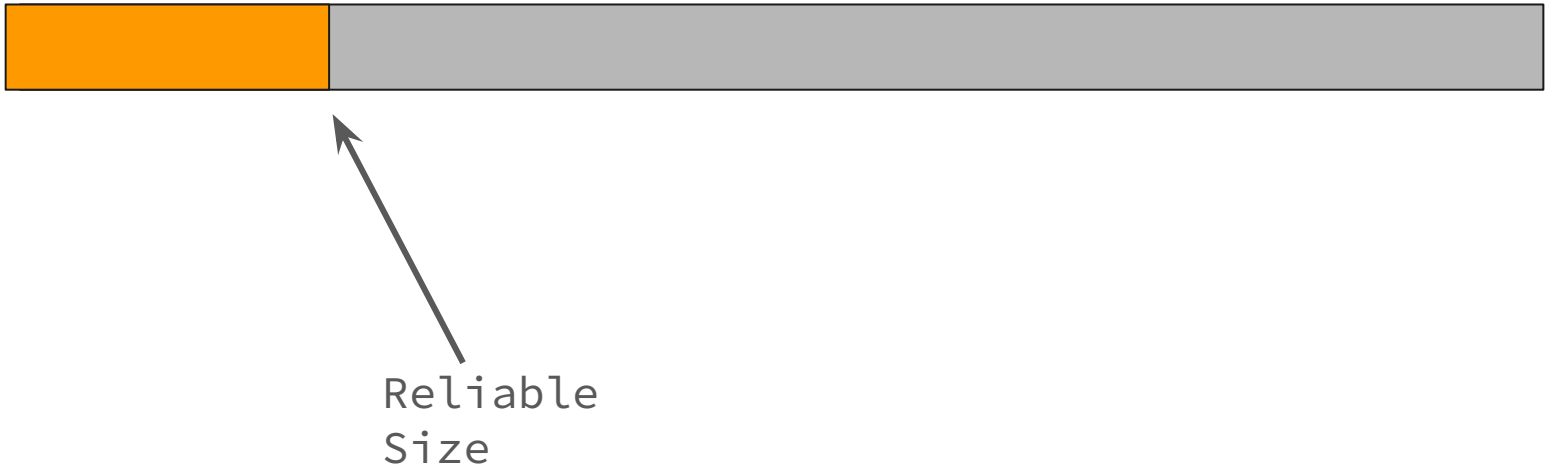
IETF 126, Vienna

Marten Seemann

QUIC Stream Reset

RFC 9000: `It is possible that all stream data has already been received when a RESET_STREAM is received [...] Similarly, it is possible for remaining stream data to arrive after receiving a RESET_STREAM frame [...]. An implementation is free to manage this situation as it chooses.`

QUIC Reliable Stream Resets



Implementation Status

Implemented in all major browsers:

- Chrome
- Firefox
- Safari

And in any server that supports WebTransport.

CONNECT proxying

RFC 9114, Section 4.4: `A TCP connection error is signaled by abruptly terminating the stream. A proxy treats any error in the TCP connection, which includes receiving a TCP segment with the RST bit set, as a stream error of type H3_CONNECT_ERROR.`

With RESET_STREAM_AT:

- protect entire data stream up to the reset

Malformed Decoded HTTP Message

- RFC 9114, Section 4.2 defines restrictions on HTTP headers
- For example: `Characters in field names MUST be converted to lowercase prior to their encoding. A request or response containing uppercase characters in field names MUST be treated as malformed.`

With RESET_STREAM_AT:

- Send HTTP status 400, and a short error message
- Then send RESET_STREAM_AT with the H3_MESSAGE_ERROR

Rejecting Requests

RFC 9114, Section 4.1.1: `When the server cancels a request without performing any application processing, the request is considered "rejected". The server SHOULD abort its response stream with the error code H3_REQUEST_REJECTED.`

With RESET_STREAM_AT:

- Send a more specific HTTP status, e.g. 429 or 503, and an error message
- Then send RESET_STREAM_AT with H3_REQUEST_REJECTED