SCONE
Internet-Draft
Intended status: Informational
Expires: 8 January 2026

M. Thomson
Mozilla
C. Huitema
Private Octopus Inc.
奥 一穂 (K. Oku)
Fastly
M. Joras
Meta
M. Ihlar
Ericsson

7 July 2025

Standard Communication with Network Elements (SCONE) Protocol
draft-ietf-scone-protocol-02

Abstract

This document describes a protocol where on-path network elements can give endpoints their perspective on what the maximum achievable throughput might be for QUIC flows.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://ietf-wg-scone.github.io/scone/draft-ietf-scone-protocol.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-scone-protocol/.

Discussion of this document takes place on the SCONE Working Group mailing list (mailto:scone@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/scone/. Subscribe at https://www.ietf.org/mailman/listinfo/scone/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-wg-scone/scone.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 8 January 2026.

Copyright Notice

Copyright (c) 2025 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

Many access networks apply rate limits to constrain the data rate of attached devices. This is often done without any indication to the applications running on devices. The result can be that application performance is degraded, as the manner in which rate limits are enforced can be incompatible with the rate estimation or congestion control algorithms used at endpoints.

Having the network indicate what its rate limiting policy is, in a way that is accessible to endpoints, allows applications to use this information when adapting their send rate.

Network elements are not limited to communicating information about rate limiting policies. Network elements in access networks could provide information to endpoints that can help account for changes in network capacity that are not suited to congestion control feedback. This might include reduced capacity due to overuse, equipment faults, or other transient issues; conversely, networks might choose to signal increased availability of capacity.

The Standard Communication with Network Elements (SCONE) protocol is negotiated by QUIC endpoints. This protocol provides a means for network elements to signal the maximum available sustained throughput, or rate limits, for flows of UDP datagrams that transit that network element to a QUIC endpoint.

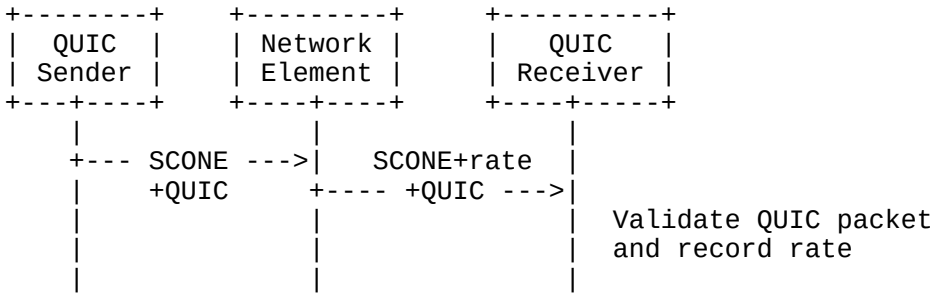
Networks with rate limiting policies use SCONE to send throughput advice to cooperating endpoints to limit overall network usage. Where congestion control signals -- such as ECN, delays and loss -- operate on a time scale of a round trip time, throughput advice operates over a much longer period. This has benefits in some networks as endpoints can fully consume network capacity in bursts, rather than extending network interaction at lower rates.

For endpoints, SCONE throughput advice makes network policies visible, which can reduce wasteful probing beyond those limits.

2. Overview

QUIC endpoints can negotiate the use of SCONE by including a transport parameter (Section 6) in the QUIC handshake. Endpoints then occasionally coalesce a SCONE packet with ordinary QUIC packets that they send.

Network elements that have rate limiting policies can detect flows that include SCONE packets. The network element can indicate a maximum sustained throughput by modifying the SCONE packet as it transits the network element.



QUIC endpoints that receive modified SCONE packets observe the indicated version, process the QUIC packet, and then record the indicated rate.

Indicated rate limits apply only in a single direction. Separate indications can be sent for the client-to-server direction and server-to-client direction. The indicated rates do not need to be the same.

Indicated rate limits only apply to the path on which they are received. A connection that migrates or uses multipath [QUIC-MP] cannot assume that rate limit indications from one path apply to new paths.

3. Applicability

This protocol only works for flows that use the SCONE packet (Section 5).

The protocol requires that packets are modified as they transit a network element, which provides endpoints strong evidence that the network element has the power to apply rate limiting; though see Section 9 for potential limitations on this.

The throughput advice signal that this protocol carries is independent of congestion signals, limited to a single path and UDP packet flow, unidirectional, and strictly advisory.

3.1. Independent of Congestion Signals

Throughput advice signals are not a substitute for congestion feedback. Congestion signals, such as acknowledgments, provide information on loss, delay, or ECN markings [ECN] that indicate the real-time condition of a network path. Congestion signals might indicate a throughput that is different from the signaled rate limit.

Endpoints cannot assume that a signaled rate limit is achievable if congestion signals indicate otherwise. Congestion could be experienced at a different point on the network path than the network element that indicates a rate limit. Therefore, endpoints need to respect the send rate constraints that are set by a congestion controller.

3.2. Unspecified Scope

Modifying a packet does not prove that the throughput that is indicated would be achievable. A signal that is sent for a specific flow is likely enforced at a different scope. The extent of that scope is not carried in the signal.

For instance, limits might apply at a network subscription level, such that multiple flows receive the same signal.

Endpoints can therefore be more confident in the throughput signal as an indication of the maximum achievable throughput than as any indication of expected throughput. That throughput will only be achievable when there is no significant data flowing in the same scope. In the presence of other flows, congestion limits are likely to determine actual throughput.

This makes the application of signals most usefully applied to a downlink flow in access networks, close to an endpoint. In that case, capacity is less likely to be split between multiple active flows.

3.3. Per-Flow Signal

The same UDP address tuple might be used for multiple QUIC connections. A single signal might be lost or only reach a single application endpoint. Network elements that signal about a flow might choose to send additional signals, using connection IDs to indicate when new connections could be involved.

3.4. Undirectional Signal

The endpoint that receives a throughput advice signal is not the endpoint that might adapt its sending behavior as a result of receiving the signal. This ensures that the throughput advice signal is attached to the flow that it is mostly likely to apply to.

An endpoint might need to communicate the value it receives to its peer in order to ensure that the limit is respected. This document does not define how that signaling occurs as this is specific to the application in use.

3.5. Advisory Signal

A signal does not prove that a higher rate would not be successful. Endpoints that receive this signal therefore need to treat the information as advisory.

The fact that an endpoint requests bitrate signals does not necessarily mean that it will adhere to them; in some cases, the endpoint cannot. For example, a flow may initially be used to serve video chunks, with the client selecting appropriate chunks based on bitrate signals, but later switch to a bulk download for which bitrate adaptation is not applicable. Composite flows from multiple applications, such as tunneled flows, might only have a subset of the involved applications that are capable of handling SCONE signals. Therefore, when a network element detects a flow using more bandwidth than advertised via SCONE, it might switch to applying its policies for non-SCONE flows, using congestion control signals.

The time and scope over which throughput advice applies is not specified. Network conditions and rate-limit policies can change in ways that make previously signaled advice obsolete, and there are no guarantees that updated advice will be sent at such events. The signaled advice can be assumed to apply to the flow of packets on the same UDP address tuple for the duration of that flow. For rate limiting networks, rate limiting policies often apply on the level of a device or subscription, but endpoints cannot assume that this is the case. A separate signal can be sent for each flow.

4. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [BCP14] when, and only when, they appear in all capitals, as shown here.

5. SCONE Packet

A SCONE packet is a QUIC long header packet that follows the QUIC invariants; see Section 5.1 of [INVARIANTS].

Figure 1 shows the format of the SCONE packet using the conventions from Section 4 of [INVARIANTS].

```
SCONE Packet {
  Header Form (1) = 1,
  Reserved (1),
  Rate Signal (6),
  Version (32) = 0xSCONE1 or 0xSCONE2,
  Destination Connection ID Length (8),
  Destination Connection ID (0..2040),
  Source Connection ID Length (8),
  Source Connection ID (0..2040),
}
```

Figure 1: SCONE Packet Format

The most significant bit (0x80) of the packet indicates that this is a QUIC long header packet. The next bit (0x40) is reserved and can be set according to [QUIC-BIT].

The low 6 bits (0x3f) of the first byte contain the Rate Signal field. Values for this field are described in Section 5.1.

This packet includes a Destination Connection ID field that is set to the same value as other packets in the same datagram; see Section 12.2 of [QUIC].

The Source Connection ID field is set to match the Source Connection ID field of any packet that follows. If the next packet in the datagram does not have a Source Connection ID field, which is the case for packets with a short header (Section 5.2 of [INVARIANTS]), the Source Connection ID field is empty.

SCONE packets SHOULD be included as the first packet in a datagram. This is necessary in many cases for QUIC versions 1 and 2 because packets with a short header cannot precede any other packets.

5.1. Rate Signals

The Rate Signal field in SCONE uses the low 6 bits (0x3f) of the first byte. This field is encoded as a logarithmically spaced distribution over a range defined by the SCONE protocol version.

When sent by a QUIC endpoint, the Version field of a SCONE packet is set to 0xSCONE2 and the Rate Signal field is set to 0x3F (63), indicating no rate limit is in place or that the SCONE protocol is not supported by network elements on the path. All other values (0x00 through 0x3F for protocol version 0xSCONE1 and 0x00 through 0x3E for protocol version 0xSCONE2) represent the ceiling of rates being advised by the network element(s) on the path.

For SCONE protocol version 0xSCONE1, the rate limits use a logarithmic scale with:

* Base rate (b_min) = 100 Kbps

* Bitrate at value n = b_min * 10^(n/20)

For SCONE protocol version 0xSCONE2, the rate limits use a logarithmic scale with:

* Bitrate at value n = b_min * 10^((n + 64)/20)

With two versions combined, bitrates between 100 Kbps and 199.5 Gbps can be expressed.

Some notable values in these ranges include:

Version	Rate Signal	Bitrate
0xSCONE1	0	100 Kbps
0xSCONE1	10	316 Kbps
0xSCONE1	20	1 Mbps
0xSCONE1	30	3.16 Mbps
0xSCONE1	40	10 Mbps
0xSCONE1	50	31.6 Mbps
0xSCONE1	60	100 Mbps
0xSCONE2	6	316 Mbps
0xSCONE2	16	1 Gbps
0xSCONE2	26	3.16 Gbps
0xSCONE2	36	10 Gbps
0xSCONE2	46	31.6 Gbps
0xSCONE2	56	100 Gbps
0xSCONE2	62	199.5 Gbps
0xSCONE2	63	No limit

Table 1

5.2. Endpoint Processing of SCONE Packets

Processing a SCONE packet involves reading the value from the Rate Signal field. However, this value MUST NOT be used unless another packet from the same datagram is successfully processed. Therefore, a SCONE packet always needs to be coalesced with other QUIC packets.

A SCONE packet is defined by the use of the longer header bit (0x80 in the first byte) and the SCONE protocol version (0xTBD in the next four bytes). A SCONE packet MAY be discarded, along with any packets that come after it in the same datagram, if the Source Connection ID is not consistent with those coalesced packets, as specified in Section 5.

A SCONE packet MUST be discarded if the Destination Connection ID does not match one recognized by the receiving endpoint.

6. Negotiating SCONE

A QUIC endpoint indicates that it is willing to receive SCONE packets by including the scone_supported transport parameter (0xTBD).

This transport parameter is valid for QUIC versions 1 [QUIC] and 2 [QUICv2] and any other version that recognizes the versions, transport parameters, and frame types registries established in Sections 22.2, 22.3, and 22.4 of [QUIC].

7. Deployment

QUIC endpoints can enable the use of the SCONE protocol by sending SCONE packets Section 5. Network elements then apply or replace the Rate Signal field (Section 7.1) according to their policies.

7.1. Applying Throughput Advice Signals

A network element detects a SCONE packet by observing that a packet has a QUIC long header and one of the SCONE protocol versions (0xSCONE1 or 0xSCONE2).

A network element then conditionally replaces the Version field and the Rate Signal field with values of its choosing.

A network element might receive a packet that already includes a rate signal. The network element replaces the rate signal if it wishes to signal a lower rate limit; otherwise, the original values are retained, preserving the signal from the network element with the lower policy.

The following pseudocode indicates how a network element might detect a SCONE packet and replace an existing rate signal, given throughput advice (target_throughput).

```
is_long = packet[0] & 0x80 == 0x80
packet_version = ntohl(packet[1..5])
if is_long and (packet_version == SCONE1_VERSION or
                packet_version == SCONE2_VERSION):
  packet_throughput = \
    signal_to_throughput(packet_version, packet[0] & 0x3f)

  if target_throughput < packet_throughput:
    target_version, target_signal = \
      throughput_to_signal(target_throughput)
    packet[0] = packet[0] & 0xc0 | target_signal
    if target_version != packet_version:
      packet[1..5] = htonl(target_version)
```

8. Version Interaction

The SCONE protocol defines two versions (0xSCONE1 and 0xSCONE2) that cover different ranges of bitrates. This design allows for:

* Support for both very low bitrates (down to 100 Kbps) and very high bitrates (up to 199.5 Gbps)

* Graceful handling of network elements that might only recognize one version.

8.1. Providing Opportunities to Apply Throughput Advice Signals

Endpoints that wish to offer network elements the option to add throughout advice signals can send SCONE packets at any time. This is a decision that a sender makes when constructing datagrams. It is recommended that endpoints promptly send an initial SCONE packet once the peer confirms its willingness to receive them.

Endpoints MUST send any SCONE packet they send as the first packet in a datagram, coalesced with additional packets. An endpoint that receives and discards a SCONE packet without also successfully processing another packet from the same datagram SHOULD ignore any throughput advice signal. Such a datagram might be entirely spoofed.

A network element that wishes to signal an updated rate limit waits for the next SCONE packet in the desired direction.

8.2. Feedback To Sender About Signals

Information about throughout advice is intended for the sending application. Any signal from network elements can be propagated to the receiving application using an implementation-defined mechanism.

This document does not define a means for indicating what was received. That is, the expectation is that any signal is propagated to the application for handling, not handled automatically by the transport layer. How a receiving application communicates the throughput advice signal to a sending application will depend on the application in use.

Different applications can choose different approaches. For example, in an application where a receiver drives rate adaptation, it might not be necessary to define additional signaling.

A sender can use any acknowledgment mechanism provided by the QUIC version in use to learn whether datagrams containing SCONE packets were likely received. This might help inform whether to send additional SCONE packets in the event that a datagram is lost. However, rather than relying on transport signals, an application might be better able to indicate what has been received and processed.

SCONE packets could be stripped from datagrams in the network, which cannot be reliably detected. This could result in a sender falsely believing that no network element applied a throughput advice signal.

8.3. Interactions with congestion control

SCONE and congestion control both provide the application with estimates of a path capacity. They are complementary. Congestion control algorithms are typically designed to quickly detect and react to congestion, i.e., to the "minimum" capacity of a path. SCONE informs the endpoint of the maximum capacity of a path based on network rate limit policy, network conditions, or a combination of the two.

Consider for example a path in which the bottleneck router implements Early Congestion Notification as specified in the L4S architecture [RFC9330]. If the path capacity diminishes, queues will build up and the router will immediately start increasing the rate at which packets are marked as "Congestion Experienced". The receiving endpoint will notice these marks, and inform its peer. The incoming congestion will be detected within 1 round trip time (RTT). This scenario will play out whatever the reason for the change in capacity, whether due to increased competition between multiple applications or, for example, to a change in capacity of a wireless channel.

9. Security Considerations

The modification of packets provides endpoints proof that a network element is in a position to drop datagrams and could apply a rate limit policy. Section 8.1 states that endpoints only accept signals if the datagram contains a packet that it accepts to prevent an off-path attacker from inserting spurious throughput advice signals.

Some off-path attackers may be able to both observe traffic and inject packets. Attackers with such capabilities could observe packets sent by an endpoint, create datagrams coalescing an arbitrary SCONE packet and the observed packet, and send these datagrams such that they arrive at the peer endpoint before the original packet. Spoofed packets that seek to advertise a higher limit than might otherwise be permitted also need to bypass any rate limiters. The attacker will thus get arbitrary SCONE packets accepted by the peer, with the result being that the endpoint receives a false or misleading rate limit.

The recipient of a throughput advice signal therefore cannot guarantee that the signal was generated by an on-path network element. However, the capabilities required of an off-path attacker are substantially similar to those of on path elements.

The actual value of the throughput advice signal is not authenticated. Any signal might be incorrectly set in order to encourage endpoints to behave in ways that are not in their interests. Endpoints are free to ignore limits that they think are incorrect. The congestion controller employed by a sender provides real-time information about the rate at which the network path is delivering data.

Similarly, if there is a strong need to ensure that a rate limit is respected, network elements cannot assume that the signaled limit will be respected by endpoints.

9.1. Flooding intermediaries with fake packets

Attackers that can inject packets may compose arbitrary "SCONE-like" packets by selecting a pair of IP addresses and ports, an arbitrary rate signal, a valid SCONE version number, an arbitrary "destination connection ID", and an arbitrary "source connection ID". The SCONE packet will carry these information. A coalesced "1RTT" packet will start with a plausible first octet, and continue with the selected destination connection ID followed by a sufficiently long series of random bytes, mimicking the content of an encrypted packets.

The injected packets will travel towards the destination. The final destination will reject such packets because the destination ID is invalid or because decryption fail, but network elements cannot do these checks, and will have to process the packets. All the network elements between the injection point and the destination will have to process these packets.

Attackers could send a high volume of these "fake" SCONE packets in a denial of service (DOS) attempt against network elements. The attack will force the intermediaries to process the fake packets. If network elements are keeping state for ongoing SCONE flows, the attack can cause the excessive allocation of memory resource. The mitigation there will be the same as mitigation of other distributed DOS attacks: limit the rate of SCONE packets that a network element is willing to process; possibly, implement logic to distinguish valid SCONE packets from fake packets; or, use generic protection against Distributed DOS attacks.

Attackers could also try to craft the fake SCONE packets in ways that trigger a processing error at network elements. For example, they might pick connection identifiers of arbitrary length. Network elements can mitigate these attacks with implementations that fully conform to the specification of Section 5.

10. Privacy Considerations

The focus of this analysis is the extent to which observing SCONE packets could be used to gain information about endpoints. This might be leaking details of how applications using QUIC operate or leaks of endpoint identity when using additional privacy protection, such as a VPN.

Any network element that can observe the content of that packet can read the rate limit that was applied. Any signal is visible on the path, from the point at which it is applied to the point at which it is consumed at an endpoint. On path elements can also alter the SCONE signal to try trigger specific reactions and gain further knowledge.

In the general case of a client connected to a server through the Internet, we believe that SCONE does not provide much advantage to attackers. The identities of the clients and servers are already visible through their IP addresses. Traffic analysis tools already provide more information than the data rate limits set by SCONE.

There are two avenues of attack that require more analysis:

* that the passive observation of SCONE packets might help identify or distinguish endpoints; and

* that active manipulation of SCONE signals might help reveal the identity of endpoints that are otherwise hidden behind VPNs or proxies.

10.1. Passive Attacks

If only few clients and server pairs negotiate the usage of SCONE, the occasional observation of SCONE packets will "stick out". That observation, could be combined with observation of timing and volume of traffic to help identify the endpoint or categorize the application that they are using.

A variation of this issue occurs if SCONE is widely implemented, but only used in some specific circumstances. In that case, observation of SCONE packets reveals information about the state of the endpoint.

If multiple servers are accessed through the same front facing server, Encrypted Client Hello (ECH) may be used to prevent outside parties to identify which specific server a client is using. However, if only a few of these servers use SCONE, any SCONE packets will help identify which specific server a client is using.

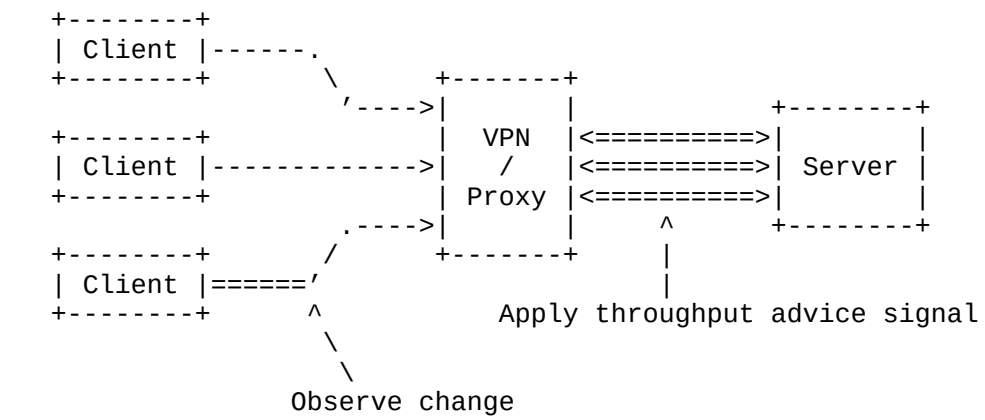
This issue will be mitigated if SCONE becomes widely implemented, and if the usage of SCONE is not limited to the type of applications that make active use of the signal.

QUIC implementations are therefore encouraged to make the feature available unconditionally. Endpoints might send SCONE packets whenever a peer can accept them.

10.2. Active Attacks

Suppose a configuration in which multiple clients use a VPN or proxy service to access the same server. The attacker sees the IP addresses in the packets behind VPN and proxy and also between the users and the VPN, but it does not know which VPN address corresponds to what user address.

Suppose now that the attacker selects a flow on the link between the VPN/proxy and server. The attacker applies throughput advice signals to SCONE packets in that flow. The attacker chooses a bandwidth that is lower than the "natural" bandwidth of the connection. A reduction in the rate of flows between client and VPN/proxy might allow the attacker to link the altered flow to the client.



An attacker that can manipulate SCONE headers can also simulate congestion signals by dropping packets or by setting the ECN CE bit. That will also likely result in changes in the congestion response by the affected client.

A VPN or proxy could defend against this style of attack by removing SCONE (and ECN) signals. There are few reasons to provide per-flow throughput advice signals in that situation. Endpoints might also either disable this feature or ignore any signals when they are aware of the use of a VPN or proxy.

11. IANA Considerations

This document registers a new QUIC version (Section 11.1) and a QUIC transport parameter (Section 11.2).

11.1. SCONE Versions

This document registers the following entries to the "QUIC Versions" registry maintained at https://www.iana.org/assignments/quic (https://www.iana.org/assignments/quic), following the guidance from Section 22.2 of [QUIC].

Value: 0xSCONE1
Status: permanent
Specification: This document
Change Controller: IETF (iesg@ietf.org)
Contact: QUIC Working Group (quic@ietf.org)
Notes: SCONE Protocol - Low Range

Value: 0xSCONE2
Status: permanent
Specification: This document

Change Controller: IETF (iesg@ietf.org)
Contact: QUIC Working Group (quic@ietf.org)
Notes: SCONE Protocol - High Range

11.2. scone_supported Transport Parameter

This document registers the scone_supported transport parameter in the "QUIC Transport Parameters" registry maintained at https://www.iana.org/assignments/quic (https://www.iana.org/assignments/quic), following the guidance from Section 22.3 of [QUIC].

Value: 0xTBD
Parameter Name: scone_supported
Status: Permanent
Specification: This document
Date: This date
Change Controller: IETF (iesg@ietf.org)
Contact: QUIC Working Group (quic@ietf.org)
Notes: (none)

12. References

12.1. Normative References

[BCP14] Best Current Practice 14, <https://www.rfc-editor.org/info/bcp14>. At the time of writing, this BCP comprises the following:

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[INVARIANTS] Thomson, M., "Version-Independent Properties of QUIC", RFC 8999, DOI 10.17487/RFC8999, May 2021, <https://www.rfc-editor.org/rfc/rfc8999>.

[QUIC] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/rfc/rfc9000>.

[QUIC-BIT] Thomson, M., "Greasing the QUIC Bit", RFC 9287, DOI 10.17487/RFC9287, August 2022, <https://www.rfc-editor.org/rfc/rfc9287>.

[QUICv2] Duke, M., "QUIC Version 2", RFC 9369, DOI 10.17487/RFC9369, May 2023, <https://www.rfc-editor.org/rfc/rfc9369>.

12.2. Informative References

[ECN] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/rfc/rfc3168>.

[QUIC-MP] Liu, Y., Ma, Y., De Coninck, Q., Bonaventure, O., Huitema, C., and M. Khlewind, "Multipath Extension for QUIC", Work in Progress, Internet-Draft, draft-ietf-quic-multipath-15, 7 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-quic-multipath-15>.

[RFC9330] Briscoe, B., Ed., De Schepper, K., Bagnulo, M., and G. White, "Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture", RFC 9330, DOI 10.17487/RFC9330, January 2023, <https://www.rfc-editor.org/rfc/rfc9330>.

Acknowledgments

Jana Iyengar has made significant contributions to the original TRAIN specification that forms the basis for a large part of this document.

Authors' Addresses

Martin Thomson
Mozilla
Email: mt@lowentropy.net

Christian Huitema
Private Octopus Inc.
Email: huitema@huitema.net

Kazuho Oku
Fastly
Email: kazuhooku@gmail.com

Additional contact information:

奥 一穂
Fastly

Matt Joras
Meta
Email: matt.joras@gmail.com

Marcus Ihlar
Ericsson
Email: marcus.ihlar@ericsson.com