AVTCORE Working Group
Internet-Draft
Updates: 3550 (if approved)
Intended status: Standards Track
Expires: December 13, 2016
June 11, 2016

C. Perkins
University of Glasgow
V. Singh
callstats.io

Multimedia Congestion Control: Circuit Breakers for Unicast RTP Sessions

draft-ietf-avtcore-rtp-circuit-breakers-16

Abstract

The Real-time Transport Protocol (RTP) is widely used in telephony, video conferencing, and telepresence applications. Such applications are often run on best-effort UDP/IP networks. If congestion control is not implemented in these applications, then network congestion can lead to uncontrolled packet loss, and a resulting deterioration of the user's multimedia experience. The congestion control algorithm acts as a safety measure, stopping RTP flows from using excessive resources, and protecting the network from overload. At the time of this writing, however, while there are several proprietary solutions, there is no standard algorithm for congestion control of interactive RTP flows.

This document does not propose a congestion control algorithm. It instead defines a minimal set of RTP circuit breakers: conditions under which an RTP sender needs to stop transmitting media data, to protect the network from excessive congestion. It is expected that, in the absence of long-lived excessive congestion, RTP applications running on best-effort IP networks will be able to operate without triggering these circuit breakers. To avoid triggering the RTP circuit breaker, any standards-track congestion control algorithms defined for RTP will need to operate within the envelope set by these RTP circuit breaker algorithms.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The Real-time Transport Protocol (RTP) [RFC3550] is widely used in voice-over-IP, video teleconferencing, and telepresence systems. Many of these systems run over best-effort UDP/IP networks, and can

suffer from packet loss and increased latency if network congestion occurs. Designing effective RTP congestion control algorithms, to adapt the transmission of RTP-based media to match the available network capacity, while also maintaining the user experience, is a difficult but important problem. Many such congestion control and media adaptation algorithms have been proposed, but to date there is no consensus on the correct approach, or even that a single standard algorithm is desirable.

This memo does not attempt to propose a new RTP congestion control algorithm. Instead, we propose a small set of RTP circuit breakers: mechanisms that terminate RTP flows in conditions under which there is general agreement that serious network congestion is occurring. The RTP circuit breakers proposed in this memo are a specific instance of the general class of network transport circuit breakers [I-D.ietf-tsvwg-circuit-breaker], designed to act as a protection mechanism of last resort to avoid persistent excessive congestion. To avoid triggering the RTP circuit breaker, any standards-track congestion control algorithms defined for RTP will need to operate within the envelope set by the RTP circuit breaker algorithms defined by this memo.

2. Background

We consider congestion control for unicast RTP traffic flows. This is the problem of adapting the transmission of an audio/visual data flow, encapsulated within an RTP transport session, from one sender to one receiver, so that it does not use more capacity than is available along the network path. Such adaptation needs to be done in a way that limits the disruption to the user experience caused by both packet loss and excessive rate changes. Congestion control for multicast flows is outside the scope of this memo. Multicast traffic needs different solutions, since the available capacity estimator for a group of receivers will differ from that for a single receiver, and because multicast congestion control has to consider issues of fairness across groups of receivers that do not apply to unicast flows.

Congestion control for unicast RTP traffic can be implemented in one of two places in the protocol stack. One approach is to run the RTP traffic over a congestion controlled transport protocol, for example over TCP, and to adapt the media encoding to match the dictates of the transport-layer congestion control algorithm. This is safe for the network, but can be suboptimal for the media quality unless the transport protocol is designed to support real-time media flows. We do not consider this class of applications further in this memo, as their network safety is guaranteed by the underlying transport.

Alternatively, RTP flows can be run over a non-congestion controlled transport protocol, for example UDP, performing rate adaptation at the application layer based on RTP Control Protocol (RTCP) feedback. With a well-designed, network-aware, application, this allows highly effective media quality adaptation, but there is potential to cause persistent congestion in the network if the application does not adapt its sending rate in a timely and effective manner. We consider this class of applications in this memo.

Congestion control relies on monitoring the delivery of a media flow, and responding to adapt the transmission of that flow when there are signs that the network path is congested. Network congestion can be detected in one of three ways: 1) a receiver can infer the onset of congestion by observing an increase in one-way delay caused by queue build-up within the network; 2) if Explicit Congestion Notification (ECN) [RFC3168] is supported, the network can signal the presence of congestion by marking packets using ECN Congestion Experienced (CE) marks (this could potentially be augmented by mechanisms such as ConEX [RFC7713], or other future protocol extensions for network signalling of congestion); or 3) in the extreme case, congestion will cause packet loss that can be detected by observing a gap in the received RTP sequence numbers.

Once the onset of congestion is observed, the receiver has to send feedback to the sender to indicate that the transmission rate needs to be reduced. How the sender reduces the transmission rate is highly dependent on the media codec being used, and is outside the scope of this memo.

There are several ways in which a receiver can send feedback to a media sender within the RTP framework:

o The base RTP specification [RFC3550] defines RTCP Reception Report (RR) packets to convey reception quality feedback information, and Sender Report (SR) packets to convey information about the media transmission. RTCP SR packets contain data that can be used to reconstruct media timing at a receiver, along with a count of the total number of octets and packets sent. RTCP RR packets report on the fraction of packets lost in the last reporting interval, the cumulative number of packets lost, the highest sequence number received, and the inter-arrival jitter. The RTCP RR packets also contain timing information that allows the sender to estimate the network round trip time (RTT) to the receivers. RTCP reports are sent periodically, with the reporting interval being determined by the number of SSRCs used in the session and a configured session bandwidth estimate (the number of synchronisation sources (SSRCs) used is usually two in a unicast session, one for each participant, but can be greater if the participants send multiple

media streams). The interval between reports sent from each receiver tends to be on the order of a few seconds on average, although it varies with the session bandwidth, and sub-second reporting intervals are possible in high bandwidth sessions, and it is randomised to avoid synchronisation of reports from multiple receivers. RTCP RR packets allow a receiver to report ongoing network congestion to the sender. However, if a receiver detects the onset of congestion part way through a reporting interval, the base RTP specification contains no provision for sending the RTCP RR packet early, and the receiver has to wait until the next scheduled reporting interval.

- The RTCP Extended Reports (XR) [RFC3611] allow reporting of more complex and sophisticated reception quality metrics, but do not change the RTCP timing rules. RTCP extended reports of potential interest for congestion control purposes are the extended packet loss, discard, and burst metrics [RFC3611], [RFC7002], [RFC7097], [RFC7003], [RFC6958]; and the extended delay metrics [RFC6843], [RFC6798]. Other RTCP Extended Reports that could be helpful for congestion control purposes might be developed in future.

- Rapid feedback about the occurrence of congestion events can be achieved using the Extended RTP Profile for RTCP-Based Feedback (RTP/AVPF) [RFC4585] (or its secure variant, RTP/SAVPF [RFC5124]) in place of the RTP/AVP profile [RFC3551]. This modifies the RTCP timing rules to allow RTCP reports to be sent early, in some cases immediately, provided the RTCP transmission rate keeps within its bandwidth allocation. It also defines transport-layer feedback messages, including negative acknowledgements (NACKs), that can be used to report on specific congestion events. RTP Codec Control Messages [RFC5104] extend the RTP/AVPF profile with additional feedback messages that can be used to influence that way in which rate adaptation occurs, but do not further change the dynamics of how rapidly feedback can be sent. Use of the RTP/AVPF profile is dependent on signalling.

- Finally, Explicit Congestion Notification (ECN) for RTP over UDP [RFC6679] can be used to provide feedback on the number of packets that received an ECN Congestion Experienced (CE) mark. This RTCP extension builds on the RTP/AVPF profile to allow rapid congestion feedback when ECN is supported.

In addition to these mechanisms for providing feedback, the sender can include an RTP header extension in each packet to record packet transmission times [RFC5450]. Accurate transmission timestamps can be helpful for estimating queuing delays, to get an early indication of the onset of congestion.

Taken together, these various mechanisms allow receivers to provide feedback on the senders when congestion events occur, with varying degrees of timeliness and accuracy. The key distinction is between systems that use only the basic RTCP mechanisms, without RTP/AVPF rapid feedback, and those that use the RTP/AVPF extensions to respond to congestion more rapidly.

3. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119]. This interpretation of these key words applies only when written in ALL CAPS. Mixed- or lower-case uses of these key words are not to be interpreted as carrying special significance in this memo.

The definition of the RTP circuit breaker is specified in terms of the following variables:

o Td is the deterministic RTCP reporting interval, as defined in Section 6.3.1 of [RFC3550].

o Tdr is the sender's estimate of the deterministic RTCP reporting interval, Td, calculated by a receiver of the data it is sending. Tdr is not known at the sender, but can be estimated by executing the algorithm in Section 6.2 of [RFC3550] using the average RTCP packet size seen at the sender, the number of members reported in the receiver's SR/RR report blocks, and whether the receiver is sending SR or RR packets. Tdr is recalculated when each new RTCP SR/RR report is received, but the media timeout circuit breaker (see Section 4.2) is only reconsidered when Tdr increases.

o Tr is the network round-trip time, calculated by the sender using the algorithm in Section 6.4.1 of [RFC3550] and smoothed using an exponentially weighted moving average as Tr = (0.8 * Tr) + (0.2 * Tr_new) where Tr_new is the latest RTT estimate obtained from an RTCP report. The weight is chosen so old estimates decay over k intervals.

o k is the non-reporting threshold (see Section 4.2).

o Tf is the media framing interval at the sender. For applications sending at a constant frame rate, Tf is the inter-frame interval. For applications that switch between a small set of possible frame rates, for example when sending speech with comfort noise, where comfort noise frames are sent less often than speech frames, Tf is set to the longest of the inter-frame intervals of the different frame rates. For applications that send periodic frames but

dynamically vary their frame rate, Tf is set to the largest inter-frame interval used in the last 10 seconds. For applications that send less than one frame every 10 seconds, or that have no concept of periodic frames (e.g., text conversation [RFC4103], or pointer events [RFC2862]), Tf is set to the time interval since the previous frame when each frame is sent.

- G is the frame group size. That is, the number of frames that are coded together based on a particular sending rate setting. If the codec used by the sender can change its rate on each frame, G = 1; otherwise G is set to the number of frames before the codec can adjust to the new rate. For codecs that have the concept of a group-of-pictures (GoP), G is likely the GoP length.

- T_rr_interval is the minimal interval between RTCP reports, as defined in Section 3.4 of [RFC4585]; it is only meaningful for implementations of RTP/AVPF profile [RFC4585] or the RTP/SAVPF profile [RFC5124].

- X is the estimated throughput a TCP connection would achieve over a path, in bytes per second.

- s is the size of RTP packets being sent, in bytes. If the RTP packets being sent vary in size, then the average size over the packet comprising the last 4 * G frames MUST be used (this is intended to be comparable to the four loss intervals used in [RFC5348]).

- p is the loss event rate, between 0.0 and 1.0, that would be seen by a TCP connection over a particular path. When used in the RTP congestion circuit breaker, this is approximated as described in Section 4.3.

- t_RTO is the retransmission timeout value that would be used by a TCP connection over a particular path, in seconds. This MUST be approximated using t_RTO = 4 * Tr when used as part of the RTP congestion circuit breaker.

- b is the number of packets that are acknowledged by a single TCP acknowledgement. Following [RFC5348], it is RECOMMENDED that the value b = 1 is used as part of the RTP congestion circuit breaker.

4. RTP Circuit Breakers for Systems Using the RTP/AVP Profile

The feedback mechanisms defined in [RFC3550] and available under the RTP/AVP profile [RFC3551] are the minimum that can be assumed for a baseline circuit breaker mechanism that is suitable for all unicast applications of RTP. Accordingly, for an RTP circuit breaker to be

useful, it needs to be able to detect that an RTP flow is causing excessive congestion using only basic RTCP features, without needing RTCP XR feedback or the RTP/AVPF profile for rapid RTCP reports.

RTCP is a fundamental part of the RTP protocol, and the mechanisms described here rely on the implementation of RTCP. Implementations that claim to support RTP, but that do not implement RTCP, will be unable to use the circuit breaker mechanisms described in this memo. Such implementations SHOULD NOT be used on networks that might be subject to congestion unless equivalent mechanisms are defined using some non-RTCP feedback channel to report congestion and signal circuit breaker conditions.

The RTCP timeout circuit breaker (Section 4.1) will trigger if an implementation of this memo attempts to interwork with an endpoint that does not support RTCP. Implementations that sometimes need to interwork with endpoints that do not support RTCP need to disable the RTP circuit breakers if they don't receive some confirmation via signalling that the remote endpoint implements RTCP (the presence of an SDP "a=rtcp:" attribute in an answer might be such an indication). The RTP Circuit Breaker SHOULD NOT be disabled on networks that might be subject to congestion, unless equivalent mechanisms are defined using some non-RTCP feedback channel to report congestion and signal circuit breaker conditions [I-D.ietf-tsvwg-circuit-breaker].

Three potential congestion signals are available from the basic RTCP SR/RR packets and are reported for each SSRC in the RTP session:

1. The sender can estimate the network round-trip time once per RTCP reporting interval, based on the contents and timing of RTCP SR and RR packets.

2. Receivers report a jitter estimate (the statistical variance of the RTP data packet inter-arrival time) calculated over the RTCP reporting interval. Due to the nature of the jitter calculation ([RFC3550], section 6.4.4), the jitter is only meaningful for RTP flows that send a single data packet for each RTP timestamp value (i.e., audio flows, or video flows where each packet comprises one video frame).

3. Receivers report the fraction of RTP data packets lost during the RTCP reporting interval, and the cumulative number of RTP packets lost over the entire RTP session.

These congestion signals limit the possible circuit breakers, since they give only limited visibility into the behaviour of the network.

RTT estimates are widely used in congestion control algorithms, as a proxy for queuing delay measures in delay-based congestion control or to determine connection timeouts. RTT estimates derived from RTCP SR and RR packets sent according to the RTP/AVP timing rules are too infrequent to be useful for congestion control, and don't give enough information to distinguish a delay change due to routing updates from queuing delay caused by congestion. Accordingly, we cannot use the RTT estimate alone as an RTP circuit breaker.

Increased jitter can be a signal of transient network congestion, but in the highly aggregated form reported in RTCP RR packets, it offers insufficient information to estimate the extent or persistence of congestion. Jitter reports are a useful early warning of potential network congestion, but provide an insufficiently strong signal to be used as a circuit breaker.

The remaining congestion signals are the packet loss fraction and the cumulative number of packets lost. If considered carefully, and over an appropriate time frame to distinguish transient problems from long term issues [I-D.ietf-tsvwg-circuit-breaker], these can be effective indicators that persistent excessive congestion is occurring in networks where packet loss is primarily due to queue overflows, although loss caused by non-congestive packet corruption can distort the result in some networks. TCP congestion control [RFC5681] intentionally tries to fill the router queues, and uses the resulting packet loss as congestion feedback. An RTP flow competing with TCP traffic will therefore expect to see a non-zero packet loss fraction, and some variation in queuing latency, in normal operation when sharing a path with other flows, that needs to be accounted for when determining the circuit breaker threshold [I-D.ietf-tsvwg-circuit-breaker]. This behaviour of TCP is reflected in the congestion circuit breaker below, and will affect the design of any RTP congestion control protocol.

Two packet loss regimes can be observed: 1) RTCP RR packets show a non-zero packet loss fraction, while the extended highest sequence number received continues to increment; and 2) RR packets show a loss fraction of zero, but the extended highest sequence number received does not increment even though the sender has been transmitting RTP data packets. The former corresponds to the TCP congestion avoidance state, and indicates a congested path that is still delivering data; the latter corresponds to a TCP timeout, and is most likely due to a path failure. A third condition is that data is being sent but no RTCP feedback is received at all, corresponding to a failure of the reverse path. We derive circuit breaker conditions for these loss regimes in the following.

4.1. RTP/AVP Circuit Breaker #1: RTCP Timeout

An RTCP timeout can occur when RTP data packets are being sent, but there are no RTCP reports returned from the receiver. This is either due to a failure of the receiver to send RTCP reports, or a failure of the return path that is preventing those RTCP reporting from being delivered. In either case, it is not safe to continue transmission, since the sender has no way of knowing if it is causing congestion.

An RTP sender that has not received any RTCP SR or RTCP RR packets reporting on the SSRC it is using, for a time period of at least three times its deterministic RTCP reporting interval, Td, without the randomization factor, and using the fixed minimum interval of Tmin=5 seconds, SHOULD cease transmission (see Section 4.5). The rationale for this choice of timeout is as described in Section 6.2 of [RFC3550] ("so that implementations which do not use the reduced value for transmitting RTCP packets are not timed out by other participants prematurely"), as updated by Section 6.1.4 of [I-D.ietf-avtcore-rtp-multi-stream] to account for the use of the RTP/AVPF profile [RFC4585] or the RTP/SAVPF profile [RFC5124].

To reduce the risk of premature timeout, implementations SHOULD NOT configure the RTCP bandwidth such that Td is larger than 5 seconds. Similarly, implementations that use the RTP/AVPF profile [RFC4585] or the RTP/SAVPF profile [RFC5124] SHOULD NOT configure T_rr_interval to values larger than 4 seconds (the reduced limit for T_rr_interval follows Section 6.1.3 of [I-D.ietf-avtcore-rtp-multi-stream]).

The choice of three RTCP reporting intervals as the timeout is made following Section 6.3.5 of RFC 3550 [RFC3550]. This specifies that participants in an RTP session will timeout and remove an RTP sender from the list of active RTP senders if no RTP data packets have been received from that RTP sender within the last two RTCP reporting intervals. Using a timeout of three RTCP reporting intervals is therefore large enough that the other participants will have timed out the sender if a network problem stops the data packets it is sending from reaching the receivers, even allowing for loss of some RTCP packets.

If a sender is transmitting a large number of RTP media streams, such that the corresponding RTCP SR or RR packets are too large to fit into the network MTU, the receiver will generate RTCP SR or RR packets in a round-robin manner. In this case, the sender SHOULD treat receipt of an RTCP SR or RR packet corresponding to any SSRC it sent on the same 5-tuple of source and destination IP address, port, and protocol, as an indication that the receiver and return path are working, preventing the RTCP timeout circuit breaker from triggering.

4.2. RTP/AVP Circuit Breaker #2: Media Timeout

If RTP data packets are being sent, but the RTCP SR or RR packets reporting on that SSRC indicate a non-increasing extended highest sequence number received, this is an indication that those RTP data packets are not reaching the receiver. This could be a short-term issue affecting only a few RTP packets, perhaps caused by a slow to open firewall or a transient connectivity problem, but if the issue persists, it is a sign of a more ongoing and significant problem (a "media timeout").

The time needed to declare a media timeout depends on the parameters Tdr, Tr, Tf, and on the non-reporting threshold k. The value of k is chosen so that when Tdr is large compared to Tr and Tf, receipt of at least k RTCP reports with non-increasing extended highest sequence number received gives reasonable assurance that the forward path has failed, and that the RTP data packets have not been lost by chance. The RECOMMENDED value for k is 5 reports.

When Tdr < Tf, then RTP data packets are being sent at a rate less than one per RTCP reporting interval of the receiver, so the extended highest sequence number received can be expected to be non-increasing for some receiver RTCP reporting intervals. Similarly, when Tdr < Tr, some receiver RTCP reporting intervals might pass before the RTP data packets arrive at the receiver, also leading to reports where the extended highest sequence number received is non-increasing. Both issues require the media timeout interval to be scaled relative to the threshold, k.

The media timeout RTP circuit breaker is therefore as follows. When starting sending, calculate MEDIA_TIMEOUT using:

```
MEDIA_TIMEOUT = ceil(k * max(Tf, Tr, Tdr) / Tdr)
```

When a sender receives an RTCP packet that indicates reception of the media it has been sending, then it cancels the media timeout circuit breaker. If it is still sending, then it MUST calculate a new value for MEDIA_TIMEOUT, and set a new media timeout circuit breaker.

If a sender receives an RTCP packet indicating that its media was not received, it MUST calculate a new value for MEDIA_TIMEOUT. If the new value is larger than the previous, it replaces MEDIA_TIMEOUT with the new value, extending the media timeout circuit breaker; otherwise it keeps the original value of MEDIA_TIMEOUT. This process is known as reconsidering the media timeout circuit breaker.

If MEDIA_TIMEOUT consecutive RTCP packets are received indicating that the media being sent was not received, and the media timeout

circuit breaker has not been cancelled, then the media timeout circuit breaker triggers. When the media timeout circuit breaker triggers, the sender SHOULD cease transmission (see Section 4.5).

When stopping sending an RTP stream, a sender MUST cancel the corresponding media timeout circuit breaker.

4.3. RTP/AVP Circuit Breaker #3: Congestion

If RTP data packets are being sent, and the corresponding RTCP SR or RR packets show non-zero packet loss fraction and increasing extended highest sequence number received, then those RTP data packets are arriving at the receiver, but some degree of congestion is occurring. The RTP/AVP profile [RFC3551] states that:

> If best-effort service is being used, RTP receivers SHOULD monitor packet loss to ensure that the packet loss rate is within acceptable parameters. Packet loss is considered acceptable if a TCP flow across the same network path and experiencing the same network conditions would achieve an average throughput, measured on a reasonable time scale, that is not less than the throughput the RTP flow is achieving. This condition can be satisfied by implementing congestion control mechanisms to adapt the transmission rate (or the number of layers subscribed for a layered multicast session), or by arranging for a receiver to leave the session if the loss rate is unacceptably high.
>
> The comparison to TCP cannot be specified exactly, but is intended as an "order-of-magnitude" comparison in time scale and throughput. The time scale on which TCP throughput is measured is the round-trip time of the connection. In essence, this requirement states that it is not acceptable to deploy an application (using RTP or any other transport protocol) on the best-effort Internet which consumes bandwidth arbitrarily and does not compete fairly with TCP within an order of magnitude.

The phase "order of magnitude" in the above means within a factor of ten, approximately. In order to implement this, it is necessary to estimate the throughput a bulk TCP connection would achieve over the path. For a long-lived TCP Reno connection, it has been shown that the TCP throughput, X, in bytes per second, can be estimated using [Padhye]:

$$X = \frac{s}{Tr*\mathrm{sqrt}(2*b*p/3)+(t\_RTO * (3*\mathrm{sqrt}(3*b*p/8) * p * (1+32*p*p)))}$$

This is the same approach to estimated TCP throughput that is used in [RFC5348]. Under conditions of low packet loss the second term on the denominator is small, so this formula can be approximated with reasonable accuracy as follows [Mathis]:

$$X = \frac{s}{Tr*\mathrm{sqrt}(2*b*p/3)}$$

It is RECOMMENDED that this simplified throughput equation be used, since the reduction in accuracy is small, and it is much simpler to calculate than the full equation. Measurements have shown that the simplified TCP throughput equation is effective as an RTP circuit breaker for multimedia flows sent to hosts on residential networks using ADSL and cable modem links [Singh]. The data shows that the full TCP throughput equation tends to be more sensitive to packet loss and triggers the RTP circuit breaker earlier than the simplified equation. Implementations that desire this extra sensitivity MAY use the full TCP throughput equation in the RTP circuit breaker. Initial measurements in LTE networks have shown that the extra sensitivity is helpful in that environment, with the full TCP throughput equation giving a more balanced circuit breaker response than the simplified TCP equation [Sarker]; other networks might see similar behaviour.

No matter what TCP throughput equation is chosen, two parameters need to be estimated and reported to the sender in order to calculate the throughput: the round trip time, Tr, and the loss event rate, p (the packet size, s, is known to the sender). The round trip time can be estimated from RTCP SR and RR packets. This is done too infrequently for accurate statistics, but is the best that can be done with the standard RTCP mechanisms.

Report blocks in RTCP SR or RR packets contain the packet loss fraction, rather than the loss event rate, so p cannot be reported (TCP typically treats the loss of multiple packets within a single RTT as one loss event, but RTCP RR packets report the overall fraction of packets lost, and does not report when the packet losses occurred). Using the loss fraction in place of the loss event rate can overestimate the loss. We believe that this overestimate will not be significant, given that we are only interested in order of magnitude comparison ([Floyd] section 3.2.1 shows that the difference is small for steady-state conditions and random loss, but using the loss fraction is more conservative in the case of bursty loss).

The congestion circuit breaker is therefore: when a sender that is transmitting at least one RTP packet every max(Tdr, Tr) seconds receives an RTCP SR or RR packet that contains a report block for an SSRC it is using, the sender MUST record the value of the fraction

lost field from the report block, and the time since the last report block was received, for that SSRC. If more than CB_INTERVAL (see below) report blocks have been received for that SSRC, the sender MUST calculate the average fraction lost over the last CB_INTERVAL reporting intervals, and then estimate the TCP throughput that would be achieved over the path using the chosen TCP throughput equation and the measured values of the round-trip time, Tr, the loss event rate, p (approximated by the average fraction lost, as is described below), and the packet size, s. The estimate of the TCP throughput, X, is then compared with the actual sending rate of the RTP stream. If the actual sending rate of the RTP stream is more than 10 * X, then the congestion circuit breaker is triggered.

The average fraction lost is calculated based on the sum, over the last CB_INTERVAL reporting intervals, of the fraction lost in each reporting interval multiplied by the duration of the corresponding reporting interval, divided by the total duration of the last CB_INTERVAL reporting intervals. The CB_INTERVAL parameter is set to:

```
CB_INTERVAL =
   ceil(3*min(max(10*G*Tf, 10*Tr, 3*Tdr), max(15, 3*Td))/(3*Tdr))
```

The parameters that feed into CB_INTERVAL are chosen to give the congestion control algorithm time to react to congestion. They give at least three RTCP reports, ten round trip times, and ten groups of frames to adjust the rate to reduce the congestion to a reasonable level. It is expected that a responsive congestion control algorithm will begin to respond with the next group of frames after it receives indication of congestion, so CB_INTERVAL ought to be a much longer interval than the congestion response.

If the RTP/AVPF profile [RFC4585] or the RTP/SAVPF [RFC5124] is used, and the T_rr_interval parameter is used to reduce the frequency of regular RTCP reports, then the value Tdr in the above expression for the CB_INTERVAL parameter MUST be replaced by max(T_rr_interval, Tdr).

The CB_INTERVAL parameter is calculated on joining the session, and recalculated on receipt of each RTCP packet, after checking whether the media timeout circuit breaker or the congestion circuit breaker has been triggered.

To ensure a timely response to persistent congestion, implementations SHOULD NOT configure the RTCP bandwidth such that Tdr is larger than 5 seconds. Similarly, implementations that use the RTP/AVPF profile [RFC4585] or the RTP/SAVPF profile [RFC5124] SHOULD NOT configure T_rr_interval to values larger than 4 seconds (the reduced limit for

T_rr_interval follows Section 6.1.3 of [I-D.ietf-avtcore-rtp-multi-stream]).

The rationale for enforcing a minimum sending rate below which the congestion circuit breaker will not trigger is to avoid spurious circuit breaker triggers when the number of packets sent per RTCP reporting interval is small, and hence the fraction lost samples are subject to measurement artefacts. The bound of at least one packet every max(Tdr, Tr) seconds is derived from the one packet per RTT minimum sending rate of TCP [RFC5405], adapted for use with RTP where the RTCP reporting interval is decoupled from the network RTT.

When the congestion circuit breaker is triggered, the sender SHOULD cease transmission (see Section 4.5). However, if the sender is able to reduce its sending rate by a factor of (approximately) ten, then it MAY first reduce its sending rate by this factor (or some larger amount) to see if that resolves the congestion. If the sending rate is reduced in this way and the congestion circuit breaker triggers again after the next CB_INTERVAL RTCP reporting intervals, the sender MUST then cease transmission. An example of such a rate reduction might be a video conferencing system that backs off to sending audio only, before completely dropping the call. If such a reduction in sending rate resolves the congestion problem, the sender MAY gradually increase the rate at which it sends data after a reasonable amount of time has passed, provided it takes care not to cause the problem to recur ("reasonable" is intentionally not defined here, since it depends on the application, media codec, and congestion control algorithm).

The RTCP reporting interval of the media sender does not affect how quickly congestion circuit breaker can trigger. The timing is based on the RTCP reporting interval of the receiver that generates the SR/RR packets from which the loss rate and RTT estimate are derived (note that RTCP requires all participants in a session to have similar reporting intervals, else the participant timeout rules in [RFC3550] will not work, so this interval is likely similar to that of the sender). If the incoming RTCP SR or RR packets are using a reduced minimum RTCP reporting interval (as specified in Section 6.2 of RFC 3550 [RFC3550] or the RTP/AVPF profile [RFC4585]), then that reduced RTCP reporting interval is used when determining if the circuit breaker is triggered.

If there are more media streams that can be reported in a single RTCP SR or RR packet, or if the size of a complete RTCP SR or RR packet exceeds the network MTU, then the receiver will report on a subset of sources in each reporting interval, with the subsets selected round-robin across multiple intervals so that all sources are eventually reported [RFC3550]. When generating such round-robin RTCP reports,

priority SHOULD be given to reports on sources that have high packet loss rates, to ensure that senders are aware of network congestion they are causing (this is an update to [RFC3550]).

4.4. RTP/AVP Circuit Breaker #4: Media Usability

Applications that use RTP are generally tolerant to some amount of packet loss. How much packet loss can be tolerated will depend on the application, media codec, and the amount of error correction and packet loss concealment that is applied. There is an upper bound on the amount of loss that can be corrected, however, beyond which the media becomes unusable. Similarly, many applications have some upper bound on the media capture to play-out latency that can be tolerated before the application becomes unusable. The latency bound will depend on the application, but typical values can range from the order of a few hundred milliseconds for voice telephony and interactive conferencing applications, up to several seconds for some video-on-demand systems.

As a final circuit breaker, RTP senders SHOULD monitor the reported packet loss and delay to estimate whether the media is likely to be suitable for the intended purpose. If the packet loss rate and/or latency is such that the media has become unusable, and has remained unusable for a significant time period, then the application SHOULD cease transmission. Similarly, receivers SHOULD monitor the quality of the media they receive, and if the quality is unusable for a significant time period, they SHOULD terminate the session. This memo intentionally does not define a bound on the packet loss rate or latency that will result in unusable media, as these are highly application dependent. Similarly, the time period that is considered significant is application dependent, but is likely on the order of seconds, or tens of seconds.

Sending media that suffers from such high packet loss or latency that it is unusable at the receiver is both wasteful of resources, and of no benefit to the user of the application. It also is highly likely to be congesting the network, and disrupting other applications. As such, the congestion circuit breaker will almost certainly trigger to stop flows where the media would be unusable due to high packet loss or latency. However, in pathological scenarios where the congestion circuit breaker does not stop the flow, it is desirable to prevent the application sending unnecessary traffic that might disrupt other uses of the network. The role of the media usability circuit breaker is to protect the network in such cases.

4.5. Ceasing Transmission

What it means to cease transmission depends on the application. The intention is that the application will stop sending RTP data packets on a particular 5-tuple (transport protocol, source and destination ports, source and destination IP addresses), until whatever network problem that triggered the RTP circuit breaker has dissipated. This could mean stopping a single RTP flow, or it could mean that multiple bundled RTP flows are stopped. RTP flows halted by the circuit breaker SHOULD NOT be restarted automatically unless the sender has received information that the congestion has dissipated, or can reasonably be expected to have dissipated. What could trigger this expectation is necessarily application dependent, but could be, for example, an indication that a competing flow has finished and freed up some capacity, or for an application running on a mobile device, that the device moved to a new location so the flow would traverse a different path if it were restarted. Ideally, a human user will be involved in the decision to try to restart the flow, since that user will eventually give up if the flows repeatedly trigger the circuit breaker. This will help avoid problems with automatic redial systems from congesting the network.

It is recognised that the RTP implementation in some systems might not be able to determine if a flow set-up request was initiated by a human user, or automatically by some scripted higher-level component of the system. These implementations MUST rate limit attempts to restart a flow on the same 5-tuple as used by a flow that triggered the circuit breaker, so that the reaction to a triggered circuit breaker lasts for at least the triggering interval [I-D.ietf-tsvwg-circuit-breaker].

The RTP circuit breaker will only trigger, and cease transmission, for media flows subject to long-term persistent congestion. Such flows are likely to have poor quality and usability for some time before the circuit breaker triggers. Implementations can monitor RTCP Reception Report blocks being returned for their media flows, and might find it beneficial to use this information to provide a user interface cue that problems are occurring, in advance of the circuit breaker triggering.

5. RTP Circuit Breakers and the RTP/AVPF and RTP/SAVPF Profiles

Use of the Extended RTP Profile for RTCP-based Feedback (RTP/AVPF) [RFC4585] allows receivers to send early RTCP reports in some cases, to inform the sender about particular events in the media stream. There are several use cases for such early RTCP reports, including providing rapid feedback to a sender about the onset of congestion. The RTP/SAVPF Profile [RFC5124] is a secure variant of the RTP/AVPF

profile, that is treated the same in the context of the RTP circuit breaker. These feedback profiles are often used with non-compound RTCP reports [RFC5506] to reduce the reporting overhead.

Receiving rapid feedback about congestion events potentially allows congestion control algorithms to be more responsive, and to better adapt the media transmission to the limitations of the network. It is expected that many RTP congestion control algorithms will adopt the RTP/AVPF profile or the RTP/SAVPF profile for this reason, defining new transport layer feedback reports that suit their requirements. Since these reports are not yet defined, and likely very specific to the details of the congestion control algorithm chosen, they cannot be used as part of the generic RTP circuit breaker.

Reduced-size RTCP reports sent under the RTP/AVPF early feedback rules that do not contain an RTCP SR or RR packet MUST be ignored by the congestion circuit breaker (they do not contain the information needed by the congestion circuit breaker algorithm), but MUST be counted as received packets for the RTCP timeout circuit breaker. Reduced-size RTCP reports sent under the RTP/AVPF early feedback rules that contain RTCP SR or RR packets MUST be processed by the congestion circuit breaker as if they were sent as regular RTCP reports, and counted towards the circuit breaker conditions specified in Section 4 of this memo. This will potentially make the RTP circuit breaker trigger earlier than it would if the RTP/AVPF profile was not used.

When using ECN with RTP (see Section 7), early RTCP feedback packets can contain ECN feedback reports. The count of ECN-CE marked packets contained in those ECN feedback reports is counted towards the number of lost packets reported if the ECN Feedback Report is sent in a compound RTCP packet along with an RTCP SR/RR report packet. Reports of ECN-CE packets sent as reduced-size RTCP ECN feedback packets without an RTCP SR/RR packet MUST be ignored.

These rules are intended to allow the use of low-overhead RTP/AVPF feedback for generic NACK messages without triggering the RTP circuit breaker. This is expected to make such feedback suitable for RTP congestion control algorithms that need to quickly report loss events in between regular RTCP reports. The reaction to reduced-size RTCP SR/RR packets is to allow such algorithms to send feedback that can trigger the circuit breaker, when desired.

The RTP/AVPF and RTP/SAVPF profiles include the T_rr_interval parameter that can be used to adjust the regular RTCP reporting interval. The use of the T_rr_interval parameter changes the behaviour of the RTP circuit breaker, as described in Section 4.

6. Impact of RTCP Extended Reports (XR)

RTCP Extended Report (XR) blocks provide additional reception quality metrics, but do not change the RTCP timing rules. Some of the RTCP XR blocks provide information that might be useful for congestion control purposes, others provide non-congestion-related metrics. With the exception of RTCP XR ECN Summary Reports (see Section 7), the presence of RTCP XR blocks in a compound RTCP packet does not affect the RTP circuit breaker algorithm. For consistency and ease of implementation, only the reception report blocks contained in RTCP SR packets, RTCP RR packets, or RTCP XR ECN Summary Report packets, are used by the RTP circuit breaker algorithm.

7. Impact of Explicit Congestion Notification (ECN)

The use of ECN for RTP flows does not affect the RTCP timeout circuit breaker (Section 4.1) or the media timeout circuit breaker (Section 4.2), since these are both connectivity checks that simply determinate if any packets are being received.

There is no consensus on what would be the correct response of the congestion circuit breaker (Section 4.3) to ECN-CE marked packets. The guidelines in [RFC3168] and [RFC6679] are that the response to receipt of an ECN-CE marked packet needs to be essentially the same as the response to a lost packet for congestion control purposes. Since the RTP congestion circuit breaker responds to the same congestion signals, this suggests that it ought to consider ECN-CE marked packets as lost packets when calculating the TCP throughput estimate to determine if the congestion circuit breaker triggers.

More recent work, however, has suggested that the response to an ECN-CE mark ought to be less severe than the response to packet loss. For example, the TCP ABE proposal [I-D.khademi-tcpm-alternativebackoff-ecn] makes the argument that TCP congestion control ought to back-off less in response to an ECN-CE mark than to packet loss, because networks that generate ECN-CE marks tend to use AQM schemes with much smaller buffers. For RTP congestion control, both NADA [I-D.ietf-rmcat-nada] and SCREAM [I-D.ietf-rmcat-scream-cc] suggest responding differently to ECN-CE marked packets than to lost packets, for quality of experience reasons, but make different proposals for how the response ought to change. Such proposals would imply that a different circuit breaker threshold be used for congestion signalled by ECN-CE marks than for congestion signalled by packet loss, but unfortunately they offer no clear guidance on how the threshold ought to be changed.

Finally, there are suggestions that forthcoming AQM proposals [I-D.briscoe-aqm-dualq-coupled] might mark packets with ECN-CE in a

significantly more aggressive manner that at present. Any such deployment would likely be incompatible with deployed TCP implementations, so is not a short-term issue, but would require significant changes to the congestion circuit breaker response.

Given the above issues, implementations MAY ignore ECN-CE marks when determining if the congestion circuit breaker triggers, since excessive persistent congestion will eventually lead to packet loss that will trigger the circuit breaker. Doing this will protect the network from congestion collapse, but might result in sub-optimal user experience for competing flows that share the bottleneck queue, since that queue will be driven to overflow, inducing high latency. If this is a concern, the only current guidance is for implementations to treat ECN-CE marked packets as equivalent to lost packets, whilst being aware that this might trigger the circuit breaker prematurely in future, depending on how AQM and ECN deployment evolves. Developers that implement a circuit breaker based on ECN-CE marks will need to track future developments in AQM standards and deployed ECN marking behaviour, and ensure their implementations are updated to match.

For the media usability circuit breaker (Section 4.4), ECN-CE marked packets arrive at the receiver, and if they arrive in time, they will be decoded and rendered as normal. Accordingly, receipt of such packets ought not affect the usability of the media, and the arrival of RTCP feedback indicating their receipt is not expected to impact the operation of the media usability circuit breaker.

8. Impact of Bundled Media and Layered Coding

The RTP circuit breaker operates on a per-RTP session basis. An RTP sender that participates in several RTP sessions MUST treat each RTP session independently with regards to the RTP circuit breaker.

An RTP sender can generate several media streams within a single RTP session, with each stream using a different SSRC. This can happen if bundled media are in use, when using simulcast, or when using layered media coding. By default, each SSRC will be treated independently by the RTP circuit breaker. However, the sender MAY choose to treat the flows (or a subset thereof) as a group, such that a circuit breaker trigger for one flow applies to the group of flows as a whole, and either causes the entire group to cease transmission, or the sending rate of the group to reduce by a factor of ten, depending on the RTP circuit breaker triggered. Grouping flows in this way is expected to be especially useful for layered flows sent using multiple SSRCs, as it allows the layered flow to react as a whole, ceasing transmission on the enhancement layers first to reduce sending rate if necessary, rather than treating each layer independently. Care needs to be

taken if the different media streams sent on a single transport layer flow use different DSCP values [RFC7657], [I-D.ietf-tsvwg-rtcweb-qos], since congestion could be experienced differently depending on the DSCP marking. Accordingly, RTP media streams with different DSCP values SHOULD NOT be considered as a group when evaluating the RTP Circuit Breaker conditions.

9. Security Considerations

The security considerations of [RFC3550] apply.

If the RTP/AVPF profile is used to provide rapid RTCP feedback, the security considerations of [RFC4585] apply. If ECN feedback for RTP over UDP/IP is used, the security considerations of [RFC6679] apply.

If non-authenticated RTCP reports are used, an on-path attacker can trivially generate fake RTCP packets that indicate high packet loss rates, causing the circuit breaker to trigger and disrupt an RTP session. This is somewhat more difficult for an off-path attacker, due to the need to guess the randomly chosen RTP SSRC value and the RTP sequence number. This attack can be avoided if RTCP packets are authenticated; authentication options are discussed in [RFC7201].

Timely operation of the RTP circuit breaker depends on the choice of RTCP reporting interval. If the receiver has a reporting interval that is overly long, then the responsiveness of the circuit breaker decreases. In the limit, the RTP circuit breaker can be disabled for all practical purposes by configuring an RTCP reporting interval that is many minutes duration. This issue is not specific to the circuit breaker: long RTCP reporting intervals also prevent reception quality reports, feedback messages, codec control messages, etc., from being used. Implementations are expected to impose an upper limit on the RTCP reporting interval they are willing to negotiate (based on the session bandwidth and RTCP bandwidth fraction) when using the RTP circuit breaker, as discussed in Section 4.3.

10. IANA Considerations

There are no actions for IANA.

11. Acknowledgements

The authors would like to thank Bernard Aboba, Harald Alvestrand, Ben Campbell, Alissa Cooper, Spencer Dawkins, Gorry Fairhurst, Stephen Farrell, Nazila Fough, Kevin Gross, Cullen Jennings, Randell Jesup, Mirja Kuehlewind, Jonathan Lennox, Matt Mathis, Stephen McQuistin, Simon Perreault, Eric Rescorla, Abheek Saha, Meral Shirazipour, Fabio Verdicchio, and Magnus Westerlund for their valuable feedback.

12. References

12.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550, July 2003, <http://www.rfc-editor.org/info/rfc3550>.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, DOI 10.17487/RFC3551, July 2003, <http://www.rfc-editor.org/info/rfc3551>.

[RFC3611] Friedman, T., Ed., Caceres, R., Ed., and A. Clark, Ed., "RTP Control Protocol Extended Reports (RTCP XR)", RFC 3611, DOI 10.17487/RFC3611, November 2003, <http://www.rfc-editor.org/info/rfc3611>.

[RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey, "Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585, DOI 10.17487/RFC4585, July 2006, <http://www.rfc-editor.org/info/rfc4585>.

[RFC5348] Floyd, S., Handley, M., Padhye, J., and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification", RFC 5348, DOI 10.17487/RFC5348, September 2008, <http://www.rfc-editor.org/info/rfc5348>.

[RFC6679] Westerlund, M., Johansson, I., Perkins, C., O'Hanlon, P., and K. Carlberg, "Explicit Congestion Notification (ECN) for RTP over UDP", RFC 6679, DOI 10.17487/RFC6679, August 2012, <http://www.rfc-editor.org/info/rfc6679>.

12.2. Informative References

[Floyd] Floyd, S., Handley, M., Padhye, J., and J. Widmer, "Equation-Based Congestion Control for Unicast Applications", Proceedings of the ACM SIGCOMM conference, 2000, DOI 10.1145/347059.347397, August 2000.

[I-D.briscoe-aqm-dualq-coupled]
Schepper, K., Briscoe, B., Bondarenko, O., and I. Tsang, "DualQ Coupled AQM for Low Latency, Low Loss and Scalable Throughput", draft-briscoe-aqm-dualq-coupled-01 (work in progress), March 2016.

[I-D.ietf-avtcore-rtp-multi-stream]
Lennox, J., Westerlund, M., Wu, Q., and C. Perkins, "Sending Multiple RTP Streams in a Single RTP Session", draft-ietf-avtcore-rtp-multi-stream-11 (work in progress), December 2015.

[I-D.ietf-rmcat-nada]
Zhu, X., Pan, R., Ramalho, M., Cruz, S., Jones, P., Fu, J., D'Aronco, S., and C. Ganzhorn, "NADA: A Unified Congestion Control Scheme for Real-Time Media", draft-ietf-rmcat-nada-02 (work in progress), March 2016.

[I-D.ietf-rmcat-scream-cc]
Johansson, I. and Z. Sarker, "Self-Clocked Rate Adaptation for Multimedia", draft-ietf-rmcat-scream-cc-04 (work in progress), June 2016.

[I-D.ietf-tsvwg-circuit-breaker]
Fairhurst, G., "Network Transport Circuit Breakers", draft-ietf-tsvwg-circuit-breaker-15 (work in progress), April 2016.

[I-D.ietf-tsvwg-rtcweb-qos]
Jones, P., Dhesikan, S., Jennings, C., and D. Druta, "DSCP Packet Markings for WebRTC QoS", draft-ietf-tsvwg-rtcweb-qos-17 (work in progress), May 2016.

[I-D.khademi-tcpm-alternativebackoff-ecn]
Khademi, N., Welzl, M., Armitage, G., and G. Fairhurst, "TCP Alternative Backoff with ECN (ABE)", draft-khademi-tcpm-alternativebackoff-ecn-00 (work in progress), May 2016.

[Mathis] Mathis, M., Semke, J., Mahdavi, J., and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm", ACM SIGCOMM Computer Communication Review 27(3), DOI 10.1145/263932.264023, July 1997.

[Padhye] Padhye, J., Firoiu, V., Towsley, D., and J. Kurose, "Modeling TCP Throughput: A Simple Model and its Empirical Validation", Proceedings of the ACM SIGCOMM conference, 1998, DOI 10.1145/285237.285291, August 1998.

[RFC2862] Civanlar, M. and G. Cash, "RTP Payload Format for Real-Time Pointers", RFC 2862, DOI 10.17487/RFC2862, June 2000, <http://www.rfc-editor.org/info/rfc2862>.

[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, <http://www.rfc-editor.org/info/rfc3168>.

[RFC4103] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, DOI 10.17487/RFC4103, June 2005, <http://www.rfc-editor.org/info/rfc4103>.

[RFC5104] Wenger, S., Chandra, U., Westerlund, M., and B. Burman, "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)", RFC 5104, DOI 10.17487/RFC5104, February 2008, <http://www.rfc-editor.org/info/rfc5104>.

[RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124, February 2008, <http://www.rfc-editor.org/info/rfc5124>.

[RFC5405] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", BCP 145, RFC 5405, DOI 10.17487/RFC5405, November 2008, <http://www.rfc-editor.org/info/rfc5405>.

[RFC5450] Singer, D. and H. Desineni, "Transmission Time Offsets in RTP Streams", RFC 5450, DOI 10.17487/RFC5450, March 2009, <http://www.rfc-editor.org/info/rfc5450>.

[RFC5506] Johansson, I. and M. Westerlund, "Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences", RFC 5506, DOI 10.17487/RFC5506, April 2009, <http://www.rfc-editor.org/info/rfc5506>.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, DOI 10.17487/RFC5681, September 2009, <http://www.rfc-editor.org/info/rfc5681>.

[RFC6798] Clark, A. and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Packet Delay Variation Metric Reporting", RFC 6798, DOI 10.17487/RFC6798, November 2012, <http://www.rfc-editor.org/info/rfc6798>.

[RFC6843] Clark, A., Gross, K., and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Delay Metric Reporting", RFC 6843, DOI 10.17487/RFC6843, January 2013, <http://www.rfc-editor.org/info/rfc6843>.

[RFC6958] Clark, A., Zhang, S., Zhao, J., and Q. Wu, Ed., "RTP Control Protocol (RTCP) Extended Report (XR) Block for Burst/Gap Loss Metric Reporting", RFC 6958, DOI 10.17487/RFC6958, May 2013, <http://www.rfc-editor.org/info/rfc6958>.

[RFC7002] Clark, A., Zorn, G., and Q. Wu, "RTP Control Protocol (RTCP) Extended Report (XR) Block for Discard Count Metric Reporting", RFC 7002, DOI 10.17487/RFC7002, September 2013, <http://www.rfc-editor.org/info/rfc7002>.

[RFC7003] Clark, A., Huang, R., and Q. Wu, Ed., "RTP Control Protocol (RTCP) Extended Report (XR) Block for Burst/Gap Discard Metric Reporting", RFC 7003, DOI 10.17487/RFC7003, September 2013, <http://www.rfc-editor.org/info/rfc7003>.

[RFC7097] Ott, J., Singh, V., Ed., and I. Curcio, "RTP Control Protocol (RTCP) Extended Report (XR) for RLE of Discarded Packets", RFC 7097, DOI 10.17487/RFC7097, January 2014, <http://www.rfc-editor.org/info/rfc7097>.

[RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP Sessions", RFC 7201, DOI 10.17487/RFC7201, April 2014, <http://www.rfc-editor.org/info/rfc7201>.

[RFC7657] Black, D., Ed. and P. Jones, "Differentiated Services (Diffserv) and Real-Time Communication", RFC 7657, DOI 10.17487/RFC7657, November 2015, <http://www.rfc-editor.org/info/rfc7657>.

[RFC7713] Mathis, M. and B. Briscoe, "Congestion Exposure (ConEx) Concepts, Abstract Mechanism, and Requirements", RFC 7713, DOI 10.17487/RFC7713, December 2015, <http://www.rfc-editor.org/info/rfc7713>.

[Sarker] Sarker, Z., Singh, V., and C. Perkins, "An Evaluation of RTP Circuit Breaker Performance on LTE Networks", Proceedings of the IEEE Infocom workshop on Communication and Networking Techniques for Contemporary Video, 2014, April 2014.

[Singh] Singh, V., McQuistin, S., Ellis, M., and C. Perkins, "Circuit Breakers for Multimedia Congestion Control", Proceedings of the International Packet Video Workshop, 2013, DOI 10.1109/PV.2013.6691439, December 2013.

Authors' Addresses

Colin Perkins
University of Glasgow
School of Computing Science
Glasgow G12 8QQ
United Kingdom

Email: csp@csperkins.org

Varun Singh
Nemu Dialogue Systems Oy
Runeberginkatu 4c A 4
Helsinki 00100
Finland

Email: varun.singh@iki.fi
URI: http://www.callstats.io/