Initial Performance Metric Registry Entries

draft-ietf-ippm-initial-registry-07

Abstract

This memo defines the Initial Entries for the Performance Metrics Registry. This version includes:

- Revised implementation of Passive TCP RTT metrics in section 10 (from comments).
- remaining question on DNS measurement method(s)

Still need: Add MBM metric entry.

Requirements Language

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.

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."
1. Introduction ......................................................... 7

2. Scope .............................................................. 8

3. Registry Categories and Columns ................................. 8

4. UDP Round-trip Latency and Loss Registry Entries ............. 9
   4.1. Summary ...................................................... 10
       4.1.1. ID (Identifier) ....................................... 10
       4.1.2. Name .................................................. 10
       4.1.3. URIs .................................................. 10
       4.1.4. Description ........................................... 10
       4.1.5. Change Controller .................................... 10
       4.1.6. Version (of Registry Format) ......................... 10
   4.2. Metric Definition ............................................ 11
       4.2.1. Reference Definition ................................ 11
       4.2.2. Fixed Parameters .................................... 12
   4.3. Method of Measurement ....................................... 12
       4.3.1. Reference Method .................................... 13
       4.3.2. Packet Stream Generation ............................. 14
       4.3.3. Traffic Filtering (observation) Details ............... 14
       4.3.4. Sampling Distribution ................................ 15
       4.3.5. Run-time Parameters and Data Format ................. 15
       4.3.6. Roles ................................................ 15
   4.4. Output ....................................................... 16
       4.4.1. Type .................................................. 16
       4.4.2. Reference Definition ................................ 16
       4.4.3. Metric Units ......................................... 17
       4.4.4. Calibration ........................................... 17
   4.5. Administrative items ......................................... 18
       4.5.1. Status ................................................ 18
       4.5.2. Requestor (keep?) .................................... 18
4.5.3. Revision .................................................. 18
4.5.4. Revision Date .............................................. 18
4.6. Comments and Remarks ....................................... 18
5. Packet Delay Variation Registry Entry ......................... 18
  5.1. Summary .................................................... 18
  5.1.1. ID (Identifier) .......................................... 18
  5.1.2. Name ..................................................... 18
  5.1.3. URIs ..................................................... 19
  5.1.4. Description .............................................. 19
  5.1.5. Change Controller ........................................ 19
  5.1.6. Version (of Registry Format) ............................. 19
  5.2. Metric Definition ........................................... 19
  5.2.1. Reference Definition .................................... 19
  5.2.2. Fixed Parameters ........................................ 20
  5.3. Method of Measurement ..................................... 21
  5.3.1. Reference Method ......................................... 21
  5.3.2. Packet Stream Generation ................................ 22
  5.3.3. Traffic Filtering (observation) Details ................ 22
  5.3.4. Sampling Distribution ................................. 23
  5.3.5. Run-time Parameters and Data Format ..................... 23
  5.3.6. Roles ................................................... 23
  5.4. Output ..................................................... 23
  5.4.1. Type .................................................... 24
  5.4.2. Reference Definition .................................... 24
  5.4.3. Metric Units ............................................ 24
  5.4.4. Calibration ............................................. 25
  5.5. Administrative items ...................................... 25
  5.5.1. Status .................................................. 25
  5.5.2. Requestor (keep?) ....................................... 26
  5.5.3. Revision ................................................ 26
  5.5.4. Revision Date ........................................... 26
  5.5.5. Requestor .............................................. 27
  5.6. Comments and Remarks ....................................... 26
6. DNS Response Latency and Loss Registry Entries ............... 26
  6.1. Summary .................................................... 26
  6.1.1. ID (Identifier) .......................................... 27
  6.1.2. Name ..................................................... 27
  6.1.3. URI ..................................................... 27
  6.1.4. Description .............................................. 27
  6.1.5. Change Controller ........................................ 27
  6.1.6. Version (of Registry Format) ............................. 27
  6.2. Metric Definition ........................................... 27
  6.2.1. Reference Definition .................................... 28
  6.2.2. Fixed Parameters ........................................ 28
  6.3. Method of Measurement ..................................... 30
  6.3.1. Reference Method ......................................... 30
  6.3.2. Packet Stream Generation ................................ 32
  6.3.3. Traffic Filtering (observation) Details ................ 32
  6.3.4. Sampling Distribution ................................. 32
6.3.5. Run-time Parameters and Data Format .................... 32
6.3.6. Roles ............................................. 34
6.4. Output ............................................... 34
  6.4.1. Type .............................................. 34
  6.4.2. Reference Definition ............................... 34
  6.4.3. Metric Units ..................................... 35
  6.4.4. Calibration ...................................... 35
6.5. Administrative items ......................................... 35
  6.5.1. Status ............................................ 35
  6.5.2. Requestor ........................................ 35
  6.5.3. Revision .......................................... 35
  6.5.4. Revision Date .................................... 36
6.6. Comments and Remarks .......................................... 36
7. UDP Poisson One-way Delay and Loss Registry Entries .......... 36
  7.1. Summary ............................................. 36
    7.1.1. ID (Identifier) ................................. 36
    7.1.2. Name ............................................ 36
    7.1.3. URI and URL .................................... 37
    7.1.4. Description .................................... 37
  7.2. Metric Definition ........................................ 37
    7.2.1. Reference Definition ............................. 38
    7.2.2. Fixed Parameters ................................. 38
  7.3. Method of Measurement ....................................... 39
    7.3.1. Reference Method ................................ 40
    7.3.2. Packet Stream Generation ......................... 40
    7.3.3. Traffic Filtering (observation) Details ............. 41
    7.3.4. Sampling Distribution ............................ 41
    7.3.5. Run-time Parameters and Data Format .................. 41
    7.3.6. Roles ........................................... 42
  7.4. Output ................................................ 42
    7.4.1. Type ............................................. 42
    7.4.2. Reference Definition ............................. 42
    7.4.3. Metric Units .................................... 45
    7.4.4. Calibration ...................................... 45
  7.5. Administrative items ....................................... 46
    7.5.1. Status ........................................... 46
    7.5.2. Requestor (keep?) ................................ 46
    7.5.3. Revision ......................................... 46
    7.5.4. Revision Date .................................... 47
  7.6. Comments and Remarks ....................................... 47
8. UDP Periodic One-way Delay and Loss Registry Entries .......... 47
  8.1. Summary ............................................... 47
    8.1.1. ID (Identifier) ................................. 47
    8.1.2. Name ............................................. 47
    8.1.3. URIs ............................................. 48
    8.1.4. Description ..................................... 48
  8.2. Metric Definition ........................................ 48
    8.2.1. Reference Definition ............................. 49
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.2.2.</td>
<td>Fixed Parameters</td>
<td>49</td>
</tr>
<tr>
<td>8.3.</td>
<td>Method of Measurement</td>
<td>50</td>
</tr>
<tr>
<td>8.3.1.</td>
<td>Reference Method</td>
<td>51</td>
</tr>
<tr>
<td>8.3.2.</td>
<td>Packet Stream Generation</td>
<td>51</td>
</tr>
<tr>
<td>8.3.3.</td>
<td>Traffic Filtering (observation) Details</td>
<td>52</td>
</tr>
<tr>
<td>8.3.4.</td>
<td>Sampling Distribution</td>
<td>52</td>
</tr>
<tr>
<td>8.3.5.</td>
<td>Run-time Parameters and Data Format</td>
<td>52</td>
</tr>
<tr>
<td>8.3.6.</td>
<td>Roles</td>
<td>53</td>
</tr>
<tr>
<td>8.4.</td>
<td>Output</td>
<td>53</td>
</tr>
<tr>
<td>8.4.1.</td>
<td>Type</td>
<td>53</td>
</tr>
<tr>
<td>8.4.2.</td>
<td>Reference Definition</td>
<td>53</td>
</tr>
<tr>
<td>8.4.3.</td>
<td>Metric Units</td>
<td>56</td>
</tr>
<tr>
<td>8.4.4.</td>
<td>Calibration</td>
<td>56</td>
</tr>
<tr>
<td>8.5.</td>
<td>Administrative items</td>
<td>57</td>
</tr>
<tr>
<td>8.5.1.</td>
<td>Status</td>
<td>57</td>
</tr>
<tr>
<td>8.5.2.</td>
<td>Requestor (keep?)</td>
<td>57</td>
</tr>
<tr>
<td>8.5.3.</td>
<td>Revision</td>
<td>57</td>
</tr>
<tr>
<td>8.5.4.</td>
<td>Revision Date</td>
<td>58</td>
</tr>
<tr>
<td>8.6.</td>
<td>Comments and Remarks</td>
<td>58</td>
</tr>
<tr>
<td>9.</td>
<td>ICMP Round-trip Latency and Loss Registry Entries</td>
<td>58</td>
</tr>
<tr>
<td>9.1.</td>
<td>Summary</td>
<td>58</td>
</tr>
<tr>
<td>9.1.1.</td>
<td>ID (Identifier)</td>
<td>58</td>
</tr>
<tr>
<td>9.1.2.</td>
<td>Name</td>
<td>58</td>
</tr>
<tr>
<td>9.1.3.</td>
<td>URIs</td>
<td>59</td>
</tr>
<tr>
<td>9.1.4.</td>
<td>Description</td>
<td>59</td>
</tr>
<tr>
<td>9.1.5.</td>
<td>Change Controller</td>
<td>59</td>
</tr>
<tr>
<td>9.1.6.</td>
<td>Version (of Registry Format)</td>
<td>59</td>
</tr>
<tr>
<td>9.2.</td>
<td>Metric Definition</td>
<td>59</td>
</tr>
<tr>
<td>9.2.1.</td>
<td>Reference Definition</td>
<td>59</td>
</tr>
<tr>
<td>9.2.2.</td>
<td>Fixed Parameters</td>
<td>60</td>
</tr>
<tr>
<td>9.3.</td>
<td>Method of Measurement</td>
<td>61</td>
</tr>
<tr>
<td>9.3.1.</td>
<td>Reference Method</td>
<td>61</td>
</tr>
<tr>
<td>9.3.2.</td>
<td>Packet Stream Generation</td>
<td>62</td>
</tr>
<tr>
<td>9.3.3.</td>
<td>Traffic Filtering (observation) Details</td>
<td>63</td>
</tr>
<tr>
<td>9.3.4.</td>
<td>Sampling Distribution</td>
<td>63</td>
</tr>
<tr>
<td>9.3.5.</td>
<td>Run-time Parameters and Data Format</td>
<td>63</td>
</tr>
<tr>
<td>9.3.6.</td>
<td>Roles</td>
<td>64</td>
</tr>
<tr>
<td>9.4.</td>
<td>Output</td>
<td>64</td>
</tr>
<tr>
<td>9.4.1.</td>
<td>Type</td>
<td>64</td>
</tr>
<tr>
<td>9.4.2.</td>
<td>Reference Definition</td>
<td>64</td>
</tr>
<tr>
<td>9.4.3.</td>
<td>Metric Units</td>
<td>66</td>
</tr>
<tr>
<td>9.4.4.</td>
<td>Calibration</td>
<td>67</td>
</tr>
<tr>
<td>9.5.</td>
<td>Administrative items</td>
<td>67</td>
</tr>
<tr>
<td>9.5.1.</td>
<td>Status</td>
<td>67</td>
</tr>
<tr>
<td>9.5.2.</td>
<td>Requestor (keep?)</td>
<td>67</td>
</tr>
<tr>
<td>9.5.3.</td>
<td>Revision</td>
<td>67</td>
</tr>
<tr>
<td>9.5.4.</td>
<td>Revision Date</td>
<td>67</td>
</tr>
<tr>
<td>9.6.</td>
<td>Comments and Remarks</td>
<td>67</td>
</tr>
</tbody>
</table>
10. TCP Round-Trip Delay and Loss Registry Entries

10.1. Summary

10.1.1. ID (Identifier)

10.1.2. Name

10.1.3. URIs

10.1.4. Description

10.1.5. Change Controller

10.1.6. Version (of Registry Format)

10.2. Metric Definition

10.2.1. Reference Definitions

10.2.2. Fixed Parameters

10.3. Method of Measurement

10.3.1. Reference Methods

10.3.2. Packet Stream Generation

10.3.3. Traffic Filtering (observation) Details

10.3.4. Sampling Distribution

10.3.5. Run-time Parameters and Data Format

10.3.6. Roles

10.4. Output

10.4.1. Type

10.4.2. Reference Definition

10.4.3. Metric Units

10.4.4. Calibration

10.5. Administrative items

10.5.1. Status

10.5.2. Requestor (keep?)

10.5.3. Revision

10.5.4. Revision Date

10.5.6. Comments and Remarks

11. ver08 BLANK Registry Entry

11.1. Summary

11.1.1. ID (Identifier)

11.1.2. Name

11.1.3. URIs

11.1.4. Description

11.1.5. Reference

11.1.6. Change Controller

11.1.7. Version (of Registry Format)

11.2. Metric Definition

11.2.1. Reference Definition

11.2.2. Fixed Parameters

11.3. Method of Measurement

11.3.1. Reference Method

11.3.2. Packet Stream Generation

11.3.3. Traffic Filtering (observation) Details

11.3.4. Sampling Distribution

11.3.5. Run-time Parameters and Data Format

11.3.6. Roles
11.4. Output ...................................................... 81
11.4.1. Type ....................................................... 81
11.4.2. Reference Definition ................................. 81
11.4.3. Metric Units ............................................. 81
11.4.4. Calibration .............................................. 81
11.5. Administrative items ................................. 82
11.5.1. Status ................................................... 82
11.5.2. Requestor ............................................... 82
11.5.3. Revision ............................................... 82
11.5.4. Revision Date ........................................... 82
11.6. Comments and Remarks ............................. 82
12. Example RTCP-XR Registry Entry ..................... 82
12.1. Registry Indexes ........................................ 82
  12.1.1. Identifier ........................................... 82
  12.1.2. Name .................................................. 82
  12.1.3. URI .................................................... 83
  12.1.4. Status ................................................ 83
  12.1.5. Requestor ............................................. 83
  12.1.6. Revision .............................................. 83
  12.1.7. Revision Date ....................................... 83
  12.1.8. Description ........................................... 83
  12.1.9. Reference Specification(s) ......................... 83
12.2. Metric Definition ..................................... 83
  12.2.1. Reference Definition ................................ 83
  12.2.2. Fixed Parameters .................................... 84
12.3. Method of Measurement ................................ 84
  12.3.1. Reference Method .................................... 84
  12.3.2. Stream Type and Stream Parameters ............... 85
  12.3.3. Output Type and Data Format ...................... 85
  12.3.4. Metric Units ......................................... 85
  12.3.5. Run-time Parameters and Data Format ............ 85
12.4. Comments and Remarks ................................ 87
13. Revision History ......................................... 87
14. Security Considerations .............................. 88
15. IANA Considerations ...................................... 88
16. Acknowledgements ....................................... 88
17. References ................................................. 88
  17.1. Normative References .............................. 88
  17.2. Informative References ........................... 91
Authors’ Addresses ........................................ 92

1. Introduction

Note: Efforts to synchronize structure and terminology with
[I-D.ietf-ippm-metric-registry] will likely be incomplete until both
drafts are stable.
This memo proposes an initial set of entries for the Performance Metric Registry. It uses terms and definitions from the IPPM literature, primarily [RFC2330].

Although there are several standard templates for organizing specifications of performance metrics (see [RFC2679] for an example of the traditional IPPM template, based to large extent on the Benchmarking Methodology Working Group’s traditional template in [RFC1242], and see [RFC6390] for a similar template), none of these templates were intended to become the basis for the columns of an IETF-wide registry of metrics. While examining aspects of metric specifications which need to be registered, it became clear that none of the existing metric templates fully satisfies the particular needs of a registry.

Therefore, [I-D.ietf-ippm-metric-registry] defines the overall format for a Performance Metric Registry. Section 5 of [I-D.ietf-ippm-metric-registry] also gives guidelines for those requesting registration of a Metric, that is the creation of entry(s) in the Performance Metric Registry: "In essence, there needs to be evidence that a candidate Registered Performance Metric has significant industry interest, or has seen deployment, and there is agreement that the candidate Registered Performance Metric serves its intended purpose." The process in [I-D.ietf-ippm-metric-registry] also requires that new entries are administered by IANA through Expert Review, which will ensure that the metrics are tightly defined.

2. Scope

This document defines the initial set of Performance Metrics Registry entries, for which IETF approval (following development in the IP Performance Metrics (IPPM) Working Group) will satisfy the requirement for Expert Review. Most are Active Performance Metrics, which are based on RFCs prepared in the IPPM working group of the IETF, according to their framework [RFC2330] and its updates.

3. Registry Categories and Columns

This section provides the categories and columns of the registry, for easy reference. An entry (row) therefore gives a complete description of a Registered Metric.
Registry Categories and Columns, shown as

<table>
<thead>
<tr>
<th>Category</th>
<th>Column</th>
<th>Column</th>
<th>Category</th>
</tr>
</thead>
</table>

Summary

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Name</th>
<th>URIs</th>
<th>Desc.</th>
<th>Reference</th>
<th>Change Controller</th>
<th>Ver</th>
</tr>
</thead>
</table>

Metric Definition

<table>
<thead>
<tr>
<th>Reference Definition</th>
<th>Fixed Parameters</th>
</tr>
</thead>
</table>

Method of Measurement

<table>
<thead>
<tr>
<th>Method</th>
<th>Reference</th>
<th>Packet</th>
<th>Traffic</th>
<th>Sampling</th>
<th>Run-time</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Method</td>
<td>Stream</td>
<td>Filter</td>
<td>Distribution</td>
<td>Parameters</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Generation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Output

<table>
<thead>
<tr>
<th>Type</th>
<th>Reference Definition</th>
<th>Units</th>
<th>Calibration</th>
</tr>
</thead>
</table>

Administrative Information

<table>
<thead>
<tr>
<th>Status</th>
<th>Request</th>
<th>Rev</th>
<th>Rev.Date</th>
</tr>
</thead>
</table>

Comments and Remarks

4. UDP Round-trip Latency and Loss Registry Entries

This section specifies an initial registry entry for the UDP Round-trip Latency, and another entry for UDP Round-trip Loss Ratio.

Note: Each Registry entry only produces a "raw" output or a statistical summary. To describe both "raw" and one or more statistics efficiently, the Identifier, Name, and Output Categories can be split and a single section can specify two or more closely-related metrics. This section specifies two Registry entries with many common columns. See Section 7 for an example specifying multiple Registry entries with many common columns.

All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes two closely-related registry entries. As a result, IANA is also asked to assign corresponding URNs and URLs to each Named Metric.
4.1. Summary

This category includes multiple indexes to the registry entry: the element ID and metric name.

4.1.1. ID (Identifier)

<insert a numeric identifier, an integer, TBD>

IANA is asked to assign different numeric identifiers to each of the two Named Metrics.

4.1.2. Name

<insert name according to metric naming convention>

RTDelay_Active_IP-UDP-Periodic_RFCXXXXsecY_Seconds_95Percentile

RTLoss_Active_IP-UDP-Periodic_RFCXXXXsecY_Percent_LossRatio

4.1.3. URIs

URN: Prefix urn:ietf:metrics:perf:<name>

URL: http://<TBD by IANA>/<name>

4.1.4. Description

RTDelay: This metric assesses the delay of a stream of packets exchanged between two hosts (which are the two measurement points), and the Output is the Round-trip delay for all successfully exchanged packets expressed as the 95th percentile of their conditional delay distribution.

RTLoss: This metric assesses the loss ratio of a stream of packets exchanged between two hosts (which are the two measurement points), and the Output is the Round-trip loss ratio for all successfully exchanged packets expressed as a percentage.

4.1.5. Change Controller

IETF

4.1.6. Version (of Registry Format)

1.0
4.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.

4.2.1. Reference Definition

<Full bibliographic reference to an immutable doc.>


[RFC2681]

<specific section reference and additional clarifications, if needed>

Section 2.4 of [RFC2681] provides the reference definition of the singleton (single value) Round-trip delay metric. Section 3.4 of [RFC2681] provides the reference definition expanded to cover a multi-singleton sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

Note that although the [RFC2681] definition of "Round-trip-Delay between Src and Dst" is directionally ambiguous in the text, this metric tightens the definition further to recognize that the host in the "Src" role will send the first packet to "Dst", and ultimately receive the corresponding return packet from "Dst" (when neither are lost).

Finally, note that the variable "dT" is used in [RFC2681] to refer to the value of Round-trip delay in metric definitions and methods. The variable "dT" has been re-used in other IPPM literature to refer to different quantities, and cannot be used as a global variable name.


[RFC6673]

Both delay and loss metrics employ a maximum waiting time for received packets, so the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 6.1 of [RFC6673].
4.2.2. Fixed Parameters

<list and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed>

Type-P as defined in Section 13 of [RFC2330]:

- IPv4 header values:
  * DSCP: set to 0
  * TTL: set to 255
  * Protocol: Set to 17 (UDP)

- IPv6 header values:
  * DSCP: set to 0
  * Hop Count: set to 255
  * Protocol: Set to 17 (UDP)

- UDP header values:
  * Checksum: the checksum MUST be calculated and included in the header

- UDP Payload
  * total of 100 bytes

Other measurement parameters:

- Tmax: a loss threshold waiting time
  * 3.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

4.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.
4.3.1. Reference Method

The methodology for this metric is defined as Type-P-Round-trip-Delay-Poisson-Stream in section 2.6 of RFC 2681 [RFC2681] and section 3.6 of RFC 2681 [RFC2681] using the Type-P and Tmax defined under Fixed Parameters. However, the Periodic stream will be generated according to [RFC3432].

The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay, and counted for the RTLoss metric.

The calculations on the delay (RTT) SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the RTT value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving packet. Sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs.

If a standard measurement protocol is employed, then the measurement process will determine the sequence numbers or timestamps applied to test packets after the Fixed and Runtime parameters are passed to that process. The chosen measurement protocol will dictate the format of sequence numbers and time-stamps, if they are conveyed in the packet payload.

Refer to Section 4.4 of [RFC6673] for expanded discussion of the instruction to "send a Type-P packet back to the Src as quickly as possible" in Section 2.6 of RFC 2681 [RFC2681]. Section 8 of [RFC6673] presents additional requirements which MUST be included in the method of measurement for this metric.
4.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.

Section 3 of [RFC3432] prescribes the method for generating Periodic streams using associated parameters.

incT the nominal duration of inter-packet interval, first bit to first bit, with value 0.0200, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms).

dT the duration of the interval for allowed sample start times, with value 1.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms).

T0 the actual start time of the periodic stream, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]).

NOTE: an initiation process with a number of control exchanges resulting in unpredictable start times (within a time interval) may be sufficient to avoid synchronization of periodic streams, and therefore a valid replacement for selecting a start time at random from a fixed interval.

The T0 parameter will be reported as a measured parameter. Parameters incT and dT are Fixed Parameters.

4.3.3. Traffic Filtering (observation) Details

The measured results based on a filtered version of the packets observed, and this section provides the filter details (when present).

<section reference>.

NA
4.3.4. Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

NA

4.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>

Src  the IP address of the host in the Src Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])

Dst  the IP address of the host in the Dst Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0 a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Tf is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.

Tf a time, the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a end time date is ignored and Tf is interpreted as the Duration of the measurement interval.

4.3.6. Roles

<lists the names of the different roles from the measurement method>

Src  launches each packet and waits for return transmissions from Dst.

Dst  waits for each packet from Src and sends a return packet to Src.
4.4. Output

This category specifies all details of the Output of measurements using the metric.

4.4.1. Type

<insert name of the output type, raw or a selected summary statistic>

Percentile -- for the conditional distribution of all packets with a valid value of Round-trip delay (undefined delays are excluded), a single value corresponding to the 95th percentile, as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The percentile = 95, meaning that the reported delay, "95Percentile", is the smallest value of Round-trip delay for which the Empirical Distribution Function (EDF), \( F(95\text{Percentile}) \geq 95\% \) of the singleton Round-trip delay values in the conditional distribution. See section 11.3 of [RFC2330] for the definition of the percentile statistic using the EDF.

LossRatio -- the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 6.1 of [RFC6673].

4.4.2. Reference Definition

<describe the reference data format for each type of result>

For all outputs ---

T0 the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

Tf the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

TotalPkts the count of packets sent by the Src to Dst during the measurement interval.

For
RTDelay_Active_IP-UDP-Periodic_RFCXXXXsecY_Seconds_95Percentile:

95Percentile  The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as.

For

RTLoss_Active_IP-UDP-Periodic_RFCXXXXsecY_Percent_LossRatio:

Percentile  The numeric value of the result is expressed in units of lost packets to total packets times 100%, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.0000000001.

4.4.3. Metric Units

<insert units for the measured results, and the reference specification>.

The 95th Percentile of Round-trip Delay is expressed in seconds.  
The Round-trip Loss Ratio is expressed as a percentage of lost packets to total packets sent.

4.4.4. Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the systematic and random errors of a time measurement.  In-situ calibration could be enabled with an internal loopback at the Source host that includes as much of the measurement system as possible, performs address manipulation as needed, and provides some form of isolation (e.g., deterministic delay) to avoid send-receive interface contention.  Some portion of the random and systematic error can be characterized this way.

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result.

Both internal loopback calibration and clock synchronization can be used to estimate the *available accuracy* of the Output Metric Units.  For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.
4.5. Administrative items

4.5.1. Status
   <current or deprecated>

4.5.2. Requestor (keep?)
   name or RFC, etc.

4.5.3. Revision
   1.0

4.5.4. Revision Date
   YYYY-MM-DD

4.6. Comments and Remarks
   Additional (Informational) details for this entry

5. Packet Delay Variation Registry Entry

   This section gives an initial registry entry for a Packet Delay
   Variation metric.

   Note: If each Registry entry should only produce a "raw" output or a
   statistical summary, then the "Output" Category can be split and this
   section can become two closely-related metrics.

5.1. Summary

   This category includes multiple indexes to the registry entries, the
   element ID and metric name.

   <skipping some Summary columns for now>

5.1.1. ID (Identifier)
   <insert numeric identifier, an integer>

5.1.2. Name
   <insert name according to metric naming convention>

OWPDV_Active_IP-UDP-Periodic/rfcXXXXsecYSeconds_95Percentile
5.1.3. URIs

URI: Prefix urn:ietf:metrics:perf:<name>

URL: http://<TBD by IANA>/<name>

5.1.4. Description

An assessment of packet delay variation with respect to the minimum delay observed on the periodic stream, and the Output is expressed as the 95th percentile of the packet delay variation distribution.

5.1.5. Change Controller

<org or person >

IETF

5.1.6. Version (of Registry Format)

1.0

5.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.

5.2.1. Reference Definition

<Full bibliographic reference to an immutable doc.>


<specific section reference and additional clarifications, if needed>
See sections 2.4 and 3.4 of [RFC3393]. Singleton delay differences measured are referred to by the variable name "ddT" (applicable to all forms of delay variation). However, this metric entry specifies the PDV form defined in section 4.2 of [RFC5481], where the singleton PDV for packet \( i \) is referred to by the variable name "PDV(i)".

5.2.2. Fixed Parameters

<list and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed>

- IPv4 header values:
  - DSCP: set to 0
  - TTL: set to 255
  - Protocol: Set to 17 (UDP)

- IPv6 header values:
  - DSCP: set to 0
  - Hop Count: set to 255
  - Protocol: Set to 17 (UDP)

- UDP header values:
  - Checksum: the checksum MUST be calculated and included in the header

- UDP Payload
  - total of 200 bytes

Other measurement parameters:

- \( T_{max} \): a loss threshold waiting time with value 3.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

- \( F \): a selection function unambiguously defining the packets from the stream selected for the metric. See section 4.2 of [RFC5481] for the PDV form.
See the Packet Stream generation category for two additional Fixed Parameters.

5.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.

5.3.1. Reference Method

<for metric, insert relevant section references and supplemental info>

See section 2.6 and 3.6 of [RFC3393] for general singleton element calculations. This metric entry requires implementation of the PDV form defined in section 4.2 of [RFC5481]. Also see measurement considerations in section 8 of [RFC5481].

The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay.

The calculations on the one-way delay SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the one-way delay value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving packet. Sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs.

If a standard measurement protocol is employed, then the measurement process will determine the sequence numbers or timestamps applied to test packets after the Fixed and Runtime parameters are passed to that process. The chosen measurement protocol will dictate the format of sequence numbers and time-stamps, if they are conveyed in the packet payload.
5.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.

Section 3 of [RFC3432] prescribes the method for generating Periodic streams using associated parameters.

\( \text{incT} \) the nominal duration of inter-packet interval, first bit to first bit, with value 0.0200, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms).

\( \text{dT} \) the duration of the interval for allowed sample start times, with value 1.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms).

\( \text{T0} \) the actual start time of the periodic stream, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]).

NOTE: an initiation process with a number of control exchanges resulting in unpredictable start times (within a time interval) may be sufficient to avoid synchronization of periodic streams, and therefore a valid replacement for selecting a start time at random from a fixed interval.

The \( \text{T0} \) parameter will be reported as a measured parameter. Parameters \( \text{incT} \) and \( \text{dT} \) are Fixed Parameters.

5.3.3. Traffic Filtering (observation) Details

<insert the measured results based on a filtered version of the packets observed, and this section provides the filter details (when present), and section reference>.

NA
5.3.4.  Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

NA

5.3.5.  Run-time Parameters and Data Format

<list of run-time parameters, and their data formats>

Src  the IP address of the host in the Src Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])

Dst  the IP address of the host in the Dst Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0 a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Tf is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.

Tf a time, the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a end time date is ignored and Tf is interpreted as the Duration of the measurement interval.

5.3.6.  Roles

<lists the names of the different roles from the measurement method>

Src  launches each packet to Dst.

Dst  waits for each packet from Src.

5.4.  Output

This category specifies all details of the Output of measurements using the metric.
5.4.1. Type

<insert name of the output type, raw or a selected summary statistic>

Percentile -- for the conditional distribution of all packets with a valid value of one-way delay (undefined delays are excluded), a single value corresponding to the 95th percentile, as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The percentile = 95, meaning that the reported delay, "95Percentile", is the smallest value of one-way PDV for which the Empirical Distribution Function (EDF), F(95Percentile) >= 95% of the singleton one-way PDV values in the conditional distribution. See section 11.3 of [RFC2330] for the definition of the percentile statistic using the EDF.

5.4.2. Reference Definition

<the output type and data format for each type of result>

T0 the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

Tf the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

95Percentile The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

5.4.3. Metric Units

<insert units for the measured results, and the reference specification>.

The 95th Percentile of one-way PDV is expressed in seconds.
5.4.4. Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the systematic and random errors of a time measurement. In-situ calibration could be enabled with an internal loopback that includes as much of the measurement system as possible, performs address manipulation as needed, and provides some form of isolation (e.g., deterministic delay) to avoid send-receive interface contention. Some portion of the random and systematic error can be characterized this way.

For one-way delay measurements, the error calibration must include an assessment of the internal clock synchronization with its external reference (this internal clock is supplying timestamps for measurement). In practice, the time offsets of clocks at both the source and destination are needed to estimate the systematic error due to imperfect clock synchronization (the time offsets are smoothed, thus the random variation is not usually represented in the results).

time_offset. The time value of the result is expressed in units of seconds, as a signed value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905].

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result. In any measurement, the measurement function SHOULD report its current estimate of time offset as an indicator of the degree of synchronization.

Both internal loopback calibration and clock synchronization can be used to estimate the "available accuracy" of the Output Metric Units. For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.

5.5. Administrative items

5.5.1. Status

<current or deprecated>
5.5.2. Requestor (keep?)
   <name of individual or RFC, etc.>

5.5.3. Revision
   1.0

5.5.4. Revision Date
   YYYY-MM-DD

5.6. Comments and Remarks
   <Additional (Informational) details for this entry>

Lost packets represent a challenge for delay variation metrics. See section 4.1 of [RFC3393] and the delay variation applicability statement[RFC5481] for extensive analysis and comparison of PDV and an alternate metric, IPDV.

6. DNS Response Latency and Loss Registry Entries

    @@@ comment from Brian: there is an interesting method for DNS measurement by encoding information in the query itself. It is a question of what exactly we are trying to measure: specific RR, or the infrastructure itself. (at this time we measure a specific RR).

    This section gives initial registry entries for DNS Response Latency and Loss from a network user’s perspective, for a specific named resource. The metric can be measured repeatedly using different names. RFC 2681 [RFC2681] defines a Round-trip delay metric. We build on that metric by specifying several of the input parameters to precisely define two metrics for measuring DNS latency and loss.

    Note to IANA: Each Registry "Name" below specifies a single registry entry, whose output format varies in accordance with the name.

    All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes two closely-related registry entries. As a result, IANA is also asked to assign corresponding URNs and URLs to each Named Metric.

6.1. Summary

    This category includes multiple indexes to the registry entries, the element ID and metric name.
6.1.1. ID (Identifier)

<insert numeric identifier, an integer>

IANA is asked to assign different numeric identifiers to each of the two Named Metrics.

6.1.2. Name

<insert name according to metric naming convention>

RTDNS_Active_IP-UDP-Poisson_RFCXXXXsecY_Seconds_Raw

RLDNS_Active_IP-UDP-Poisson_RFCXXXXsecY_Logical_Raw

6.1.3. URI

URI: Prefix urn:ietf:metrics:perf:<name>

URL: http://<TBD by IANA>/<name>

6.1.4. Description

This is a metric for DNS Response performance from a network user’s perspective, for a specific named resource. The metric can be measured repeatedly using different resource names.

RTDNS: This metric assesses the response time, the interval from the query transmission to the response.

RLDNS: This metric indicates that the response was deemed lost. In other words, the response time exceeded the maximum waiting time.

6.1.5. Change Controller

IETF

6.1.6. Version (of Registry Format)

1.0

6.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.
6.2.1. Reference Definition


[RFC1035]


[RFC2681]

Section 2.4 of [RFC2681] provides the reference definition of the singleton (single value) Round-trip delay metric. Section 3.4 of [RFC2681] provides the reference definition expanded to cover a multi-singleton sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

For DNS Response Latency, the entities in [RFC1035] must be mapped to [RFC2681]. The Local Host with its User Program and Resolver take the role of "Src", and the Foreign Name Server takes the role of "Dst".


[RFC6673]

Both response time and loss metrics employ a maximum waiting time for received responses, so the count of lost packets to total packets sent is the basis for the loss determination as per Section 4.3 of [RFC6673].

6.2.2. Fixed Parameters

Morton, et al. Expires January 1, 2019
Type-P as defined in Section 13 of [RFC2330]:

- IPv4 header values:
  - DSCP: set to 0
  - TTL set to 255
  - Protocol: Set to 17 (UDP)

- IPv6 header values:
  - DSCP: set to 0
  - Hop Count: set to 255
  - Protocol: Set to 17 (UDP)

- UDP header values:
  - Source port: 53
  - Destination port: 53
  - Checksum: the checksum must be calculated and included in the header

- Payload: The payload contains a DNS message as defined in RFC 1035 [RFC1035] with the following values:
  - The DNS header section contains:
    - Identification (see the Run-time column)
    - QR: set to 0 (Query)
    - OPCODE: set to 0 (standard query)
    - AA: not set
    - TC: not set
    - RD: set to one (recursion desired)
    - RA: not set
    - RCODE: not set
+ QDCOUNT: set to one (only one entry)
+ ANCOUNT: not set
+ NSCOUNT: not set
+ ARCOUNT: not set

* The Question section contains:
  + QNAME: the Fully Qualified Domain Name (FQDN) provided as input for the test, see the Run-time column
  + QTYPE: the query type provided as input for the test, see the Run-time column
  + QCLASS: set to 1 for IN

* The other sections do not contain any Resource Records.

Other measurement parameters:

- Tmax: a loss threshold waiting time (and to help disambiguate queries)

  * 5.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

Observation: reply packets will contain a DNS response and may contain RRs.

6.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.

6.3.1. Reference Method

<for metric, insert relevant section references and supplemental info>

The methodology for this metric is defined as Type-P-Round-trip-Delay-Poisson-Stream in section 2.6 of RFC 2681 [RFC2681] and section
3.6 of RFC 2681 [RFC2681] using the Type-P and Timeout defined under Fixed Parameters.

The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay, and counted for the RLDNS metric.

The calculations on the delay (RTT) SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the RTT value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving reply. Therefore, sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs. Sequence number is part of the payload described under Fixed Parameters.

DNS Messages bearing Queries provide for random ID Numbers in the Identification header field, so more than one query may be launched while a previous request is outstanding when the ID Number is used.

IF a DNS response does not arrive within Tmax, the response time is undefined, and RTDNS = 1. The Message ID SHALL be used to disambiguate the successive queries.

This would require support of ID generation and population in the Message. An alternative would be to use a random Source port on the Query Message, but we would choose ONE before proceeding.

Refer to Section 4.4 of [RFC6673] for expanded discussion of the instruction to "send a Type-P packet back to the Src as quickly as possible" in Section 2.6 of RFC 2681 [RFC2681]. Section 8 of [RFC6673] presents additional requirements which shall be included in the method of measurement for this metric.

In addition to operations described in [RFC2681], the Src MUST parse the DNS headers of the reply and prepare the information for subsequent reporting as a measured result, along with the Round-Trip Delay.
6.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.

Section 11.1.3 of RFC 2681 [RFC2330] provides three methods to generate Poisson sampling intervals. The reciprocal of lambda is the average packet rate, thus the Run-time Parameter is Reciprocal_lambda = 1/lambda, in seconds.

Method 3 is used, where given a start time (Run-time Parameter), the subsequent send times are all computed prior to measurement by computing the pseudo-random distribution of inter-packet send times, (truncating the distribution as specified in the Run-time Parameters), and the Src sends each packet at the computed times.

Note that Trunc is the upper limit on inter-packet times in the Poisson distribution. A random value greater than Trunc is set equal to Trunc instead.

6.3.3. Traffic Filtering (observation) Details

The measured results based on a filtered version of the packets observed, and this section provides the filter details (when present).

<section reference>.

NA

6.3.4. Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

NA

6.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>
Src  the IP address of the host in the Src Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])

Dst  the IP address of the host in the Dst Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0 a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Tf is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.

Tf a time, the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a end time date is ignored and Tf is interpreted as the Duration of the measurement interval.

Reciprocal_lambda  average packet interval for Poisson Streams expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) with resolution of 0.0001 seconds (0.1 ms), and with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

Trunc  Upper limit on Poisson distribution expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) with resolution of 0.0001 seconds (0.1 ms), and with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905] (values above this limit will be clipped and set to the limit value). (if fixed, Trunc = 30.0000 seconds.)

ID The 16-bit identifier assigned by the program that generates the query, and which must vary in successive queries, see Section 4.1.1 of [RFC1035]. This identifier is copied into the corresponding reply and can be used by the requester (Src) to match-up replies to outstanding queries.

QNAME  The domain name of the Query, formatted as specified in section 4 of [RFC6991].

QTYPE  The Query Type, which will correspond to the IP address family of the query (decimal 1 for IPv4 or 28 for IPv6, formatted as a uint16, as per section 9.2 of [RFC6020].
6.3.6. Roles

Src launches each packet and waits for return transmissions from Dst.

Dst waits for each packet from Src and sends a return packet to Src.

6.4. Output

This category specifies all details of the Output of measurements using the metric.

6.4.1. Type

Raw -- for each DNS Query packet sent, sets of values as defined in the next column, including the status of the response, only assigning delay values to successful query-response pairs.

6.4.2. Reference Definition

For all outputs:

\[ T \] the time the DNS Query was sent during the measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

\[ dT \] The time value of the round-trip delay to receive the DNS response, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]. This value is undefined when the response packet is not received at Src within waiting time Tmax seconds.

\[ Rcode \] The value of the Rcode field in the DNS response header, expressed as a uint64 as specified in section 9.2 of [RFC6020]. Non-zero values convey errors in the response, and such replies must be analyzed separately from successful requests.
6.4.3. Metric Units

<insert units for the measured results, and the reference specification>.

RTDNS: Round-trip Delay, dT, is expressed in seconds.

RTLDNS: the Logical value, where 1 = Lost and 0 = Received.

6.4.4. Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the systematic and random errors of a time measurement. In-situ calibration could be enabled with an internal loopback at the Source host that includes as much of the measurement system as possible, performs address and payload manipulation as needed, and provides some form of isolation (e.g., deterministic delay) to avoid send-receive interface contention. Some portion of the random and systematic error can be characterized this way.

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result.

Both internal loopback calibration and clock synchronization can be used to estimate the *available accuracy* of the Output Metric Units. For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.

6.5. Administrative items

6.5.1. Status

<current or deprecated>

6.5.2. Requestor

name or RFC, etc.

6.5.3. Revision

1.0
6.5.4. Revision Date

YYYY-MM-DD

6.6. Comments and Remarks

Additional (Informational) details for this entry

7. UDP Poisson One-way Delay and Loss Registry Entries

This section specifies five initial registry entries for the UDP Poisson One-way Delay, and one for UDP Poisson One-way Loss.

IANA Note: Registry "Name" below specifies a single registry entry, whose output format varies according to the <statistic> element of the name that specifies one form of statistical summary. There is an additional metric name for the Loss metric.

All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes six closely-related registry entries. As a result, IANA is also asked to assign corresponding URNs and URLs to each Named Metric.

7.1. Summary

This category includes multiple indexes to the registry entries, the element ID and metric name.

7.1.1. ID (Identifier)

<insert numeric identifier, an integer, one corresponding to each name below>

IANA is asked to assign different numeric identifiers to each of the six Metrics.

7.1.2. Name

<insert name according to metric naming convention>

OWDelay_Active_IP-UDP-Poisson-Payload250B_RFCXXXsecY_Seconds_<statistic>

where <statistic> is one of:

- 95Percentile
- Mean
7.1.3. URI and URL

URI: Prefix urn:ietf:metrics:perf:<name>

URL: http:\www.iana.org\ ... <name>

7.1.4. Description

OWDelay: This metric assesses the delay of a stream of packets exchanged between two hosts (or measurement points), and reports the one-way delay for all successfully exchanged packets based on their conditional delay distribution.

where <statistic> is one of:

- 95Percentile
- Mean
- Min
- Max
- StdDev

OWLoss: This metric assesses the loss ratio of a stream of packets exchanged between two hosts (which are the two measurement points), and the output is the one-way loss ratio for all successfully received packets expressed as a percentage.

7.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.
7.2.1. Reference Definition

For Delay:


[MRF7679]


[RFC6049]

Section 3.4 of [RFC7679] provides the reference definition of the singleton (single value) One-way delay metric. Section 4.4 of [RFC7679] provides the reference definition expanded to cover a multi-value sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

Only successful packet transfers with finite delay are included in the sample, as prescribed in section 4.1.2 of [RFC6049].

For loss:


Section 2.4 of [RFC7680] provides the reference definition of the singleton (single value) one-way loss metric. Section 3.4 of [RFC7680] provides the reference definition expanded to cover a multi-singleton sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

7.2.2. Fixed Parameters

List and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed.
Type-P:

- IPv4 header values:
  - DSCP: set to 0
  - TTL: set to 255
  - Protocol: Set to 17 (UDP)

- IPv6 header values:
  - DSCP: set to 0
  - Hop Count: set to 255
  - Protocol: Set to 17 (UDP)

- UDP header values:
  - Checksum: the checksum MUST be calculated and included in the header

- UDP Payload: TWAMP Test Packet Formats, Section 4.1.2 of [RFC5357]
  - Security features in use influence the number of Padding octets.
  - 250 octets total, including the TWAMP format

Other measurement parameters:

Tmax: a loss threshold waiting time with value 3.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

See the Packet Stream generation category for two additional Fixed Parameters.

7.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.
7.3.1. Reference Method

The methodology for this metric is defined as Type-P-One-way-Delay-Poisson-Stream in section 3.6 of [RFC7679] and section 4.6 of [RFC7679] using the Type-P and Tmax defined under Fixed Parameters.

The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay, and counted for the OWLoss metric.

The calculations on the one-way delay SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the one-way delay value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving packet. Sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs.

Since a standard measurement protocol is employed [RFC5357], then the measurement process will determine the sequence numbers or timestamps applied to test packets after the Fixed and Runtime parameters are passed to that process. The measurement protocol dictates the format of sequence numbers and time-stamps conveyed in the TWAMP-Test packet payload.

7.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.

<list of generation parameters and section/spec references if needed>
Section 11.1.3 of RFC 2681 [RFC2330] provides three methods to generate Poisson sampling intervals. The reciprocal of lambda is the average packet spacing, thus the Run-time Parameter is Reciprocal_lambda = 1/lambda, in seconds.

Method 3 SHALL be used, where given a start time (Run-time Parameter), the subsequent send times are all computed prior to measurement by computing the pseudo-random distribution of inter-packet send times, (truncating the distribution as specified in the Parameter Trunc), and the Src sends each packet at the computed times.

Note that Trunc is the upper limit on inter-packet times in the Poisson distribution. A random value greater than Trunc is set equal to Trunc instead.

Reciprocal_lambda  average packet interval for Poisson Streams expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) with resolution of 0.0001 seconds (0.1 ms), and with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905]. Reciprocal_lambda = 1 packet per second.

Trunc  Upper limit on Poisson distribution expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) with resolution of 0.0001 seconds (0.1 ms), and with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905] (values above this limit will be clipped and set to the limit value). Trunc = 30.0000 seconds.

7.3.3. Traffic Filtering (observation) Details

NA

7.3.4. Sampling Distribution

NA

7.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>
7.3.6. Roles

Src  launches each packet and waits for return transmissions from Dst. This is the TWAMP Session-Sender.

Dst  waits for each packet from Src and sends a return packet to Src. This is the TWAMP Session-Reflector.

7.4. Output

This category specifies all details of the Output of measurements using the metric.

7.4.1. Type

See subsection titles below for Types.

7.4.2. Reference Definition

For all output types ---
To the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

To the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

For LossRatio -- the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 4.1 of [RFC7680].

For each <statistic>, one of the following sub-sections apply:

7.4.2.1. Percentile95

The 95th percentile SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3 of [RFC3393] for details on the percentile statistic (where Round-trip delay should be substituted for "ipdv").

The percentile = 95, meaning that the reported delay, "95Percentile", is the smallest value of one-way delay for which the Empirical Distribution Function (EDF), F(95Percentile) >= 95% of the singleton one-way delay values in the conditional distribution. See section 11.3 of [RFC2330] for the definition of the percentile statistic using the EDF.

95Percentile The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

7.4.2.2. Mean

The mean SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:
See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.2.2 of [RFC6049] for details on calculating this statistic, and 4.2.3 of [RFC6049].

Mean The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

7.4.2.3. Min

The minimum SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for details on calculating this statistic, and 4.3.3 of [RFC6049].

Min The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

7.4.2.4. Max

The maximum SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for calculating this statistic, and 4.3.3 of [RFC6049]. The formula is as follows:
Max = \( (\text{FiniteDelay}[j]) \)

such that for some index, \( j \), where \( 1 \leq j \leq N \)
\( \text{FiniteDelay}[j] \geq \text{FiniteDelay}[n] \) for all \( n \)

Max  The time value of the result is expressed in units of seconds,
as a positive value of type decimal64 with fraction digits = 9
(see section 9.3 of [RFC6020]) with resolution of 0.000000001
seconds (1.0 ns), and with lossless conversion to/from the 64-bit
NTP timestamp as per section 6 of RFC [RFC5905]

7.4.2.5.  Std_Dev

The Std_Dev SHALL be calculated using the conditional distribution of
all packets with a finite value of One-way delay (undefined delays
are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional
distribution to exclude undefined values of delay, and Section 5 of
[RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for
calculating this statistic, and 4.3.3 of [RFC6049].  The formula is
the classic calculation for standard deviation of a population.

Std_Dev  The time value of the result is expressed in units of
seconds, as a positive value of type decimal64 with fraction
digits = 9 (see section 9.3 of [RFC6020]) with resolution of
0.000000001 seconds (1.0 ns), and with lossless conversion to/from
the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

7.4.3.  Metric Units

<insert units for the measured results, and the reference
specification>.

The <statistic> of One-way Delay is expressed in seconds.

The One-way Loss Ratio is expressed as a percentage of lost packets
to total packets sent.

7.4.4.  Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the
systematic and random errors of a time measurement.  In-situ
calibration could be enabled with an internal loopback that includes
as much of the measurement system as possible, performs address
manipulation as needed, and provides some form of isolation (e.g.,
deterministic delay) to avoid send-receive interface contention. Some portion of the random and systematic error can be characterized this way.

For one-way delay measurements, the error calibration must include an assessment of the internal clock synchronization with its external reference (this internal clock is supplying timestamps for measurement). In practice, the time offsets of clocks at both the source and destination are needed to estimate the systematic error due to imperfect clock synchronization (the time offsets are smoothed, thus the random variation is not usually represented in the results).

time_offset  The time value of the result is expressed in units of seconds, as a signed value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result. In any measurement, the measurement function SHOULD report its current estimate of time offset as an indicator of the degree of synchronization.

Both internal loopback calibration and clock synchronization can be used to estimate the *available accuracy* of the Output Metric Units. For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.

7.5. Administrative items

7.5.1. Status

   <current or deprecated>

7.5.2. Requestor (keep?)

   name or RFC, etc.

7.5.3. Revision

   1.0
7.5.4. Revision Date

YYYY-MM-DD

7.6. Comments and Remarks

Additional (Informational) details for this entry

8. UDP Periodic One-way Delay and Loss Registry Entries

This section specifies five initial registry entries for the UDP Periodic One-way Delay, and one for UDP Periodic One-way Loss.

IANA Note: Registry "Name" below specifies a single registry entry, whose output format varies according to the <statistic> element of the name that specifies one form of statistical summary. There is an additional metric name for the Loss metric.

All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes six closely-related registry entries. As a result, IANA is also asked to assign corresponding URNs and URLs to each Named Metric.

8.1. Summary

This category includes multiple indexes to the registry entries, the element ID and metric name.

8.1.1. ID (Identifier)

<insert numeric identifier, an integer, one corresponding to each name below>

IANA is asked to assign a different numeric identifiers to each of the six Metrics.

8.1.2. Name

<insert name according to metric naming convention>

OWDelay_Active_IP-UDP-Periodic-Payload142B_RFCXXXXsecY_Seconds_<statistic>

where <statistic> is one of:

- 95Percentile
- Mean
8.1.3. URIs

URI: Prefix urn:ietf:metrics:perf:<name>

URL: http://www.iana.org/...<name>

8.1.4. Description

OWDelay: This metric assesses the delay of a stream of packets exchanged between two hosts (or measurement points), and reports the one-way delay for all successfully exchanged packets based on their conditional delay distribution.

where <statistic> is one of:

- 95Percentile
- Mean
- Min
- Max
- StdDev

OWLoss: This metric assesses the loss ratio of a stream of packets exchanged between two hosts (which are the two measurement points), and the output is the one-way loss ratio for all successfully received packets expressed as a percentage.

8.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.
8.2.1. Reference Definition

For Delay:


[RFC7679]


[RFC6049]

Section 3.4 of [RFC7679] provides the reference definition of the singleton (single value) one-way delay metric. Section 4.4 of [RFC7679] provides the reference definition expanded to cover a multi-value sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

Only successful packet transfers with finite delay are included in the sample, as prescribed in section 4.1.2 of [RFC6049].

For loss:


Section 2.4 of [RFC7680] provides the reference definition of the singleton (single value) one-way loss metric. Section 3.4 of [RFC7680] provides the reference definition expanded to cover a multi-singleton sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

8.2.2. Fixed Parameters

(list and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed)
Type-P:

- IPv4 header values:
  * DSCP: set to 0
  * TTL: set to 255
  * Protocol: Set to 17 (UDP)

- IPv6 header values:
  * DSCP: set to 0
  * Hop Count: set to 255
  * Protocol: Set to 17 (UDP)

- UDP header values:
  * Checksum: the checksum MUST be calculated and included in the header

- UDP Payload: TWAMP Test Packet Formats, Section 4.1.2 of [RFC5357]
  * Security features in use influence the number of Padding octets.
  * 142 octets total, including the TWAMP format

Other measurement parameters:

Tmax: a loss threshold waiting time with value 3.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

See the Packet Stream generation category for two additional Fixed Parameters.

8.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.
8.3.1. Reference Method

The methodology for this metric is defined as Type-P-One-way-Delay-Poisson-Stream in section 3.6 of [RFC7679] and section 4.6 of [RFC7679] using the Type-P and Tmax defined under Fixed Parameters. However, a Periodic stream is used, as defined in [RFC3432].

The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay, and counted for the OWLoss metric.

The calculations on the one-way delay SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the one-way delay value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving packet. Sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs.

Since a standard measurement protocol is employed [RFC5357], then the measurement process will determine the sequence numbers or timestamps applied to test packets after the Fixed and Runtime parameters are passed to that process. The measurement protocol dictates the format of sequence numbers and time-stamps conveyed in the TWAMP-Test packet payload.

8.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.
Section 3 of [RFC3432] prescribes the method for generating Periodic streams using associated parameters.

incT  the nominal duration of inter-packet interval, first bit to first bit

dT   the duration of the interval for allowed sample start times

T0  the actual start time of the periodic stream

NOTE: an initiation process with a number of control exchanges resulting in unpredictable start times (within a time interval) may be sufficient to avoid synchronization of periodic streams, and therefore a valid replacement for selecting a start time at random from a fixed interval.

These stream parameters will be specified as Run-time parameters.

8.3.3. Traffic Filtering (observation) Details

NA

8.3.4. Sampling Distribution

NA

8.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>

Src  the IP address of the host in the Src Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])

Dst  the IP address of the host in the Dst Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0  a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Tf is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.
Tf a time, the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a end time date is ignored and Tf is interpreted as the Duration of the measurement interval.

@@@@ should Periodic run-time params be fixed instead? Probably yes if modeling a specific version of tests. Note in the NAME, i.e. Poisson3.3

8.3.6. Roles

<lists the names of the different roles from the measurement method>

Src launches each packet and waits for return transmissions from Dst. This is the TWAMP Session-Sender.

Dst waits for each packet from Src and sends a return packet to Src. This is the TWAMP Session-Reflector.

8.4. Output

This category specifies all details of the Output of measurements using the metric.

8.4.1. Type

<insert name of the output type, raw or a selected summary statistic>

See subsection titles in Reference Definition for Latency Types.

8.4.2. Reference Definition

<describe the data format for each type of result>

For all output types ---

T0 the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

Tf the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].
For LossRatio -- the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 4.1 of [RFC7680].

For each <statistic>, one of the following sub-sections apply:

8.4.2.1. Percentile95

The 95th percentile SHALL be calculated using the conditional distribution of all packets with a finite value of one-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3 of [RFC3393] for details on the percentile statistic (where Round-trip delay should be substituted for "ipdv").

The percentile = 95, meaning that the reported delay, "95Percentile", is the smallest value of one-way delay for which the Empirical Distribution Function (EDF), F(95Percentile) >= 95% of the singleton one-way delay values in the conditional distribution. See section 11.3 of [RFC2330] for the definition of the percentile statistic using the EDF.

95Percentile The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

8.4.2.2. Mean

The mean SHALL be calculated using the conditional distribution of all packets with a finite value of one-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.2.2 of [RFC6049] for details on calculating this statistic, and 4.2.3 of [RFC6049].

Mean The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001
seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

8.4.2.3. **Min**

The minimum SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for details on calculating this statistic, and 4.3.3 of [RFC6049].

Min The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

8.4.2.4. **Max**

The maximum SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for calculating this statistic, and 4.3.3 of [RFC6049]. The formula is as follows:

\[ \text{Max} = \text{FiniteDelay}[j] \]

such that for some index, \( j \), where \( 1 \leq j \leq N \)

\[ \text{FiniteDelay}[j] \geq \text{FiniteDelay}[n] \text{ for all } n \]

Max The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]
8.4.2.5. Std_Dev

The Std_Dev SHALL be calculated using the conditional distribution of all packets with a finite value of One-way delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for calculating this statistic, and 4.3.3 of [RFC6049]. The formula is the classic calculation for standard deviation of a population.

Std_Dev The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

8.4.3. Metric Units

<insert units for the measured results, and the reference specification>.

The <statistic> of One-way Delay is expressed in seconds, where <statistic> is one of:

- 95Percentile
- Mean
- Min
- Max
- StdDev

The One-way Loss Ratio is expressed as a percentage of lost packets to total packets sent.

8.4.4. Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the systematic and random errors of a time measurement. In-situ calibration could be enabled with an internal loopback that includes as much of the measurement system as possible, performs address manipulation as needed, and provides some form of isolation (e.g.,
deterministic delay) to avoid send-receive interface contention. Some portion of the random and systematic error can be characterized this way.

For one-way delay measurements, the error calibration must include an assessment of the internal clock synchronization with its external reference (this internal clock is supplying timestamps for measurement). In practice, the time offsets of clocks at both the source and destination are needed to estimate the systematic error due to imperfect clock synchronization (the time offsets are smoothed, thus the random variation is not usually represented in the results).

\texttt{time\_offset}  The time value of the result is expressed in units of seconds, as a signed value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result. In any measurement, the measurement function SHOULD report its current estimate of time offset as an indicator of the degree of synchronization.

Both internal loopback calibration and clock synchronization can be used to estimate the *available accuracy* of the Output Metric Units. For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.

8.5. Administrative items

8.5.1. Status

\texttt{<current or deprecated>}

8.5.2. Requestor (keep?)

name or RFC, etc.

8.5.3. Revision

1.0
8.5.4. Revision Date

YYYY-MM-DD

8.6. Comments and Remarks

Additional (Informational) details for this entry

9. ICMP Round-trip Latency and Loss Registry Entries

This section specifies three initial registry entries for the ICMP Round-trip Latency, and another entry for ICMP Round-trip Loss Ratio.

This section specifies four Registry entries with many common columns.

All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes two closely-related registry entries. As a result, IANA is also asked to assign four corresponding URNs and URLs to each Named Metric.

9.1. Summary

This category includes multiple indexes to the registry entry: the element ID and metric name.

9.1.1. ID (Identifier)

<insert a numeric identifier, an integer, TBD>

IANA is asked to assign different numeric identifiers to each of the four Named Metrics.

9.1.2. Name

<insert name according to metric naming convention>

RTDelay_Active_IP-ICMP-SendOnRcv_RFCXXXXsecY_Sseconds_<statistic>

where <statistic> is one of:

- Mean
- Min
- Max
9.1.3. URIs

URN: Prefix urn:ietf:metrics:perf:<name>

URL: http://<TBD by IANA>/<name>

9.1.4. Description

RTDelay: This metric assesses the delay of a stream of ICMP packets exchanged between two hosts (which are the two measurement points), and the Output is the Round-trip delay for all successfully exchanged packets expressed as the <statistic> of their conditional delay distribution, where <statistic> is one of:

- Mean
- Min
- Max

RTLoss: This metric assesses the loss ratio of a stream of ICMP packets exchanged between two hosts (which are the two measurement points), and the Output is the Round-trip loss ratio for all successfully exchanged packets expressed as a percentage.

9.1.5. Change Controller

IETF

9.1.6. Version (of Registry Format)

1.0

9.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.

9.2.1. Reference Definition

<Full bibliographic reference to an immutable doc.>


Morton, et al. Expires January 1, 2019
Section 2.4 of [RFC2681] provides the reference definition of the singleton (single value) Round-trip delay metric. Section 3.4 of [RFC2681] provides the reference definition expanded to cover a multi-singleton sample. Note that terms such as singleton and sample are defined in Section 11 of [RFC2330].

Note that although the [RFC2681] definition of "Round-trip-Delay between Src and Dst" is directionally ambiguous in the text, this metric tightens the definition further to recognize that the host in the "Src" role will send the first packet to "Dst", and ultimately receive the corresponding return packet from "Dst" (when neither are lost).

Finally, note that the variable "dT" is used in [RFC2681] to refer to the value of Round-trip delay in metric definitions and methods. The variable "dT" has been re-used in other IPPM literature to refer to different quantities, and cannot be used as a global variable name.


Both delay and loss metrics employ a maximum waiting time for received packets, so the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 6.1 of [RFC6673].

9.2.2. Fixed Parameters

Type-P as defined in Section 13 of [RFC2330]:

- IPv4 header values:
  - DSCP: set to 0
  - TTL: set to 255
  - Protocol: Set to 01 (ICMP)

- IPv6 header values:
* DSCP: set to 0
* Hop Limit: set to 255
* Protocol: Set to 01 (ICMP)

- ICMP header values:
  * Type: 8 (Echo Request)
  * Code: 0
  * Checksum: the checksum MUST be calculated and included in the header
  * (Identifier and Sequence Number set at Run-Time)

- ICMP Payload
  * total of 32 bytes of random info

Other measurement parameters:

- Tmax: a loss threshold waiting time
  * 3.0, expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 4 (see section 9.3 of [RFC6020]) and with resolution of 0.0001 seconds (0.1 ms), with lossless conversion to/from the 32-bit NTP timestamp as per section 6 of [RFC5905].

9.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.

9.3.1. Reference Method

<for metric, insert relevant section references and supplemental info>

The methodology for this metric is defined as Type-P-Round-trip-Delay-Poisson-Stream in section 2.6 of RFC 2681 [RFC2681] and section 3.6 of RFC 2681 [RFC2681] using the Type-P and Tmax defined under Fixed Parameters.
The reference method distinguishes between long-delayed packets and lost packets by implementing a maximum waiting time for packet arrival. Tmax is the waiting time used as the threshold to declare a packet lost. Lost packets SHALL be designated as having undefined delay, and counted for the RTLoss metric.

The calculations on the delay (RTD) SHALL be performed on the conditional distribution, conditioned on successful packet arrival within Tmax. Also, when all packet delays are stored, the process which calculates the RTD value MAY enforce the Tmax threshold on stored values before calculations. See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

The reference method requires some way to distinguish between different packets in a stream to establish correspondence between sending times and receiving times for each successfully-arriving packet. Sequence numbers or other send-order identification MUST be retained at the Src or included with each packet to disambiguate packet reordering if it occurs.

The measurement process will determine the sequence numbers applied to test packets after the Fixed and Runtime parameters are passed to that process. The ICMP measurement process and protocol will dictate the format of sequence numbers and other identifiers.

Refer to Section 4.4 of [RFC6673] for expanded discussion of the instruction to "send a Type-P packet back to the Src as quickly as possible" in Section 2.6 of RFC 2681 [RFC2681]. Section 8 of [RFC6673] presents additional requirements which MUST be included in the method of measurement for this metric.

9.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.

The ICMP metrics use a sending discipline called "SendOnRcv" or Send On Receive. This is a modification of Section 3 of [RFC3432], which prescribes the method for generating Periodic streams using associated parameters:

incT the nominal duration of inter-packet interval, first bit to first bit
dT the duration of the interval for allowed sample start times
T0 the actual start time of the periodic stream

The incT and T0 stream parameters will be specified as Run-time parameters, dT is not used in SendOnRcv.

A SendOnRcv sender behaves exactly like a Periodic stream generator while all reply packets arrive with RTD < incT, and the inter-packet interval will be constant.

If a reply packet arrives with RTD >= incT, then the inter-packet interval for the next sending time is nominally RTD.

If a reply packet fails to arrive within Tmax, then the inter-packet interval for the next sending time is nominally Tmax.

If an immediate send on reply arrival is desired, then set incT=0.

9.3.3. Traffic Filtering (observation) Details

The measured results based on a filtered version of the packets observed, and this section provides the filter details (when present).

<section reference>.

NA

9.3.4. Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

NA

9.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>

Src the IP address of the host in the Src Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])
Dst  the IP address of the host in the Dst Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0 a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Tf is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.

Count  The total count of ICMP Echo Requests to send, formatted as a uint16, as per section 9.2 of [RFC6020].

(see the Packet Stream Generation section for additional Run-time parameters)

9.3.6. Roles

Src  launches each packet and waits for return transmissions from Dst.

Dst  waits for each packet from Src and sends a return packet to Src.

9.4. Output

This category specifies all details of the Output of measurements using the metric.

9.4.1. Type

See subsection titles in Reference Definition for Latency Types.

LossRatio -- the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 6.1 of [RFC6673].

9.4.2. Reference Definition

For all output types ---

T0 the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]).
The UTC Time Zone is required by Section 6.1 of [RFC2330].

TotalCount the count of packets actually sent by the Src to Dst during the measurement interval.

For LossRatio -- the count of lost packets to total packets sent is the basis for the loss ratio calculation as per Section 4.1 of [RFC7600].

For each <statistic>, one of the following sub-sections apply:

9.4.2.1. Mean

The mean SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.2.2 of [RFC6049] for details on calculating this statistic, and 4.2.3 of [RFC6049].

Mean The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

9.4.2.2. Min

The minimum SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for details on calculating this statistic, and 4.3.3 of [RFC6049].
Min The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

9.4.2.3. Max

The maximum SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for calculating this statistic, and 4.3.3 of [RFC6049]. The formula is as follows:

\[
\text{Max} = \text{ FiniteDelay}[j]
\]

such that for some index, \( j \), where \( 1 \leq j \leq N \)
\[\text{ FiniteDelay}[j] \geq \text{ FiniteDelay}[n] \text{ for all } n\]

Max The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

9.4.3. Metric Units

<insert units for the measured results, and the reference specification>.

The <statistic> of Round-trip Delay is expressed in seconds, where <statistic> is one of:

- Mean
- Min
- Max

The Round-trip Loss Ratio is expressed as a percentage of lost packets to total packets sent.
9.4.4. Calibration

Section 3.7.3 of [RFC7679] provides a means to quantify the systematic and random errors of a time measurement. In-situ calibration could be enabled with an internal loopback at the Source host that includes as much of the measurement system as possible, performs address manipulation as needed, and provides some form of isolation (e.g., deterministic delay) to avoid send-receive interface contention. Some portion of the random and systematic error can be characterized this way.

When a measurement controller requests a calibration measurement, the loopback is applied and the result is output in the same format as a normal measurement with additional indication that it is a calibration result.

Both internal loopback calibration and clock synchronization can be used to estimate the *available accuracy* of the Output Metric Units. For example, repeated loopback delay measurements will reveal the portion of the Output result resolution which is the result of system noise, and thus inaccurate.

9.5. Administrative items

9.5.1. Status

<current or deprecated>

9.5.2. Requestor (keep?)

name or RFC, etc.

9.5.3. Revision

1.0

9.5.4. Revision Date

YYYY-MM-DD

9.6. Comments and Remarks

Additional (Informational) details for this entry
10. TCP Round-Trip Delay and Loss Registry Entries

This section specifies three initial registry entries for the Passive assessment of TCP Round-Trip Delay (RTD) and another entry for TCP Round-trip Loss Count.

This section specifies four Registry entries with many common columns.

All column entries beside the ID, Name, Description, and Output Reference Method categories are the same, thus this section proposes four closely-related registry entries. As a result, IANA is also asked to assign four corresponding URNs and URLs to each Named Metric.

10.1. Summary

This category includes multiple indexes to the registry entry: the element ID and metric name.

10.1.1. ID (Identifier)

<insert a numeric identifier, an integer, TBD>

IANA is asked to assign different numeric identifiers to each of the four Named Metrics.

10.1.2. Name

<insert name according to metric naming convention>

RTDelay_Passive_IP-TCP_RFCXXXXsecY_Seconds_\(<statistic>\)

where \(<statistic>\) is one of:

- Mean
- Min
- Max

RTDelay_Passive_IP-TCP-HS_RFCXXXXsecY_Seconds_Singleton

Note that a mid-point observer only has the opportunity to compose a single RTDelay on the TCP Hand Shake.

RTLoss_Passive_IP-TCP_RFCXXXXsecY_Packet_Count
10.1.3. URIs

URN: Prefix urn:ietf:metrics:perf:<name>

URL: http://<TBD by IANA>/<name>

10.1.4. Description

RTDelay: This metric assesses the round-trip delay of TCP packets constituting a single connection, exchanged between two hosts. We consider the measurement of round-trip delay based on a single Observation Point [RFC7011] somewhere in the network. The Output is the Round-trip delay for all successfully exchanged packets expressed as the <statistic> of their conditional delay distribution, where <statistic> is one of:

- Mean
- Min
- Max

RTLoss: This metric assesses the estimated loss count for TCP packets constituting a single connection, exchanged between two hosts. We consider the measurement of round-trip delay based on a single Observation Point [RFC7011] somewhere in the network. The Output is the estimated Loss Count for the measurement interval.

10.1.5. Change Controller

IETF

10.1.6. Version (of Registry Format)

1.0

10.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.

10.2.1. Reference Definitions

<Full bibliographic reference to an immutable doc.>

Although there is no RFC that describes passive measurement of Round-Trip Delay, the parallel definition for Active measurement is:

This metric definition uses the terms singleton and sample as defined in Section 11 of [RFC2330]. (Section 2.4 of [RFC2681] provides the reference definition of the singleton (single value) Round-trip delay metric. Section 3.4 of [RFC2681] provides the reference definition expanded to cover a multi-singleton sample.)

With the Observation Point [RFC7011] (OP) typically located between the hosts participating in the TCP connection, the Round-trip Delay metric requires two individual measurements between the OP and each host, such that the Spatial Composition [RFC6049] of the measurements yields a Round-trip Delay singleton (we are extending the composition of one-way subpath delays to subpath round-trip delay).

Using the direction of TCP SYN transmission to anchor the nomenclature, host A sends the SYN and host B replies with SYN-ACK during connection establishment. The direction of SYN transfer is considered the Forward direction of transmission, from A through OP to B (Reverse is B through OP to A).

Traffic filters reduce the packet stream at the OP to a Qualified bidirectional flow packets.

In the definitions below, Corresponding Packets are transferred in different directions and convey a common value in a TCP header field that establishes correspondence (to the extent possible). Examples may be found in the TCP timestamp fields.

For a real number, RTD_fwd >> the Round-trip Delay in the Forward direction from OP to host B at time T' is RTD_fwd << REQUIRES that OP observed a Qualified Packet to host B at wire-time T’, that host B received that packet and sent a Corresponding Packet back to host A, and OP observed the Corresponding Packet at wire-time T’ + RTD_fwd.

For a real number, RTD_rev >> the Round-trip Delay in the Reverse direction from OP to host A at time T’’ is RTD_rev << REQUIRES that OP observed a Qualified Packet to host A at wire-time T’’, that host A received that packet and sent a Corresponding Packet back to host B, and that OP observed the Corresponding Packet at wire-time T’’ + RTD_rev.
Ideally, the packet sent from host B to host A in both definitions above SHOULD be the same packet (or, when measuring RTD_rev first, the packet from host A to host B in both definitions should be the same).

The REQUIRED Composition Function for a singleton of Round-trip Delay at time T (where T is the earliest of T’ and T’’ above) is:

$$RTDelay = RTD_{fwd} + RTD_{rev}$$

Note that when OP is located at host A or host B, one of the terms composing RTDelay will be zero or negligible.

When the Qualified and Corresponding Packets are a TCP-SYN and a TCP-SYN-ACK, then RTD_fwd == RTD_HS_fwd.

When the Qualified and Corresponding Packets are a TCP-SYN-ACK and a TCP-ACK, then RTD_rev == RTD_HS_rev.

The REQUIRED Composition Function for a singleton of Round-trip Delay for the connection Hand Shake:

$$RTDelay_{HS} = RTD_{HS}_{fwd} + RTD_{HS}_{rev}$$

The definition of Round-trip Loss Count uses the nomenclature developed above, based on observation of the TCP header sequence numbers and storing the sequence number gaps observed. Packet Losses can be inferred from:

- Out-of-order segments: TCP segments are transmitted with monotonically increasing sequence numbers, but these segments may be received out of order. Section 3 of [RFC4737] describes the notion of "next expected" sequence numbers which can be adapted to TCP segments (for the purpose of detecting reordered packets). Observation of out-of-order segments indicates loss on the path prior to the OP, and creates a gap.

- Duplicate segments: Section 2 of [RFC5560] defines identical packets and is suitable for evaluation of TCP packets to detect duplication. Observation of duplicate segments "without a corresponding gap" indicates loss on the path following the OP (because they overlap part of the delivered sequence numbers already observed at OP).

Each observation of an out-of-order or duplicate infers a singleton of loss, but composition of Round-trip Loss Counts will be conducted over a measurement interval which is synonymous with a single TCP connection.
With the above observations in the Forward direction over a measurement interval, the count of out-of-order and duplicate segments is defined as $\text{RTL}_\text{fwd}$. Comparable observations in the Reverse direction are defined as $\text{RTL}_\text{rev}$.

For a measurement interval (corresponding to a single TCP connection), $T_0$ to $T_f$, the REQUIRED Composition Function for the two single-direction counts of inferred loss is:

$$\text{RTL}_\text{Loss} = \text{RTL}_\text{fwd} + \text{RTL}_\text{rev}$$

10.2.2. Fixed Parameters

<list and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed>

Traffic Filters:

- IPv4 header values:
  * DSCP: set to 0
  * Protocol: Set to 06 (TCP)

- IPv6 header values:
  * DSCP: set to 0
  * Protocol: Set to 06 (TCP)

- TCP header values:
  * Flags: ACK, SYN, FIN, others??
  * Timestamp Option (TSopt): Set
    + Kind: 8
    + Length: 10 bytes

10.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.
10.3.1. Reference Methods

The foundation methodology for this metric is defined in Section 4 of [RFC7323] using the Timestamp Option with modifications that allow application at a mid-path Observation Point (OP) [RFC7011]. Further details and applicable heuristics were derived from [Strowes] and [Trammell-14].

The Traffic Filter at the OP is configured to observe a single TCP connection. When the SYN, SYN-ACK, ACK handshake occurs, it offers the first opportunity to measure both RTD_fwd (on the SYN to SYN-ACK pair) and RTD_rev (on the SYN-ACK to ACK pair). Label this singleton of RTDelay as RTDelay_HS (composed using the forward and reverse measurement pair). RTDelay_HS SHALL be treated separately from other RTDelays on data-bearing packets and their ACKs. The RTDelay_HS value MAY be used as a sanity check on other Composed values of RTDelay.

For payload bearing packets, the OP measures the time interval between observation of a packet with Sequence Number s, and the corresponding ACK with same Sequence number. When the payload is transferred from host A to host B, the observed interval is RTD_fwd.

Because many data transfers are unidirectional (say, in the forward direction from host A to host B), it is necessary to use pure ACK packets with Timestamp (TSval) and their Timestamp value echo to perform a RTD_rev measurement. The time interval between observation of the ACK from B to A, and the corresponding packet with Timestamp echo (TSecr) is the RTD_rev.

Delay Measurement Filtering Heuristics:

If Data payloads were transferred in both Forward and Reverse directions, then the Round-Trip Time Measurement Rule in Section 4.1 of [RFC7323] could be applied. This rule essentially excludes any measurement using a packet unless it makes progress in the transfer (advances the left edge of the send window, consistent with [Strowes]).

A different heuristic from [Trammell-14] is to exclude any RTD_rev that is larger than previously observed values. This would tend to exclude Reverse measurements taken when the Application has no data ready to send, because considerable time could be added to RTD_rev from this source of error.
Note that the above heuristic assumes that host A is sending data. Host A expecting a download would mean that this heuristic should be applied to RTD_fwd.

The statistic calculations to summarize the delay (RTDelay) SHALL be performed on the conditional distribution, conditioned on successful Forward and Reverse measurements which follow the Heuristics.

Method for Inferring Loss:

The OP tracks sequence numbers and stores gaps for each direction of transmission, as well as the next-expected sequence number as in [Trammell-14] and [RFC4737]. Loss is inferred from Out-of-order segments and Duplicate segments.

Loss Measurement Filtering Heuristics:

[Trammell-14] adds a window of evaluation based on the RTDelay.

Distinguish Re-ordered from OOO due to loss, because sequence number gap is filled during the same RTDelay window. Segments detected as re-ordered according to [RFC4737] MUST reduce the Loss Count inferred from Out-of-order segments.

Spurious (unneeded) retransmissions (observed as duplicates) can also be reduced this way, as described in [Trammell-14].

Sources of Error:

The principal source of RTDelay error is the host processing time to return a packet that defines the termination of a time interval. The heuristics above intend to mitigate these errors by excluding measurements where host processing time is a significant part of RTD_fwd or RTD_rev.

A key source of RTLoss error is observation loss, described in section 3 of [Trammell-14].

10.3.2. Packet Stream Generation

This section gives the details of the packet traffic which is the basis for measurement. In IPPM metrics, this is called the Stream, and can easily be described by providing the list of stream parameters.
10.3.3. Traffic Filtering (observation) Details

The measured results based on a filtered version of the packets observed, and this section provides the filter details (when present).

The Fixed Parameters above give a portion of the Traffic Filter. Other aspects will be supplied as Run-time Parameters (below).

10.3.4. Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

This metric requires a complete sample of all packets that qualify according to the Traffic Filter criteria.

10.3.5. Run-time Parameters and Data Format

Run-time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete.

<list of run-time parameters, and their data formats>

Src the IP address of the host in the host A Role (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see Section 4 of [RFC6991])

Dst the IP address of the host in the host B (format ipv4-address-no-zone value for IPv4, or ipv6-address-no-zone value for IPv6, see section 4 of [RFC6991])

T0 a time, the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. When T0 is "all-zeros", a start time is unspecified and Td is to be interpreted as the Duration of the measurement interval. The start time is controlled through other means.

Td Optionally, the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]), or the duration (see T0). The UTC Time Zone is required by Section 6.1 of [RFC2330]. Alternatively, the end of the measurement interval MAY be controlled by the measured connection, where the second pair of FIN and ACK packets exchanged between host A and B effectively ends the interval.
TTL or Hop Limit  Set at desired value.

10.3.6. Roles

<lists the names of the different roles from the measurement method>

host A launches the SYN packet to open the connection, and synonymous with an IP address.

host B replies with the SYN-ACK packet to open the connection, and synonymous with an IP address.

10.4. Output

This category specifies all details of the Output of measurements using the metric.

10.4.1. Type

<insert name of the output type, raw or a selected summary statistic>

See subsection titles in Reference Definition for RTDelay Types.

For RTLoss -- the count of lost packets.

10.4.2. Reference Definition

<describe the data format for each type of result>

For all output types ---

T0 the start of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330].

Tf the end of a measurement interval, (format "date-and-time" as specified in Section 5.6 of [RFC3339], see also Section 3 of [RFC6991]). The UTC Time Zone is required by Section 6.1 of [RFC2330]. The end of the measurement interval MAY be controlled by the measured connection, where the second pair of FIN and ACK packets exchanged between host A and B effectively ends the interval.

... ... 

For RTDelay_HS -- the Round trip delay of the Handshake.
For RTLoss -- the count of lost packets.

For each <statistic>, one of the following sub-sections apply:

10.4.2.1. Mean

The mean SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.2.2 of [RFC6049] for details on calculating this statistic, and 4.2.3 of [RFC6049].

Mean  The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

10.4.2.2. Min

The minimum SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:

See section 4.1 of [RFC3393] for details on the conditional distribution to exclude undefined values of delay, and Section 5 of [RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for details on calculating this statistic, and 4.3.3 of [RFC6049].

Min  The time value of the result is expressed in units of seconds, as a positive value of type decimal64 with fraction digits = 9 (see section 9.3 of [RFC6020]) with resolution of 0.000000001 seconds (1.0 ns), and with lossless conversion to/from the 64-bit NTP timestamp as per section 6 of RFC [RFC5905]

10.4.2.3. Max

The maximum SHALL be calculated using the conditional distribution of all packets with a finite value of Round-trip delay (undefined delays are excluded), a single value as follows:
See section 4.1 of [RFC3393] for details on the conditional
distribution to exclude undefined values of delay, and Section 5 of
[RFC6703] for background on this analysis choice.

See section 4.3.2 of [RFC6049] for a closely related method for
calculating this statistic, and 4.3.3 of [RFC6049]. The formula is
as follows:

\[
\text{Max} = (\text{FiniteDelay} [j])
\]

such that for some index, \( j \), where \( 1 \leq j \leq N \)
\[\text{FiniteDelay}[j] \geq \text{FiniteDelay}[n] \text{ for all } n\]

Max The time value of the result is expressed in units of seconds,
as a positive value of type decimal64 with fraction digits = 9
(see section 9.3 of [RFC6020]) with resolution of 0.000000001
seconds (1.0 ns), and with lossless conversion to/from the 64-bit
NTP timestamp as per section 6 of RFC [RFC5905]

10.4.3. Metric Units

<insert units for the measured results, and the reference
specification>.

The <statistic> of Round-trip Delay is expressed in seconds, where
<statistic> is one of:

o Mean

o Min

o Max

The Round-trip Delay of the Hand Shake is expressed in seconds.
The Round-trip Loss Count is expressed as a number of packets.

10.4.4. Calibration

Passive measurements at an OP could be calibrated against an active
measurement (with loss emulation) at host A or B, where the active
measurement represents the ground-truth.

10.5. Administrative items
10.5.1. Status

<current or deprecated>

10.5.2. Requestor (keep?)

name or RFC, etc.

10.5.3. Revision

1.0

10.5.4. Revision Date

YYYY-MM-DD

10.6. Comments and Remarks

Additional (Informational) details for this entry

11. ver08 BLANK Registry Entry

This section gives an initial registry entry for ....

11.1. Summary

This category includes multiple indexes to the registry entries, the element ID and metric name.

11.1.1. ID (Identifier)

<insert numeric identifier, an integer>

11.1.2. Name

<insert name according to metric naming convention>

11.1.3. URIs

URI: Prefix urn:ietf:metrics:perf:<name>

URL:

11.1.4. Description

TBD.
11.1.5. Reference

<reference to the RFC of spec where the registry entry is defined>

11.1.6. Change Controller

<org or person>

11.1.7. Version (of Registry Format)

<currently 1.0>

11.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters.

11.2.1. Reference Definition

<FULL bibliographic reference to an immutable doc.>

<specific section reference and additional clarifications, if needed>

11.2.2. Fixed Parameters

<list and specify Fixed Parameters, input factors that must be determined and embedded in the measurement system for use when needed>

11.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations.

11.3.1. Reference Method

<for metric, insert relevant section references and supplemental info>

11.3.2. Packet Stream Generation

<list of generation parameters and section/spec references if needed>
11.3.3. Traffic Filtering (observation) Details

<insert the measured results based on a filtered version of the packets observed, and this section provides the filter details (when present), and section reference>.

11.3.4. Sampling Distribution

<insert time distribution details, or how this is diff from the filter>

11.3.5. Run-time Parameters and Data Format

<list of run-time parameters, and any reference(s)>

11.3.6. Roles

<lists the names of the different roles from the measurement method>

11.4. Output

This category specifies all details of the Output of measurements using the metric.

11.4.1. Type

<insert name of the output type, raw or a selected summary statistic>

11.4.2. Reference Definition

<pointer to section/spec where output type/format is defined>

11.4.3. Metric Units

<insert units for the measured results, and the reference specification>

11.4.4. Calibration

<describe the error calibration, a way to indicate that the results were collected in a calibration mode of operation, and a way to report internal status metrics related to calibration, such as time offset>
11.5. Administrative items

11.5.1. Status

<current or deprecated>

11.5.2. Requestor

<name of individual or Internet Draft, etc.>

11.5.3. Revision

1.0

11.5.4. Revision Date

YYYY-MM-DD

11.6. Comments and Remarks

Additional (Informational) details for this entry

12. Example RTCP-XR Registry Entry

This section is MAY BE DELETED or adapted before submission.

This section gives an example registry entry for the end-point metric described in RFC 7003 [RFC7003], for RTCP-XR Burst/Gap Discard Metric reporting.

12.1. Registry Indexes

This category includes multiple indexes to the registry entries, the element ID and metric name.

12.1.1. Identifier

An integer having enough digits to uniquely identify each entry in the Registry.

12.1.2. Name

A metric naming convention is TBD.
12.1.3. URI

Prefix urn:ietf:metrics:param:<name>

12.1.4. Status

current

12.1.5. Requestor

Alcelip Mornuley

12.1.6. Revision

1.0

12.1.7. Revision Date

2014-07-04

12.1.8. Description

TBD.

12.1.9. Reference Specification(s)

[RFC3611][RFC4566][RFC6776][RFC6792][RFC7003]

12.2. Metric Definition

This category includes columns to prompt the entry of all necessary details related to the metric definition, including the RFC reference and values of input factors, called fixed parameters. Section 3.2 of [RFC7003] provides the reference information for this category.

12.2.1. Reference Definition

Packets Discarded in Bursts:

The total number of packets discarded during discard bursts. The measured value is unsigned value. If the measured value exceeds 0xFFFFFFFF, the value 0xFFFFFFE MUST be reported to indicate an over-range measurement. If the measurement is unavailable, the value 0xFFFFFFFF MUST be reported.
12.2.2. Fixed Parameters

Fixed Parameters are input factors that must be determined and embedded in the measurement system for use when needed. The values of these parameters is specified in the Registry.

Threshold: 8 bits, set to value = 3 packets.

The Threshold is equivalent to Gmin in [RFC3611], i.e., the number of successive packets that must not be discarded prior to and following a discard packet in order for this discarded packet to be regarded as part of a gap. Note that the Threshold is set in accordance with the Gmin calculation defined in Section 4.7.2 of [RFC3611].

Interval Metric flag: 2 bits, set to value 11=Cumulative Duration

This field is used to indicate whether the burst/gap discard metrics are Sampled, Interval, or Cumulative metrics [RFC6792]:

I=10: Interval Duration - the reported value applies to the most recent measurement interval duration between successive metrics reports.

I=11: Cumulative Duration - the reported value applies to the accumulation period characteristic of cumulative measurements.

Senders MUST NOT use the values I=00 or I=01.

12.3. Method of Measurement

This category includes columns for references to relevant sections of the RFC(s) and any supplemental information needed to ensure an unambiguous methods for implementations. For the Burst/Gap Discard Metric, it appears that the only guidance on methods of measurement is in Section 3.0 of [RFC7003] and its supporting references. Relevant information is repeated below, although there appears to be no section titled "Method of Measurement" in [RFC7003].

12.3.1. Reference Method

Metrics in this block report on burst/gap discard in the stream arriving at the RTP system. Measurements of these metrics are made at the receiving end of the RTP stream. Instances of this metrics block use the synchronization source (SSRC) to refer to the separate auxiliary Measurement Information Block [RFC6776], which describes measurement periods in use (see [RFC6776], Section 4.2).
This metrics block relies on the measurement period in the Measurement Information Block indicating the span of the report. Senders MUST send this block in the same compound RTCP packet as the Measurement Information Block. Receivers MUST verify that the measurement period is received in the same compound RTCP packet as this metrics block. If not, this metrics block MUST be discarded.

12.3.2. Stream Type and Stream Parameters

Since RTCP-XR Measurements are conducted on live RTP traffic, the complete description of the stream is contained in SDP messages that proceed the establishment of a compatible stream between two or more communicating hosts. See Run-time Parameters, below.

12.3.3. Output Type and Data Format

The output type defines the type of result that the metric produces.

- Value: Packets Discarded in Bursts
- Data Format: 24 bits
- Reference: Section 3.2 of [RFC7003]

12.3.4. Metric Units

The measured results are apparently expressed in packets, although there is no section of [RFC7003] titled "Metric Units".

12.3.5. Run-time Parameters and Data Format

Run-Time Parameters are input factors that must be determined, configured into the measurement system, and reported with the results for the context to be complete. However, the values of these parameters is not specified in the Registry, rather these parameters are listed as an aid to the measurement system implementor or user (they must be left as variables, and supplied on execution).

The Data Format of each Run-time Parameter SHALL be specified in this column, to simplify the control and implementation of measurement devices.

- SSRC of Source: 32 bits As defined in Section 4.1 of [RFC3611].
- SDP Parameters: As defined in [RFC4566]
- Session description v= (protocol version number, currently only 0)
o= (originator and session identifier : username, id, version number, network address)

s= (session name : mandatory with at least one UTF-8-encoded character)

i=* (session title or short information) u=* (URI of description)

e=* (zero or more email address with optional name of contacts)

p=* (zero or more phone number with optional name of contacts)

c=* (connection information--not required if included in all media)

b=* (zero or more bandwidth information lines) One or more Time descriptions ("t=" and "r=" lines; see below)

z=* (time zone adjustments)

k=* (encryption key)

a=* (zero or more session attribute lines)

Zero or more Media descriptions (each one starting by an "m=" line; see below)

m= (media name and transport address)

i=* (media title or information field)

k=* (encryption key)

a=* (zero or more media attribute lines -- overriding the Session attribute lines)

An example Run-time SDP description follows:

v=0

o=jdoe 2890844526 2890842807 IN IP4 192.0.2.5

s=SDP Seminar i=A Seminar on the session description protocol
Internet-Draft Initial Registry June 2018

u=http://www.example.com/seminars/sdp.pdf e=j.doe@example.com (Jane Doe)
c=IN IP4 233.252.0.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 99
a=rtpmap:99 h263-1998/90000

12.4. Comments and Remarks

TBD.

13. Revision History

This section may be removed for publication. It contains overview information on updates.

This draft replaced draft-mornuley-ippm-initial-registry.

In version 02, Section 4 has been edited to reflect recent discussion on the ippm-list: * Removed the combination or "Raw" and left 95th percentile. * Hanging Indent on Run-time parameters (Fixed parameters use bullet lists and other indenting formats. * Payload format for measurement has been removed. * Explanation of Conditional delay distribution.

Version 03 addressed Phil Eardley’s comments and suggestions in sections 1-4. and resolved the definition of Percentiles.

Version 04 * All section 4 parameters reference YANG types for alternate data formats. * Discussion has concluded that usecase(s) for machine parse-able registry columns are not needed.

Version 05 * Revised several Poisson streams to Periodic, sections 4 & 5. * Addition of ICMP (ping) metrics in section 9. * First implementation of Passive TCP RTT metrics in section 10.
14. Security Considerations

These registry entries represent no known security implications for Internet Security. Each referenced Metric contains a Security Considerations section.

15. IANA Considerations

IANA is requested to populate The Performance Metric Registry defined in [I-D.ietf-ippm-metric-registry] with the values defined above.

See the IANA Considerations section of [I-D.ietf-ippm-metric-registry] for additional requests and considerations.

16. Acknowledgements

The authors thank Brian Trammell for suggesting the term "Run-time Parameters", which led to the distinction between run-time and fixed parameters implemented in this memo, for identifying the IPFIX metric with Flow Key as an example, for suggesting the Passive TCP RTD metric and supporting references, and for many other productive suggestions. Thanks to Peter Koch, who provided several useful suggestions for disambiguating successive DNS Queries in the DNS Response time metric.

The authors also acknowledge the constructive reviews and helpful suggestions from Barbara Stark, Juergen Schoenwaelder, Tim Carey, and participants in the LMAP working group. Thanks to Michelle Cotton for her early IANA review, and to Amanda Barber for answering questions related to the presentation of the registry and accessibility of the complete template via URL.

17. References

17.1. Normative References

[I-D.ietf-ippm-metric-registry]


17.2. Informative References


Authors' Addresses

Al Morton
AT&T Labs
200 Laurel Avenue South
Middletown,, NJ 07748
USA

Phone: +1 732 420 1571
Fax: +1 732 368 1192
Email: acmorton@att.com
URI: http://home.comcast.net/~acmacm/

Marcelo Bagnulo
Universidad Carlos III de Madrid
Av. Universidad 30
Leganes, Madrid 28911
SPAIN

Phone: 34 91 6249500
Email: marcelo@it.uc3m.es
URI: http://www.it.uc3m.es

Philip Eardley
BT
Adastral Park, Martlesham Heath
Ipswich
ENGLAND

Email: philip.eardley@bt.com

Kevin D’Souza
AT&T Labs
200 Laurel Avenue South
Middletown,, NJ 07748
USA

Phone: +1 732 420 xxxx
Email: kld@att.com
Considerations for Selecting RTCP Extended Report (XR) Metrics for the WebRTC Statistics API
draft-ietf-xrblock-rtcweb-rtcp-xr-metrics-10

Abstract

This document describes monitoring features related to media streams in Web real-time communication (WebRTC). It provides a list of RTCP Sender Report, Receiver Report and Extended Report metrics, which may need to be supported by RTP implementations in some diverse environments. It lists a set of identifiers for the WebRTC’s statistics API. These identifiers are a set of RTCP SR, RR, and XR metrics related to the transport of multimedia flows.

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 http://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 January 21, 2018.

Copyright Notice

Singh, et al. Expires November 26, 2018
Table of Contents

1. Introduction ................................................. 3
2. Terminology ................................................... 3
3. RTP Statistics in WebRTC Implementations ...................... 3
4. Considerations for Impact of Measurement Interval ............ 4
5. Candidate Metrics ............................................ 5
   5.1. Network Impact Metrics .................................. 5
      5.1.1. Loss and Discard Packet Count Metric ............... 5
      5.1.2. Burst/Gap Pattern Metrics for Loss and Discard .... 6
      5.1.3. Run Length Encoded Metrics for Loss, Discard ... 7
   5.2. Application Impact Metrics .............................. 7
      5.2.1. Discard Octets Metric ............................... 7
      5.2.2. Frame Impairment Summary Metrics .................... 8
      5.2.3. Jitter Buffer Metrics ............................... 8
   5.3. Recovery metrics ........................................ 9
      5.3.1. Post-repair Packet Count Metrics ................... 9
      5.3.2. Run Length Encoded Metric for Post-repair .......... 9
6. Identifiers from Sender, Receiver, and Extended Report Blocks 10
   6.1. Cumulative Number of Packets and Octets Sent ............. 10
   6.2. Cumulative Number of Packets and Octets Received ....... 10
   6.3. Cumulative Number of Packets Lost ...................... 11
   6.4. Interval Packet Loss and Jitter ........................ 11
   6.5. Cumulative Number of Packets and Octets Discarded ... 11
   6.6. Cumulative Number of Packets Repaired .................. 11
   6.7. Burst Packet Loss and Burst Discards ................... 11
   6.8. Burst/Gap Rates ....................................... 12
   6.9. Frame Impairment Metrics ............................... 12
7. Adding new metrics to WebRTC Statistics API ................... 13
8. Security Considerations ....................................... 13
9. Acknowledgements ............................................. 13
10. References .................................................. 13
    10.1. Normative References ................................... 13
    10.2. Informative References ................................. 15
1. Introduction

Web real-time communication (WebRTC) [I-D.ietf-rtcweb-overview] deployments are emerging and applications need to be able to estimate the service quality. If sufficient information (metrics or statistics) is provided to the application, it can attempt to improve the media quality. [RFC7478] specifies a requirement for statistics:

F38 The browser must be able to collect statistics, related to the transport of audio and video between peers, needed to estimate quality of experience.

The WebRTC Stats API [W3C.WD-webrtc-stats] currently lists metrics reported in the RTCP Sender and Receiver Report (SR/RR) [RFC3550] to fulfill this requirement. However, the basic metrics from RTCP SR/RR are not sufficient for precise quality monitoring, or diagnosing potential issues.

Standards such as "RTP Control Protocol Extended Reports (RTCP XR)" [RFC3611] as well as other extensions standardized in the XRBLOCK working group, e.g., burst/gap loss metric reporting [RFC6958], burst/gap discard metric reporting [RFC7003], and etc., have been produced for the purpose of collecting and reporting performance metrics from RTP endpoint devices that can be used to have a end-to-end service visibility and measure the delivering quality in various RTP services. These metrics are able to complement those in [RFC3550].

In this document, we provide rationale for choosing additional RTP metrics for the WebRTC getStats() API [W3C.WD-webrtc]. All identifiers proposed in this document are recommended to be implemented by an WebRTC endpoint. An endpoint may choose not to expose an identifier if it does not implement the corresponding RTCP Report. This document only considers RTP layer metrics. Other metrics, e.g., IP layer metrics, are out of scope.

2. Terminology

ReportGroup: It is a set of metrics identified by a common Synchronization source (SSRC).
3. RTP Statistics in WebRTC Implementations

The RTCP Sender Reports (SRs) and Receiver Reports (RRs) [RFC3550] expose the basic metrics for the local and remote media streams. However, these metrics provide only partial or limited information, which may not be sufficient for diagnosing problems or quality monitoring. For example, it may be useful to distinguish between packets lost and packets discarded due to late arrival. Even though they have the same impact on the multimedia quality, it helps in identifying and diagnosing problems. RTP Control Protocol Extended Reports (XRs) [RFC3611] and other extensions discussed in the XRBLOCK working group provide more detailed statistics, which complement the basic metrics reported in the RTCP SR and RRs.

The WebRTC application extracts the statistic from the browser by querying the getStats() API [W3C.WD-webrtc]. The browser can easily report the local variables i.e., the statistics related to the outgoing RTP media streams and the incoming RTP media streams. However, without the support of RTCP XRs or some other signaling mechanism, the WebRTC application cannot expose the remote endpoints’ statistics. [I-D.ietf-rtcweb-rtp-usage] does not mandate the use of any RTCP XRs and their usage is optional. If the use of RTCP XRs is successfully negotiated between endpoints (via SDP), thereafter the application has access to both local and remote statistics. Alternatively, once the WebRTC application gets the local information, they can report it to an application server or a third-party monitoring system, which provides quality estimations or diagnosis services for application developers. The exchange of statistics between endpoints or between a monitoring server and an endpoint is outside the scope of this document.

4. Considerations for Impact of Measurement Interval

RTCP extensions like RTCP XR usually share the same timing interval with the RTCP SR/RR, i.e., they are sent as compound packets, together with the RTCP SR/RR. Alternatively, if the RTCP XR uses a different measurement interval, all XRs using the same measurement interval are compounded together and the measurement interval is indicated in a specific measurement information block defined in [RFC6776].

When using WebRTC getStats() APIs (see section 7 of [W3C.WD-webrtc]), the applications can query this information at arbitrary intervals. For the statistics reported by the remote endpoint, e.g., those conveyed in an RTCP SR/RR/XR, these will not change until the next RTCP report is received. However, statistics generated by the local endpoint have no such restrictions as long as the endpoint is sending and receiving media. For example, an application may choose to poll
the stack for statistics every 1 second. In this case the underlying stack local will return the current snapshot of the local statistics (for incoming and outgoing media streams). However, it may return the same remote statistics as before for the remote statistics, as no new RTCP reports may have been received in the past 1 second. This can occur when the polling interval is shorter than the average RTCP reporting interval.

5. Candidate Metrics

Since the following metrics are all defined in RTCP XR, which is not mandated in WebRTC, all of them are local. However, if RTCP XR is supported by negotiation between two browsers, the following metrics can also be generated remotely and be sent to local by RTCP XR packets.

The following metrics are classified into 3 categories: network impact metrics, application impact metrics and recovery metrics. Network impact metrics are the statistics recording the information only for network transmission. They are useful for network problem diagnosis. Application impact metrics mainly collect the information from the viewpoint of application, e.g., bit rate, frame rate or jitter buffers. Recovery metrics reflect how well the repair mechanisms perform, e.g. loss concealment, retransmission or Forward Error Correction (FEC). All of the 3 types of metrics are useful for quality estimations of services in WebRTC implementations. WebRTC applications can use these metrics to calculate the estimated Mean Opinion Score (MOS) [ITU-T P.800.1] values or Media Delivery Index (MDI) [RFC4445] for their services.

5.1. Network Impact Metrics

5.1.1. Loss and Discard Packet Count Metric

In multimedia transport, packets which are received abnormally are classified into 3 types: lost, discarded and duplicate packets. Packet loss may be caused by network device breakdown, bit-error corruption or network congestion (packets dropped by an intermediate router queue). Duplicate packets may be a result of network delays that causes the sender to retransmit the original packets. Discarded packets are packets that have been delayed long enough (perhaps they missed the playout time) and are considered useless by the receiver. Lost and discarded packets cause problems for multimedia services, as missing data and long delays can cause degradation in service quality, e.g., missing large blocks of contiguous packets (lost or discarded) may cause choppy audio, and long network transmission delay time may cause audio or video buffering. The RTCP SR/RR defines a metric for counting the total number of RTP data packets
that have been lost since the beginning of reception. But this statistic does not distinguish lost packets from discarded and duplicate packets. Packets that arrive late will be discarded and are not reported as lost, and duplicate packets will be regarded as a normally received packet. Hence, the loss metric can be misleading if many duplicate packets are received or packets are discarded, which causes the quality of the media transport to appear okay from the statistic point of view, but meanwhile the users may actually be experiencing bad service quality. So in such cases, it is better to use more accurate metrics in addition to those defined in RTCP SR/RR.

The lost packets and duplicated packets metrics defined in Statistics Summary Report Block of [RFC3611] extend the information of loss carried in standard RTCP SR/RR. They explicitly give an account of lost and duplicated packets. Lost packet counts are useful for network problem diagnosis. It is better to use the loss packets metrics of [RFC3611] to indicate the packet lost count instead of the cumulative number of packets lost metric of [RFC3550]. Duplicated packets are usually rare and have little effect on QoS evaluation. So it may not be suitable for use in WebRTC.

Using loss metrics without considering discard metrics may result in inaccurate quality evaluation, as packet discard due to jitter is often more prevalent than packet loss in modern IP networks. The discarded metric specified in [RFC7002] counts the number of packets discarded due to the jitter. It augments the loss statistics metrics specified in standard RTCP SR/RR. For those RTCWEB services with jitter buffers requiring precise quality evaluation and accurate troubleshooting, this metric is useful as a complement to the metrics of RTCP SR/RR.

5.1.2. Burst/Gap Pattern Metrics for Loss and Discard

RTCP SR/RR defines coarse metrics regarding loss statistics: the metrics are all about per call statistics and are not detailed enough to capture the transitory nature of some impairments like bursty packet loss. Even if the average packet loss rate is low, the lost packets may occur during short dense periods, resulting in short periods of degraded quality. Bursts cause lower quality experience than the non-bursts for low packet loss rates, whereas for high packet loss rates the converse is true. So capturing burst gap information is very helpful for quality evaluation and locating impairments. If the WebRTC application needs to evaluate the services quality, burst gap metrics provides more accurate information than RTCP SR/RR.

[RFC3611] introduces burst gap metrics in VoIP report block. These metrics record the density and duration of burst and gap periods,
which are helpful in isolating network problems since bursts correspond to periods of time during which the packet loss/discard rate is high enough to produce noticeable degradation in audio or video quality. Burst gap related metrics are also introduced in [RFC7003] and [RFC6958] which define two new report blocks for usage in a range of RTP applications beyond those described in [RFC3611]. These metrics distinguish discarded packets from loss packets that occur in the bursts period and provides more information for diagnosing network problems. Additionally, the block reports the frequency of burst events which is useful information for evaluating the quality of experience. Hence, if WebRTC applications need to do quality evaluation and observe when and why quality degrades, these metrics should be considered.

5.1.3. Run Length Encoded Metrics for Loss, Discard

Run-length encoding uses a bit vector to encode information about the packet. Each bit in the vector represents a packet and depending on the signaled metric it defines if the packet was lost, duplicated, discarded, or repaired. An endpoint typically uses the run length encoding to accurately communicate the status of each packet in the interval to the other endpoint. [RFC3611], [RFC7097] define run-length encoding for lost and duplicate packets, and discarded packets, respectively.

The WebRTC application could benefit from the additional information. If losses occur after discards, an endpoint may be able to correlate the two run length vectors to identify congestion-related losses, e.g., a router queue became overloaded causing delays and then overflowed. If the losses are independent, it may indicate bit-error corruption. For the WebRTC Stats API [W3C.WD-webrtc-stats], these types of metrics are not recommended for use due to the large amount of data and the computation involved.

5.2. Application Impact Metrics

5.2.1. Discarded Octets Metric

The metric reports the cumulative size of the packets discarded in the interval. It is complementary to number of discarded packets. An application measures sent octets and received octets to calculate sending rate and receiving rate, respectively. The application can calculate the actual bit rate in a particular interval by subtracting the discarded octets from the received octets.

For WebRTC, discarded octets supplements the sent and received octets and provides an accurate method for calculating the actual bit rate which is an important parameter to reflect the quality of the media.
The discarded bytes metric is defined in [RFC7243].

5.2.2. Frame Impairment Summary Metrics

RTP has different framing mechanisms for different payload types. For audio streams, a single RTP packet may contain one or multiple audio frames. On the other hand, in video streams, a single video frame may be transmitted in multiple RTP packets. The size of each packet is limited by the Maximum Transmission Unit (MTU) of the underlying network. However, statistics from standard SR/RR only collect information from transport layer, which may not fully reflect the quality observed by the application. Video is typically encoded using two frame types i.e., key frames and derived frames. Key frames are normally just spatially compressed, i.e., without prediction from other pictures. The derived frames are temporally compressed, i.e., depend on the key frame for decoding. Hence, key frames are much larger in size than derived frames. The loss of these key frames results in a substantial reduction in video quality. Thus it is reasonable to consider this application layer information in WebRTC implementations, which influence sender strategies to mitigate the problem or require the accurate assessment of users’ quality of experience.

The metrics in this category include: number of discarded key frames, number of lost key frames, number of discarded derived frames, number of lost derived frames. These metrics can be used to calculate Media Loss Rate (MLR) of MDI [RFC4445]. Details of the definition of these metrics are described in [RFC7003]. Additionally, the metric provides the rendered frame rate, an important parameter for quality estimation.

5.2.3. Jitter Buffer Metrics

The size of the jitter buffer affects the end-to-end delay on the network and also the packet discard rate. When the buffer size is too small, slower packets are not played out and dropped, while when the buffer size is too large, packets are held longer than necessary and consequently reduce conversational quality. Measurement of jitter buffer should not be ignored in the evaluation of end user perception of conversational quality. Jitter buffer related metrics, such as maximum and nominal jitter buffer, could be used to show how the jitter buffer behaves at the receiving endpoint. They are useful for providing better end-user quality of experience (QoE) when jitter buffer factors are used as inputs to calculate estimated MOS values. Thus for those cases, jitter buffer metrics should be considered.

The definition of these metrics is provided in [RFC7005].

5.3. Recovery metrics
This document does not consider concealment metrics [RFC7294] as part of recovery metrics.

5.3.1. Post-repair Packet Count Metrics

Web applications can support certain RTP error-resilience mechanisms following the recommendations specified in [draft-ietf-rtcweb-rtp-usage]. For these web applications using repair mechanisms, providing some statistic information for the performance of their repair mechanisms could help to have a more accurate quality evaluation.

The unrepaired packet count and repaired loss count defined in [RFC7509] provide the recovery information of the error-resilience mechanisms to the monitoring application or the sending endpoint. The endpoint can use these metrics to ascertain the ratio of repaired packets to lost packets. Including post-repair packet count metrics helps the application evaluate the effectiveness of the applied repair mechanisms.

5.3.2. Run Length Encoded Metric for Post-repair

[RFC5725] defines run-length encoding for post-repair packets. When using error-resilience mechanisms, the endpoint can correlate the loss run length with this metric to ascertain where the losses and repairs occurred in the interval. This provides more accurate information for recovery mechanisms evaluation than those in Section 5.3.1. However, it is not suggested to use due to their enormous amount of data when RTCP XR are supported.

For WebRTC, the application may benefit from the additional information. If losses occur after discards, an endpoint may be able to correlate the two run length vectors to identify congestion-related losses, e.g., a router queue became overloaded causing delays and then overflowed. If the losses are independent, it may indicate bit-error corruption. Lastly, when using error-resilience mechanisms, the endpoint can correlate the loss and post-repair run lengths to ascertain where the losses and repairs occurred in the interval. For example, consecutive losses are likely not to be repaired by a simple FEC scheme.

6. Identifiers from Sender, Receiver, and Extended Report Blocks

This document describes a list of metrics and corresponding identifiers relevant to RTP media in WebRTC. This group of identifiers are defined on a ReportGroup corresponding to a synchronization source (SSRC). In practice the application needs to be able to query the statistic identifiers on both an incoming (remote) and outgoing (local) media stream. Since sending and
receiving SR and RR are mandatory, the metrics defined in the SR and RR report blocks are always available. For XR metrics, it depends on two factors: 1) if it is measured at the endpoint, 2) if it is reported by the endpoint in an XR report. If a metric is only measured by the endpoint and not reported, the metrics will only be available for the incoming (remote) media stream. Alternatively, if the corresponding metric is also reported in an XR report, it will be available for both the incoming (remote) and outgoing (local) media stream.

For a remote statistic, the timestamp represents the timestamp from an incoming SR/RR/XR packet. Conversely, for a local statistic, it refers to the current timestamp generated by the local clock (typically the POSIX timestamp, i.e., milliseconds since Jan 1, 1970).

As per [RFC3550], the octets metrics represent the payload size (i.e., not including header or padding).

6.1. Cumulative Number of Packets and Octets Sent

Name: packetsSent
Definition: section 6.4.1 in [RFC3550].

Name: bytesSent
Definition: section 6.4.1 in [RFC3550].

6.2. Cumulative Number of Packets and Octets Received

Name: packetsReceived
Definition: section 6.4.1 in [RFC3550].

Name: bytesReceived
Definition: section 6.4.1 in [RFC3550].
6.3. Cumulative Number of Packets Lost

Name: packetsLost

Definition: section 6.4.1 in [RFC3550].

6.4. Interval Packet Loss and Jitter

Name: jitter

Definition: section 6.4.1 in [RFC3550].

Name: fractionLost

Definition: section 6.4.1 in [RFC3550].

6.5. Cumulative Number of Packets and Octets Discarded

Name: packetsDiscarded

Definition: The cumulative number of RTP packets discarded due to late or early-arrival, Appendix A (a) of [RFC7002].

Name: bytesDiscarded

Definition: The cumulative number of octets discarded due to late or early-arrival, Appendix A of [RFC7243].

6.6. Cumulative Number of Packets Repaired

Name: packetsRepaired

Definition: The cumulative number of lost RTP packets repaired after applying a error-resilience mechanism, Appendix A (b) of [RFC7509]. To clarify, the value is upper bound to the cumulative number of lost packets.

6.7. Burst Packet Loss and Burst Discards

Name: burstPacketsLost

Definition: The cumulative number of RTP packets lost during loss bursts, Appendix A (c) of [RFC6958].

Name: burstLossCount

Definition: The cumulative number of bursts of lost RTP packets, Appendix A (e) of [RFC6958].
Name: burstPacketsDiscarded
Definition: The cumulative number of RTP packets discarded during discard bursts, Appendix A (b) of [RFC7003].

Name: burstDiscardCount
Definition: The cumulative number of bursts of discarded RTP packets, Appendix A (e) of [RFC8015].

[RFC3611] recommends a Gmin (threshold) value of 16 for classifying packet loss or discard burst.

6.8. Burst/Gap Rates

Name: burstLossRate
Definition: The fraction of RTP packets lost during bursts, Appendix A (a) of [RFC7004].

Name: gapLossRate
Definition: The fraction of RTP packets lost during gaps, Appendix A (b) of [RFC7004].

Name: burstDiscardRate
Definition: The fraction of RTP packets discarded during bursts, Appendix A (e) of [RFC7004].

Name: gapDiscardRate
Definition: The fraction of RTP packets discarded during gaps, Appendix A (f) of [RFC7004].

6.9. Frame Impairment Metrics

Name: framesLost
Definition: The cumulative number of full frames lost, Appendix A (i) of [RFC7004].

Name: framesCorrupted
Definition: The cumulative number of frames partially lost, Appendix A (j) of [RFC7004].

Name: framesDropped
Definition: The cumulative number of full frames discarded, Appendix A (g) of [RFC7004].

Name: framesSent

Definition: The cumulative number of frames sent.

Name: framesReceived

Definition: The cumulative number of partial or full frames received.

7. Adding new metrics to WebRTC Statistics API

During the progress of this work, the metrics defined in this draft have already been added to the W3C WebRTC specification. The working process to add new metrics for future is to create an issue or pull request on the repository of the W3C WebRTC specification (https://github.com/w3c/webrtc-stats).

8. Security Considerations

This document focuses on listing the RTCP XR metrics defined in the corresponding RTCP reporting extensions and do not give rise to any new security vulnerabilities beyond those described in [RFC3611] and [RFC6792].

The overall security considerations for RTP used in WebRTC applications is described in [I-D.ietf-rtcweb-rtp-usage] and [I-D.ietf-rtcweb-security], which are also apply to this memo.

9. IANA Consideration

This document requests no action by IANA.

10. Acknowledgements

The authors would like to thank Bernard Aboba, Harald Alvestrand, Al Morton, Colin Perkins, and Shida Schubert for their valuable comments and suggestions on earlier version of this document.

11. References

11.1. Normative References


Internet-Draft         RTCP XR Metrics for RTCWEB           May 25, 2018

Packets", RFC 7097, DOI 10.17487/RFC7097, January 2014,

Protocol (RTCP) Extended Report (XR) Block for the Bytes
Discarded Metric", RFC 7243, DOI 10.17487/RFC7243, May

Extended Report (XR) for Post-Repair Loss Count Metrics",
RFC 7509, DOI 10.17487/RFC7509, May 2015,

Control Protocol (RTCP) Extended Report (XR) Block for
Independent Reporting of Burst/Gap Discard Metrics",
RFC 8015, DOI 10.17487/RFC8015, November 2016,

11.2. Informative References

[I-D.ietf-rtcweb-overview] H. Alverstrand, "Overview: Real Time
Protocols for Browser-based Applications", draft-ietf-

[ITU-T P.800.1] "Mean Opinion Score (MOS) terminology", ITU-T
P.800.1, July 2016.

(MDI)", RFC4445, April 2006.

[I-D.ietf-rtcweb-rtp-usage] Perkins, C., Westerlund, M., and J. Ott,
"Web Real-Time Communication (WebRTC): Media Transport and

[I-D.ietf-rtcweb-security] Rescorla, E., "Security Considerations for

[RFC7294] Clark, A., Zorn, G., Bi, C. and Q. Wu, "RTP Control
Protocol (RTCP) Extended Report (XR) Blocks for
Concealment Metrics Reporting on Audio Applications",

[RFC7478] Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-
Time Communication Use Cases and Requirements", RFC 7478,


Authors’ Addresses

Varun Singh  
CALLSTATS I/O Oy  
Annankatu 31-33 C 42  
Helsinki 00100  
Finland  
Email: varun@callstats.io  
URI: https://www.callstats.io/about

Rachel Huang  
Huawei  
101 Software Avenue, Yuhua District  
Nanjing, CN 210012  
China  
Email: rachel.huang@huawei.com

Roni Even  
Huawei  
14 David Hamelech  
Tel Aviv 64953  
Israel  
Email: roni.even@huawei.com

Dan Romascanu
RTP Control Protocol (RTCP) Extended Report (XR) Block for Effective Loss Index Reporting
draft-zheng-xrblock-effective-loss-index-02

Abstract

This document defines a new metric for RTP monitors to estimate the effectiveness of stream repair means, and an RTP Control Protocol (RTCP) Extended Report (XR) Block to report the metric.

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 http://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 May 3, 2018.

Copyright Notice

Copyright (c) 2018 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 (http://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
1. Introduction

RTP applications often use stream repair means, e.g. FEC (Forward Error Correction) [RFC5109] and/or retransmission [RFC4588] to improve the robustness of media streams. With the presence of those stream repair means, a degree of packet loss can be recovered for a media stream. In the past, some RTCP Extend Reports (XRs) were defined to reflect the situation of post-repair loss. For example, [RFC5725] defines an XR block using Run Length Encoding (RLE) to report post-repair loss; [RFC7509] defines count metrics for post-repair loss.

This document proposes a new metric Effective Loss Index (ELI) to estimate the effectiveness of stream repair means by calculating the probability of the post-repair losses. The new metric provides a simpler view on the post-repair loss than the mechanisms documented in [RFC5725] and [RFC7509]. ELI is an index, so the values reported from the monitors deployed in the different places in the network can be compared and analyzed.
be compared directly, which makes it easier to diagnose the network problem when delivering the RTP streams. A use case is to compare the ELI value reported by a monitor in the network with a certain reasonable threshold to see if there are any problems in the IPTV services. For those endpoints, more informative XR reports such as those in [RFC5725] and [RFC7509] can then be used to discover more details about the loss situations.

This document also defines in section 3, an XR block to report the metric.

1.1. Effective Loss Index

Effective Loss Index (ELI) uses a simple model to measure the loss impact after applying loss repair of loss repair. It is useful especially in the middleboxes which usually are passive observer and do not have the ability to recover the loss data.

The model assumes that repair means are applied onto packets by batches of equal size. Lower ELI means that loss impact is minimal. Specifically, a batch is identified by a range of RTP sequence numbers. The size of a batch is number of packets. An application can agree upon a default batch size, or use the SDP signaling defined in Section 4.1 to communicate one if the middlebox can see the SDP, or just configure it.

An RTP endpoint is assumed to process received packets and apply repair means batch by batch. For each batch, if there is still some unrecoverable loss after having applied the repair means, then the repair means are deemed as ineffective. The ineffectiveness is denoted by Effective Loss Factor (ELF), along with a parameter Loss Repair Threshold, showing below:

\[
\text{if Loss Packets Number} \text{ > Loss Repair Threshold} \\
\quad \text{Effective Loss Factor} = 1 \\
\text{else} \\
\quad \text{Effective Loss Factor} = 0 \\
\text{endif}
\]

Figure 1: Calculation of Effective Loss Factor

The parameters in Figure 1 are explained below:

- Loss Packets Number is the number of packet lost in the batch.
- Loss Repair Threshold indicates the maximum loss packets number that can be recovered.
The minimum value of Loss Repair Threshold is zero, which means there is no loss repair. This document does not mandate any value for Loss Repair Threshold. Applications can prescribe a value for themselves without signaling. For example, it can be calculated by the batch size multiplied by the fixed redundancy ratio of the FEC algorithm; and when used in the retransmission case, it can be set to the maximum number of lost packets to be retransmitted in a batch. On the other hand, SDP signaling defined in Section 4.1 can be used to communicate the value.

Effective Loss Index is an integer derived by calculating the average Effective Loss Factor across a sequence of consecutive batches of RTP packets. Let \( \text{ELF}(i) \) be the Effective Loss Factor calculated for \( i \)-th batch, and \( N \) as number of batches in the sequence, then Effective Loss Index is calculated as:

\[
\text{Effective Loss Index} = \frac{\text{ELF}(1)+\text{ELF}(2)+...+\text{ELF}(N)}{N}
\]

Figure 2: Calculation of Effective Loss Index

The following is an example of how to calculate Effective Loss Index. For simplicity and demonstration purpose, the size of a batch is assumed to be 3, and the Loss Repair Threshold is assumed to be 1. The example processes a sequence of 9 RTP packets (x means lost) in 7 batches.

\[
\begin{array}{|c|c|c|}
\hline
\text{Batch} & \text{Loss} & \text{Effective Loss Factor} \\
\hline
1 & 2, 3 & 1 \\
2 & 3, 4 & 1 \\
3 & 4, 5 & 0 \\
4 & 5, 6 & 0 \\
5 & 6, 7 & 1 \\
6 & 7, 8 & 0 \\
7 & 8, 9 & 0 \\
\hline
\end{array}
\]

Effective Loss Index = \[
\frac{1+1+0+0+1+0+0}{7}
\] = 0.4285

This example provides fine grained estimation for loss recovery. It can detect the loss burst happening over batches. Implementations can also do coarse grained estimation by simply dividing total packets into several batches.
1.2. Applicability

The metric defined by this document is applicable to a range of RTP applications that send packets with stream repair means (e.g., Forward Error Correction (FEC) [RFC5109] and/or retransmission [RFC4588]) applied on them. Note that this metric is only valuable for FECs where the redundant data are sent in a different RTP stream from the original media stream.

This document does not mandate any value for the batch size. Applications can prescribe a value for themselves without signaling. For example, the batch size can be set to the number of packets containing source symbols in a source block in the case of FEC, and can be prescribed arbitrarily, e.g. 100, in the case of retransmission.

The number of batches among which ELI is calculated should not be too few, otherwise the result may be biased. It is suggested to calculate it based on the total number of RTP packets during the measurement interval, as in the section 1.1 example:

The number of batches = (The total number of RTP packets - the size of a batch) + 1.

1.3. RTCP and RTCP XR Reports

The use of RTCP for reporting is defined in [RFC3550]. [RFC3611] defines an extensible structure for reporting by using an RTCP Extended Report (XR). This document defines a new Extended Report block for use with [RFC3550] and [RFC3611].

1.4. Performance Metrics Framework

The Performance Metrics Framework [RFC6390] provides guidance on the definition and specification of performance metrics. The "Guidelines for Use of the RTP Monitoring Framework" [RFC6792] provides guidelines for reporting block format using RTCP XR. The Metrics Block described in this document is in accordance with the guidelines in [RFC6390] and [RFC6792].

2. 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].

3. Effective Loss Index Report Block
The Effective Loss Index Report Block has the following format:

```
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     BT=TBD    |   Reserved    |      Block length = 3         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       SSRC of Source                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Effective Loss Index       |          Padding              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- **Block Type (BT):** 8 bits: An Effect Loss Index Report Block is identified by the constant ‘TBD’.
  
  [[Editor Note: should replace ‘TBD’ with assigned value]]

- **Reserved:** 8 bits: These bits are reserved for future use. They MUST be set to zero by senders and ignored by receivers (see Section 4.2 of [RFC6709]).

- **Block length:** 16 bits: This field is in accordance with the definition in [RFC3611]. In this report block, it MUST be set to 3. The block MUST be discarded if the block length is set to a different value.

- **SSRC of source:** 32 bits: The SSRC of the RTP data packet source being reported upon by this report block, as defined in Section 4.1 of [RFC3611].

- **Effective Loss Index:** 16 bits: The value of Effective Loss Index, equivalent to taking the integer part after multiplying the calculated result of Effective Loss Index (as in Figure 2) by 65535.

- **Padding:** 16 bits: These bits MUST be set to zero by senders and ignored by receivers.

4. **SDP Signaling**

[RFC3611] defines the use of SDP (Session Description Protocol) for signaling the use of RTCP XR blocks. However, XR blocks MAY be used without prior signaling (see Section 5 of [RFC3611]).

4.1. **SDP rtcp-xr-attrib Attribute Extension**
This session augments the SDP attribute "rtcp-xr" defined in Section 5.1 of [RFC3611] by providing an additional value of "xr-format" to signal the use of the report block defined in this document. The ABNF [RFC5234] syntax is as follows.

xr-format =/ xr-eli-block

xr-eli-block = "effective-loss-index"
[ ":" effective-loss-batch-size]
[ ">" effective-loss-threshold]

effective-loss-batch-size  = 1*DIGIT
; the batch size is in number of packets

effective-loss-threshold   = 1*DIGIT
; the threshold is in number of packets

DIGIT = %x30-39

The SDP attribute "xr-eli-block" is designed to contain two optional values, one for signaling the batch size, another for the Effective Loss Threshold. Here are some examples:
1. signaling both batch size (100) and Effective Loss Threshold (2)

\[
xr-eli-block = "effective-loss-index" : "100" > "2"
\]

2. signaling only batch size (100)

\[
xr-eli-block = "effective-loss-index" : "100"
\]

3. signaling only Effective Loss Threshold (2)

\[
xr-eli-block = "effective-loss-index" > "2"
\]

4.2. Offer/Answer Usage

When SDP is used in offer/answer context, the SDP Offer/Answer usage defined in [RFC3611] for the unilateral "rtcp-xr" attribute parameters applies. For detailed usage of Offer/Answer for unilateral parameters, refer to Section 5.2 of [RFC3611].

5. Security Considerations

This proposed RTCP XR block introduces no new security considerations beyond those described in [RFC3611]. This block does not provide per-packet statistics, so the risk to confidentiality documented in Section 7, paragraph 3 of [RFC3611] does not apply.

An attacker may put incorrect information in the Effective Loss Index reports. Implementers should consider the guidance in [RFC7202] for using appropriate security mechanisms, i.e., where security is a concern, the implementation should apply encryption and authentication to the report block. For example, this can be achieved by using the AVPF profile together with the Secure RTP profile as defined in [RFC3711] an appropriate combination of the two profiles (an "SAVPF") is specified in [RFC5124]. However, other mechanisms also exist (documented in [RFC7201]) and might be more suitable.

6. IANA Considerations

New block types for RTCP XR are subject to IANA registration. For general guidelines on IANA considerations for RTCP XR, refer to [RFC3611].

6.1. New RTCP XR Block Type Value

This document assigns the block type value ‘TBD’ in the IANA "RTP Control Protocol Extended Reports (RTCP XR) Block Type Registry" to the "Post-Repair Loss Count Metrics Report Block".
6.2. New RTCP XR SDP Parameter

This document also registers a new parameter "effective-loss-index" in the "RTP Control Protocol Extended Reports (RTCP XR) Session Description Protocol (SDP) Parameters Registry".

6.3. Contact Information for Registrations

The contact information for the registrations is:

RAI Area Directors <rai-ads@ietf.org>

7. Acknowledgements

This document has benefited greatly from the comments of various people. The following individuals have contributed to this document: Colin Perkins, Yanfang Zhang.

8. References

8.1. Normative References


8.2. Informative References


Appendix A. Metric Represented Using the Template from RFC 6390

A.1. Effective Loss Index

- Metric Name: RTP Effective Loss Index.
- Metric Description: The effectiveness of stream repair means applied on a sequence of RTP packets.
- Method of Measurement or Calculation: See the "Effective Loss Index" definition in Section 1.1. It is directly measured and must be measured for the primary source RTP packets with no further chance of repair.
- Units of Measurement: This metric is expressed as a 16-bit unsigned integer value representing the effectiveness of stream repair means.
- Measurement Point(s) with Potential Measurement Domain: It is measured at the receiving end of the RTP stream.
- Measurement Timing: This metric relies on the sequence number interval to determine measurement timing.
- Use and Applications: These metrics are applicable to any RTP applications, especially those that use loss-repair mechanisms. See Section 1 for details.
- Reporting Model: See RFC 3611.

Authors’ Addresses

Hui Zheng (Marvin)
Individual
Email: zh4ui@huawei.comoutlook.com

Roni Even
Huawei
Email: roni.even@huawei.com
Qin Wu
Huawei
Email: bill.wu@huawei.com

Rong Gu
China Mobile
Email: gurong_cmcc@outlook.com

Rachel Huang
Huawei
Email: rachel.huang@huawei.com