SCONE
Internet-Draft
Intended status: Informational
Expires: 7 May 2026

A. Tomar
Meta
M. Ihlar
Ericsson
W. Eddy
Meta
I. Swett
Google
A. Tiwari
M. Joras
Meta

3 November 2025

SCONE Need for Defining A New On-Path Signaling Mechanism
draft-tomar-scone-ecn-02

Abstract

This document discusses the need for defining a new on-path signaling mechanism and addresses the question “why can’t we use Explicit Congestion Notification (ECN)” for the SCONE use-case.

The SCONE objective is to optimize user QoE for streaming media/video services through network assisted application-level self-adaptation. This requires a Communication Service Provider’s (CSP’s) network device to send streaming media/video traffic profile characteristics (e.g. for allowed average media/video bit-rate, burst rate etc.) with the client application endpoint to enable content self adaptation implementations by content application providers (CAPs).

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 7 May 2026.

Table of Contents

1. SCONE Background and Introduction

Currently Communication Service Provider (CSP) networks (mainly cellular and satellite networks) perform bit-rate throttling (shaping or policing) of streaming video flows. The motivation behind throttling may vary across CSPs. For example, the motivation can be:

* To support different data-rates based on the subscribers' data plans;

* To reduce egress (tonnage) towards radio base-stations in the downlink direction;

* To limit the overall capacity/bandwidth required and to manage CAPEX requirements (e.g. the needs for more RF spectrum and deployment of more radio base-stations);

* Or other reasons.

Video traffic is already 70% of all traffic on the Internet and is expected to grow to 80% by 2028. New formats like short form videos have seen tremendous growth in recent years. Both in developed and emerging markets video traffic forms 50-80% of traffic on mobile networks. These growth trends are likely to increase with new populations coming online on mobile-first markets and the observation that unlike text content, video content consumption is not being limited by literacy barriers. On the other hand, the electromagnetic spectrum is a limited resource. In order to ensure that mobile networks continue functioning in a healthy state despite this incredible growth, CSPs will be required to make infrastructure investments such as more licensed spectrum, cell densification, massive MIMO etc.

In order to flatten the rate of growth, CSPs in several markets attempt to identify and throttle video traffic based on user data plans. CSPs currently use this throttling as a way to create service differentiation between users on different payment plans and to enforce fair usage plans. There are several problems with this kind of throttling:

1. CSPs can not explicitly measure the effect that throttling has on the end user's quality of experience (QoE) making this an open loop approach. Throttling in the CSP network has a significant negative impact on streaming video application QoE and also degrades User Equipment (UE) battery performance.

2. Traffic detection and throttling for every flow is compute intensive for CSPs. With distributed UPF (user plane function) in 5G mobile networks more nodes in CSP network may need to support traffic detection and throttling. Traffic detection can have inaccuracies and these inaccuracies are expected to increase as the content delivery industry moves towards end-2-end encryption like TLS 1.3 and encrypted client hello (ECH). To perform throttling, CSP networks need to detect streaming video flows, which needs deep packet inspection and trial decryption of QUIC initial packets in order to decode and read the Server Name Indication (SNI) field present in the ClientHello message. This requires significant compute resources and risks ossifying QUIC. The throttling (shaping or policing) itself also requires non-trivial compute and memory resources.

3. The unpredictable and non-transparent behavior of traffic throttlers used by CSPs confuse the bandwidth estimation and congestion control protocols being used within end-2-end video delivery sessions between content server and client. This results in poor quality of experience (QoE) for the end user.

4. Content and Application Providers (CAPs) are designing algorithms to detect the presence of such traffic throttlers to counter their detrimental effects. These algorithms have their own inaccuracies in detection and add compute resources on the CAP side. CAPs do not and should not have access to subscriber payment plan information, making these algorithms complex to create and maintain.

There is a need for a solution to these problems which achieves both superior user QoE for CAPs, along with supporting CSPs' needs for differentiated data limitations consistent with subscriber plans.

An alternative approach is for CAPs to self-adapt the traffic corresponding to video flows. Since CAPs control the client and server endpoints and can measure end user QoE, they are in a better position to do this self-adaptation in a close loop manner. This alternative approach has already been proven to improve user QoE in production deployments [YouTube].

For this alternative approach to work a standardized secure on-path network interface is required which will enable CSP controlled network elements to signal the desired traffic profile characteristics to the CAP client/server endpoints. The Standard Communication with Network Elements (SCONE) protocol (previously known as SADCDN and SCONEPRO) is an IETF working group [SCONE-Charter] motivated by this alternate approach.

Early requirements for a technical solution based on this approach are described in [I-D.joras-sconepro-video-opt-requirements], and the protocol that the IETF working group is defining is in [I-D.ietf-scone-protocol].

2. 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 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. ECN Overview

Explicit Congestion Notification (ECN) is an extension to the Internet Protocol. ECN allows end-to-end notification of network congestion without dropping packets. ECN is an optional feature that may be used between two ECN-enabled endpoints when the underlying network infrastructure also supports it.

3.1. ECN IP Header Bits

ECN uses the two least significant (right-most) bits of the Traffic Class field in the IPv4 or IPv6 header to encode four different code points:

00 Not ECN-Capable Transport, Not-ECT

01 ECN Capable Transport(1), ECT(1) - For L4S ECN enabled Transport

10 ECN Capable Transport(0), ECT(0)

11 Congestion Experienced, CE. - To be set by a network element which can detect congestion in the link.

3.2. ECN Principles

In general both classic ECN [RFC3168] and L4S ECN RFC 9330 [RFC9330], RFC 9331 [RFC9331] and RFC 9332 [RFC9332] are mechanisms to send end-to-end notification of network congestion. Use of ECN relies on the following principles:

1. Sender to set the ECT code-points correctly for a particular flow.

2. Receiver to send the feedback back to the sender correctly based on CE value.

3. Network elements to set the CE bit correctly based on actual congestion conditions in the network.

4. ECN codepoints are not bleached or remarked within the network, other than to set the CE bit when appropriate.

4. Questions

4.1. Why can’t we use ECN instead of defining a new SCONE signal?

The CE bit in ECN is used by the network element to notify the application end-points about the congestion in the network.

SCONE is addressing a use-case wherein the network element does "intentional throttling" (shaping/policing) due to various reasons as mentioned earlier in this document such as to create service differentiation between users on different payment plans, to enforce fair usage plans, and to reduce egress (tonnage) towards radio base-stations in downlink direction. In order to replace throttling by "application level self-adaptation" SCONE signaling is required to carry video/media bit-rate etc. between network element and application end-point (note: exact metadata and data-types are to be defined during the solution definition phase of SCONE)

ECN signaling can not support this use-case due to reasons mentioned below:

* In the case of intentional throttling, CSP networks throttle the video flow to a fixed bit rate instantly. To replace the throttling in the network by "self bitrate adaptation", the CSP network is required to send a specific video bitrate within SCONE signaling meta-data to enable instant convergence of flow's bit rate to a specific bit-rate. This is not possible with ECN signaling.

* Throttling is a CSP's policy-based restriction of the flow rather than a congestion-based one. Throttling is not based on queue occupancy, competing traffic, contention for bandwidth/radio-resource etc. SCONE signal is intended for application layer adaptation whereas ECN is designed for transport layer adaptation. Through ECN, the CSP network forces the sender's transport rate to be within a specific range, rather than communicating what the application layer media bitrate should be. This implies that usage of ECN will retain some of the negative effects of network shaping such as delaying the video startup time.

  - For example, a key part of the success of YouTube's Plan Aware Streaming [YouTube] is that YouTube could still burst at a much faster rate than the long term media bitrate. This is more efficient for CAPs' servers, more efficient on client devices, and means CAPs can still use existing ABR algorithms, and just cap the quality based on the communicated long term bandwidth limit. This just isn't possible with ECN. If one allows sending 10 Mbps for some period of time, but not average, BBR is going to think it can send that fast and the ABR algorithm is going to upswitch and then both are going to have a poor experience when the delivery rate suddenly decreases. ECN avoids some packet loss, but otherwise it can't do anything fundamentally new.

* Congestion based ECN signaling is typically performed at the element which is next to the congested link so that it can detect congestion and set the CE bits accordingly. To enable subscriber' s data-plan based bit-rate signaling as well as to reduce egress towards radio base-stations in downlink direction for video streaming flows, SCONE signaling needs be performed at the element which has access to subscriber policy and which is hosted between radio base-stations and CDNs in downlink direction. In mobile networks this network element is PGW/UPF in 4G/5G networks.

* The primary targeted consumers of SCONE information are HTTP adptive bitrate video applications. The ABR decisions are typically made on the client side by the video player itself. While these players could in principle take something like ECN into account, this is not in line with current practices. There is no JavaScript API provided by browsers to get ECN information, and none of the most popular HTTP libraries used to build video players in mobile applications expose ECN information as part of their HTTP API. ECN is largely consumed by transport protocols (rather than applications) and actuated on by servers (rather than clients).

* SCONE information is intended to be per flow, not per-packet like ECN. Among typical transport protocols, only UDP and UDP-Lite support application access to the ECN bits (because other standard transport protocols typically implement congestion control themselves) [RFC8803] [RFC8804], though implementations may vary in their capabilities. ECN APIs, where they exist, are at the socket layer for datagram protocols, and have semantic binding to given packets. This does not match with the SCONE needs to signal at the flow-level and for QUIC transport.

* ECN bits within the IP header will not be enough to carry the meta-data required to be exchanged between network element and client endpoint.

  - Note - In addition to SCONE, CAPs are actively exploring L4S, but CAPs don't believe that L4S addresses the use-case SCONE is trying to solve.

4.2. Can the SCONE signal be designed so that there is no incentive to fake it, like with ECN?

Any network device which can alter ECN bits can simply drop the packets. And packet drop may have more negative impact on application' s performance compared to using ECN bits to indicate congestion in the network.

Similarly any network device which can send SCONE signaling can throttle the application flow. Throttling may have a more negative impact on an application's performance compared to using SCONE signaling to influence the incoming flow bit-rate from the sender. So like ECN, there should not be any incentive for the network device to fake the SCONE signal.

Regarding falsely manipulating CE bit in ECN (either setting or clearing the CE bit), there is no incentive either way, because both cases may have more negative impact on application's performance within the network faking the ECN signals

Similarly, there is no incentive for faking SCONE signaling (sending incorrect meta-data) because sending incorrect meta-data may have more negative impact on an application's performance within the network faking the SCONE signals.

5. Security Considerations

General SCONE security considerations are discussed in the other documents covering the specific network-to-host signaling methods. This document only addresses questions regarding use of ECN for SCONE.

6. IANA Considerations

This document has no IANA actions.

7. References

7.1. Normative References

[RFC2119] 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/rfc/rfc2119>.

[RFC8174] 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/rfc/rfc8174>.

7.2. Informative References

[I-D.ietf-scone-protocol]
Thomson, M., Huitema, C., Oku, K., Joras, M., and L. M. Ihlar, "Standard Communication with Network Elements (SCONE) Protocol", Work in Progress, Internet-Draft, draft-ietf-scone-protocol-03, 20 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-scone-protocol-03>.

[I-D.joras-sconepro-video-opt-requirements]
Joras, M., Tomar, A., Tiwari, A., and A. Frindell, "SCONEPRO Video Optimization Requirements", Work in Progress, Internet-Draft, draft-joras-sconepro-video-opt-requirements-00, 17 May 2024, <https://datatracker.ietf.org/doc/html/draft-joras-sconepro-video-opt-requirements-00>.

[RFC3168] 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>.

[RFC8803] Bonaventure, O., Ed., Boucadair, M., Ed., Gundavelli, S., Seo, S., and B. Hesmans, "0-RTT TCP Convert Protocol", RFC 8803, DOI 10.17487/RFC8803, July 2020, <https://www.rfc-editor.org/rfc/rfc8803>.

[RFC8804] Finkelman, O. and S. Mishra, "Content Delivery Network Interconnection (CDNI) Request Routing Extensions", RFC 8804, DOI 10.17487/RFC8804, September 2020, <https://www.rfc-editor.org/rfc/rfc8804>.

[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>.

[RFC9331] De Schepper, K. and B. Briscoe, Ed., "The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)", RFC 9331, DOI 10.17487/RFC9331, January 2023, <https://www.rfc-editor.org/rfc/rfc9331>.

[RFC9332] De Schepper, K., Briscoe, B., Ed., and G. White, "Dual-Queue Coupled Active Queue Management (AQM) for Low Latency, Low Loss, and Scalable Throughput (L4S)", RFC 9332, DOI 10.17487/RFC9332, January 2023, <https://www.rfc-editor.org/rfc/rfc9332>.

[SCONE-Charter]
IETF, "SCONE Working Group Charter", 31 October 2024,
<https://datatracker.ietf.org/wg/scone/about/>.

[YouTube] YouTube, "YouTube Plan Aware Streaming", 21 March 2024,
<https://datatracker.ietf.org/meeting/119/materials/
slides-119-sconepro-youtube-plan-aware-streaming-01>.

Acknowledgments

This document represents collaboration, comments, and inputs from others, including:

* Spencer Dawkins
* Alan Frindell
* Bryan Tan
* Michael Welzl
* Ted Hardie

Authors' Addresses

Anoop Tomar
Meta
Email: anooptomar@meta.com

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

Wesley Eddy
Meta
Email: wesleyeddy@meta.com

Ian Swett
Google
Email: ianswett@google.com

Abhishek Tiwari
Meta
Email: atiwari@meta.com

Matt Joras
Meta
Email: mjoras@meta.com