Internet Engineering Task Force (IETF)
Request for Comments: 8321
Category: Experimental
ISSN: 2070-1721

G. Fioccola, Ed.
A. Capello
M. Cociglio
L. Castaldelli
Telecom Italia
M. Chen
L. Zheng
Huawei Technologies
G. Mirsky
ZTE
T. Mizrahi
Marvell

January 2018

Alternate-Marking Method for Passive and Hybrid Performance Monitoring

Abstract

This document describes a method to perform packet loss, delay, and jitter measurements on live traffic. This method is based on an Alternate-Marking (coloring) technique. A report is provided in order to explain an example and show the method applicability. This technology can be applied in various situations, as detailed in this document, and could be considered Passive or Hybrid depending on the application.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8321.

Copyright Notice

Table of Contents

1. Introduction

Nowadays, most Service Providers' networks carry traffic with contents that are highly sensitive to packet loss [RFC7680], delay [RFC7679], and jitter [RFC3393].

In view of this scenario, Service Providers need methodologies and tools to monitor and measure network performance with an adequate accuracy, in order to constantly control the quality of experience perceived by their customers. On the other hand, performance monitoring provides useful information for improving network management (e.g., isolation of network problems, troubleshooting, etc.).

A lot of work related to Operations, Administration, and Maintenance (OAM), which also includes performance monitoring techniques, has been done by Standards Developing Organizations (SDOs): [RFC7276] provides a good overview of existing OAM mechanisms defined in the IETF, ITU-T, and IEEE. In the IETF, a lot of work has been done on fault detection and connectivity verification, while a minor effort has been thus far dedicated to performance monitoring. The IPPM WG has defined standard metrics to measure network performance; however, the methods developed in this WG mainly refer to focus on Active measurement techniques. More recently, the MPLS WG has defined mechanisms for measuring packet loss, one-way and two-way delay, and delay variation in MPLS networks [RFC6374], but their applicability to Passive measurements has some limitations, especially for pure connection-less networks.

The lack of adequate tools to measure packet loss with the desired accuracy drove an effort to design a new method for the performance monitoring of live traffic, which is easy to implement and deploy. The effort led to the method described in this document: basically, it is a Passive performance monitoring technique, potentially applicable to any kind of packet-based traffic, including Ethernet, IP, and MPLS, both unicast and multicast. The method addresses primarily packet loss measurement, but it can be easily extended to one-way or two-way delay and delay variation measurements as well.

The method has been explicitly designed for Passive measurements, but it can also be used with Active probes. Passive measurements are usually more easily understood by customers and provide much better accuracy, especially for packet loss measurements.

RFC 7799 [RFC7799] defines Passive and Hybrid Methods of Measurement. In particular, Passive Methods of Measurement are based solely on observations of an undisturbed and unmodified packet stream of interest; Hybrid Methods are Methods of Measurement that use a combination of Active Methods and Passive Methods.

Taking into consideration these definitions, the Alternate-Marking Method could be considered Hybrid or Passive, depending on the case. In the case where the marking method is obtained by changing existing field values of the packets (e.g., the Differentiated Services Code Point (DSCP) field), the technique is Hybrid. In the case where the marking field is dedicated, reserved, and included in the protocol specification, the Alternate-Marking technique can be considered as Passive (e.g., Synonymous Flow Label as described in [SFL-FRAMEWORK] or OAM Marking Bits as described in [PM-MM-BIER]).

The advantages of the method described in this document are:

- easy implementation: it can be implemented by using features already available on major routing platforms, as described in Section 5.1, or by applying an optimized implementation of the method for both legacy and newest technologies;

- low computational effort: the additional load on processing is negligible;

- accurate packet loss measurement: single packet loss granularity is achieved with a Passive measurement;

- potential applicability to any kind of packet-based or frame-based traffic: Ethernet, IP, MPLS, etc., and both unicast and multicast;

- robustness: the method can tolerate out-of-order packets, and it’s not based on "special" packets whose loss could have a negative impact;

- flexibility: all the timestamp formats are allowed, because they are managed out of band. The format (the Network Time Protocol (NTP) [RFC5905] or the IEEE 1588 Precision Time Protocol (PTP) [IEEE-1588]) depends on the precision you want; and

- no interoperability issues: the features required to experiment and test the method (as described in Section 5.1) are available on all current routing platforms. Both a centralized or distributed solution can be used to harvest data from the routers.

The method doesn’t raise any specific need for protocol extension, but it could be further improved by means of some extension to existing protocols. Specifically, the use of Diffserv bits for coloring the packets could not be a viable solution in some cases: a standard method to color the packets for this specific application could be beneficial.

1.1. 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.

2. Overview of the Method

In order to perform packet loss measurements on a production traffic flow, different approaches exist. The most intuitive one consists in numbering the packets so that each router that receives the flow can immediately detect a packet that is missing. This approach, though very simple in theory, is not simple to achieve: it requires the insertion of a sequence number into each packet, and the devices must be able to extract the number and check it in real time. Such a task can be difficult to implement on live traffic: if UDP is used as the transport protocol, the sequence number is not available; on the other hand, if a higher-layer sequence number (e.g., in the RTP header) is used, extracting that information from each packet and processing it in real time could overload the device.

An alternate approach is to count the number of packets sent on one end, count the number of packets received on the other end, and compare the two values. This operation is much simpler to implement, but it requires the devices performing the measurement to be in sync: in order to compare two counters, it is required that they refer exactly to the same set of packets. Since a flow is continuous and cannot be stopped when a counter has to be read, it can be difficult to determine exactly when to read the counter. A possible solution to overcome this problem is to virtually split the flow in consecutive blocks by periodically inserting a delimiter so that each counter refers exactly to the same block of packets. The delimiter could be, for example, a special packet inserted artificially into the flow. However, delimiting the flow using specific packets has some limitations. First, it requires generating additional packets within the flow and requires the equipment to be able to process those packets. In addition, the method is vulnerable to out-of-order reception of delimiting packets and, to a lesser extent, to their loss.

The method proposed in this document follows the second approach, but it doesn’t use additional packets to virtually split the flow in blocks. Instead, it "marks" the packets so that the packets belonging to the same block will have the same color, whilst consecutive blocks will have different colors. Each change of color represents a sort of auto-synchronization signal that guarantees the consistency of measurements taken by different devices along the path (see also [IP-MULTICAST-PM] and [OPSAWG-P3M], where this technique was introduced).

Figure 1 represents a very simple network and shows how the method can be used to measure packet loss on different network segments: by enabling the measurement on several interfaces along the path, it is possible to perform link monitoring, node monitoring, or end-to-end monitoring. The method is flexible enough to measure packet loss on any segment of the network and can be used to isolate the faulty element.

```
                               Traffic Flow
        ========================================================>
          +------+       +------+       +------+       +------+
      ---<>  R1  <>-----<>  R2  <>-----<>  R3  <>-----<>  R4  <>---
          +------+       +------+       +------+       +------+
          .              .      .              .       .      .
          .              .      .              .       .      .
          .              <------>              <------->      .
          .          Node Packet Loss       Link Packet Loss  .
          .                                                   .
          <--------------------------------------------------->
                           End-to-End Packet Loss
```

Figure 1: Available Measurements

3. Detailed Description of the Method

This section describes, in detail, how the method operates. A special emphasis is given to the measurement of packet loss, which represents the core application of the method, but applicability to delay and jitter measurements is also considered.

3.1. Packet Loss Measurement

The basic idea is to virtually split traffic flows into consecutive blocks: each block represents a measurable entity unambiguously recognizable by all network devices along the path. By counting the number of packets in each block and comparing the values measured by different network devices along the path, it is possible to measure packet loss occurred in any single block between any two points.

As discussed in the previous section, a simple way to create the blocks is to "color" the traffic (two colors are sufficient), so that packets belonging to different consecutive blocks will have different colors. Whenever the color changes, the previous block terminates and the new one begins. Hence, all the packets belonging to the same block will have the same color and packets of different consecutive blocks will have different colors. The number of packets in each block depends on the criterion used to create the blocks:

o if the color is switched after a fixed number of packets, then each block will contain the same number of packets (except for any losses); and

o if the color is switched according to a fixed timer, then the number of packets may be different in each block depending on the packet rate.

The following figure shows how a flow looks like when it is split in traffic blocks with colored packets.

A: packet with A coloring
B: packet with B coloring

```
            |           |           |           |           |
            |           |    Traffic Flow       |           |
    -------------------------------------------------------------------->
     BBBBBBB AAAAAAAAAAA BBBBBBBBBBB AAAAAAAAAAA BBBBBBBBBBB AAAAAAA
    -------------------------------------------------------------------->
       ...  |  Block 5  |  Block 4  |  Block 3  |  Block 2  |  Block 1
            |           |           |           |           |
```

Figure 2: Traffic Coloring

Figure 3 shows how the method can be used to measure link packet loss between two adjacent nodes.

Referring to the figure, let's assume we want to monitor the packet loss on the link between two routers: router R1 and router R2. According to the method, the traffic is colored alternatively with two different colors: A and B. Whenever the color changes, the transition generates a sort of square-wave signal, as depicted in the following figure.

```
   Color A   ----------+           +-----------+           +----------
                       |           |           |           |
   Color B             +-----------+           +-----------+
              Block n        ...       Block 3     Block 2     Block 1
            <---------> <---------> <---------> <---------> <--------->

                                Traffic Flow
            ============================================================>
   Color   ...AAAAAAAAAAA BBBBBBBBBBB AAAAAAAAAAA BBBBBBBBBBB AAAAAAA...
            ============================================================>
```

Figure 3: Computation of Link Packet Loss

Traffic coloring can be done by R1 itself if the traffic is not already colored. R1 needs two counters, C(A)R1 and C(B)R1, on its egress interface: C(A)R1 counts the packets with color A and C(B)R1 counts those with color B. As long as traffic is colored as A, only counter C(A)R1 will be incremented, while C(B)R1 is not incremented; conversely, when the traffic is colored as B, only C(B)R1 is incremented. C(A)R1 and C(B)R1 can be used as reference values to determine the packet loss from R1 to any other measurement point down the path. Router R2, similarly, will need two counters on its ingress interface, C(A)R2 and C(B)R2, to count the packets received on that interface and colored with A and B, respectively. When an A block ends, it is possible to compare C(A)R1 and C(A)R2 and calculate the packet loss within the block; similarly, when the successive B block terminates, it is possible to compare C(B)R1 with C(B)R2, and so on, for every successive block.

Likewise, by using two counters on the R2 egress interface, it is possible to count the packets sent out of the R2 interface and use them as reference values to calculate the packet loss from R2 to any measurement point down R2.

Using a fixed timer for color switching offers better control over the method: the (time) length of the blocks can be chosen large enough to simplify the collection and the comparison of measures taken by different network devices. It’s preferable to read the value of the counters not immediately after the color switch: some packets could arrive out of order and increment the counter associated with the previous block (color), so it is worth waiting for some time. A safe choice is to wait L/2 time units (where L is the duration for each block) after the color switch, to read the still counter of the previous color, so the possibility of reading a running counter instead of a still one is minimized. The drawback is that the longer the duration of the block, the less frequent the measurement can be taken.

The following table shows how the counters can be used to calculate the packet loss between R1 and R2. The first column lists the sequence of traffic blocks, while the other columns contain the counters of A-colored packets and B-colored packets for R1 and R2. In this example, we assume that the values of the counters are reset to zero whenever a block ends and its associated counter has been read: with this assumption, the table shows only relative values, which is the exact number of packets of each color within each block. If the values of the counters were not reset, the table would contain cumulative values, but the relative values could be determined simply by the difference from the value of the previous block of the same color.

The color is switched on the basis of a fixed timer (not shown in the table), so the number of packets in each block is different.

Block	C(A)R1	C(B)R1	C(A)R2	C(B)R2	Loss
1	375	0	375	0	0
2	0	388	0	388	0
3	382	0	381	0	1
4	0	377	0	374	3
...	...	...	...	...	...
2n	0	387	0	387	0
2n+1	379	0	377	0	2

Table 1: Evaluation of Counters for Packet Loss Measurements

During an A block (blocks 1, 3, and 2n+1), all the packets are A-colored; therefore, the C(A) counters are incremented to the number seen on the interface, while C(B) counters are zero. Conversely, during a B block (blocks 2, 4, and 2n), all the packets are B-colored: C(A) counters are zero, while C(B) counters are incremented.

When a block ends (because of color switching), the relative counters stop incrementing; it is possible to read them, compare the values measured on routers R1 and R2, and calculate the packet loss within that block.

For example, looking at the table above, during the first block (A-colored), C(A)R1 and C(A)R2 have the same value (375), which corresponds to the exact number of packets of the first block (no loss). Also, during the second block (B-colored), R1 and R2 counters have the same value (388), which corresponds to the number of packets of the second block (no loss). During the third and fourth blocks, R1 and R2 counters are different, meaning that some packets have been lost: in the example, one single packet (382-381) was lost during block three, and three packets (377-374) were lost during block four.

The method applied to R1 and R2 can be extended to any other router and applied to more complex networks, as far as the measurement is enabled on the path followed by the traffic flow(s) being observed.

It’s worth mentioning two different strategies that can be used when implementing the method:

- flow-based: the flow-based strategy is used when only a limited number of traffic flows need to be monitored. According to this strategy, only a subset of the flows is colored. Counters for packet loss measurements can be instantiated for each single flow, or for the set as a whole, depending on the desired granularity. A relevant problem with this approach is the necessity to know in advance the path followed by flows that are subject to measurement. Path rerouting and traffic load-balancing increase the issue complexity, especially for unicast traffic. The problem is easier to solve for multicast traffic, where load-balancing is seldom used and static joins are frequently used to force traffic forwarding and replication.

- link-based: measurements are performed on all the traffic on a link-by-link basis. The link could be a physical link or a logical link. Counters could be instantiated for the traffic as a whole or for each traffic class (in case it is desired to monitor each class separately), but in the second case, a couple of counters are needed for each class.

As mentioned, the flow-based measurement requires the identification of the flow to be monitored and the discovery of the path followed by the selected flow. It is possible to monitor a single flow or multiple flows grouped together, but in this case, measurement is consistent only if all the flows in the group follow the same path. Moreover, if a measurement is performed by grouping many flows, it is not possible to determine exactly which flow was affected by packet loss. In order to have measures per single flow, it is necessary to configure counters for each specific flow. Once the flow(s) to be monitored has been identified, it is necessary to configure the monitoring on the proper nodes. Configuring the monitoring means configuring the rule to intercept the traffic and configuring the counters to count the packets. To have just an end-to-end monitoring, it is sufficient to enable the monitoring on the first- and last-hop routers of the path: the mechanism is completely transparent to intermediate nodes and independent from the path followed by traffic flows. On the contrary, to monitor the flow on a hop-by-hop basis along its whole path, it is necessary to enable the monitoring on every node from the source to the destination. In case the exact path followed by the flow is not known a priori (i.e., the flow has multiple paths to reach the destination), it is necessary to enable the monitoring system on every path: counters on interfaces traversed by the flow will report packet count, whereas counters on other interfaces will be null.

3.1.1. Coloring the Packets

The coloring operation is fundamental in order to create packet blocks. This implies choosing where to activate the coloring and how to color the packets.

In case of flow-based measurements, the flow to monitor can be defined by a set of selection rules (e.g., header fields) used to match a subset of the packets; in this way, it is possible to control the number of involved nodes, the path followed by the packets, and the size of the flows. It is possible, in general, to have multiple coloring nodes or a single coloring node that is easier to manage and doesn’t raise any risk of conflict. Coloring in multiple nodes can be done, and the requirement is that the coloring must change periodically between the nodes according to the timing considerations in Section 3.2; so every node that is designated as a measurement point along the path should be able to identify unambiguously the colored packets. Furthermore, [MULTIPOINT-ALT-MM] generalizes the coloring for multipoint-to-multipoint flow. In addition, it can be advantageous to color the flow as close as possible to the source because it allows an end-to-end measure if a measurement point is enabled on the last-hop router as well.

For link-based measurements, all traffic needs to be colored when transmitted on the link. If the traffic had already been colored, then it has to be re-colored because the color must be consistent on the link. This means that each hop along the path must (re-)color the traffic; the color is not required to be consistent along different links.

Traffic coloring can be implemented by setting a specific bit in the packet header and changing the value of that bit periodically. How to choose the marking field depends on the application and is out of scope here. However, some applications are reported in Section 5.

3.1.2. Counting the Packets

For flow-based measurements, assuming that the coloring of the packets is performed only by the source nodes, the nodes between source and destination (included) have to count the colored packets that they receive and forward: this operation can be enabled on every router along the path or only on a subset, depending on which network segment is being monitored (a single link, a particular metro area, the backbone, or the whole path). Since the color switches periodically between two values, two counters (one for each value) are needed: one counter for packets with color A and one counter for packets with color B. For each flow (or group of flows) being monitored and for every interface where the monitoring is Active, a couple of counters are needed. For example, in order to separately monitor three flows on a router with four interfaces involved, 24 counters are needed (two counters for each of the three flows on each of the four interfaces). Furthermore, [MULTIPOINT-ALT-MM] generalizes the counting for multipoint-to-multipoint flow.

In case of link-based measurements, the behavior is similar except that coloring and counting operations are performed on a link-by-link basis at each endpoint of the link.

Another important aspect to take into consideration is when to read the counters: in order to count the exact number of packets of a block, the routers must perform this operation when that block has ended; in other words, the counter for color A must be read when the current block has color B, in order to be sure that the value of the counter is stable. This task can be accomplished in two ways. The general approach suggests reading the counters periodically, many times during a block duration, and comparing these successive readings: when the counter stops incrementing, it means that the current block has ended, and its value can be elaborated safely. Alternatively, if the coloring operation is performed on the basis of a fixed timer, it is possible to configure the reading of the counters according to that timer: for example, reading the counter for color A every period in the middle of the subsequent block with color B is a safe choice. A sufficient margin should be considered between the end of a block and the reading of the counter, in order to take into account any out-of-order packets.

3.1.3. Collecting Data and Calculating Packet Loss

The nodes enabled to perform performance monitoring collect the value of the counters, but they are not able to directly use this information to measure packet loss, because they only have their own samples. For this reason, an external Network Management System (NMS) can be used to collect and elaborate data and to perform packet loss calculation. The NMS compares the values of counters from different nodes and can calculate if some packets were lost (even a single packet) and where those packets were lost.

The value of the counters needs to be transmitted to the NMS as soon as it has been read. This can be accomplished by using SNMP or FTP and can be done in Push Mode or Polling Mode. In the first case, each router periodically sends the information to the NMS; in the latter case, it is the NMS that periodically polls routers to collect information. In any case, the NMS has to collect all the relevant values from all the routers within one cycle of the timer.

It would also be possible to use a protocol to exchange values of counters between the two endpoints in order to let them perform the packet loss calculation for each traffic direction.

A possible approach for the performance measurement (PM) architecture is explained in [COLORING], while [IP-FLOW-REPORT] introduces new information elements of IP Flow Information Export (IPFIX) [RFC7011].

3.2. Timing Aspects

This document introduces two color-switching methods: one is based on a fixed number of packets, and the other is based on a fixed timer. But the method based on a fixed timer is preferable because it is more deterministic, and it will be considered in the rest of the document.

In general, clocks in network devices are not accurate and for this reason, there is a clock error between the measurement points R1 and R2. But, to implement the methodology, they must be synchronized to the same clock reference with an accuracy of +/- L/2 time units, where L is the fixed time duration of the block. So each colored packet can be assigned to the right batch by each router. This is because the minimum time distance between two packets of the same color but that belong to different batches is L time units.

In practice, in addition to clock errors, the delay between measurement points also affects the implementation of the methodology because each packet can be delayed differently, and this can produce out of order at batch boundaries. This means that, without considering clock error, we wait L/2 after color switching to be sure to take a still counter.

In summary, we need to take into account two contributions: clock error between network devices and the interval we need to wait to avoid packets being out of order because of network delay.

The following figure explains both issues.

```
   ...BBBBBBBBB | AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA | BBBBBBBBB...
                |<====================================>|
                |                   L                    |
   ...=========>|<==================><==================>|<==========...
                |       L/2                   L/2        |
                |<===>|                            |<===>|
                   d  |                            |   d
                      |<==========================>|
                       available counting interval
```

Figure 4: Timing Aspects

It is assumed that all network devices are synchronized to a common reference time with an accuracy of +/- A/2. Thus, the difference between the clock values of any two network devices is bounded by A.

The guard band d is given by:

d = A + D_max - D_min,

where A is the clock accuracy, D_max is an upper bound on the network delay between the network devices, and D_min is a lower bound on the delay.

The available counting interval is L - 2d that must be > 0.

The condition that must be satisfied and is a requirement on the synchronization accuracy is:

d < L/2.

3.3. One-Way Delay Measurement

The same principle used to measure packet loss can be applied also to one-way delay measurement. There are three alternatives, as described hereinafter.

Note that, for all the one-way delay alternatives described in the next sections, by summing the one-way delays of the two directions of a path, it is always possible to measure the two-way delay (round-trip "virtual" delay).

3.3.1. Single-Marking Methodology

The alternation of colors can be used as a time reference to calculate the delay. Whenever the color changes (which means that a new block has started), a network device can store the timestamp of the first packet of the new block; that timestamp can be compared with the timestamp of the same packet on a second router to compute packet delay. When looking at Figure 2, R1 stores the timestamp TS(A1)R1 when it sends the first packet of block 1 (A-colored), the timestamp TS(B2)R1 when it sends the first packet of block 2 (B-colored), and so on for every other block. R2 performs the same operation on the receiving side, recording TS(A1)R2, TS(B2)R2, and so on. Since the timestamps refer to specific packets (the first packet of each block), we are sure that timestamps compared to compute delay refer to the same packets. By comparing TS(A1)R1 with TS(A1)R2 (and similarly TS(B2)R1 with TS(B2)R2, and so on), it is possible to measure the delay between R1 and R2. In order to have more measurements, it is possible to take and store more timestamps, referring to other packets within each block.

In order to coherently compare timestamps collected on different routers, the clocks on the network nodes must be in sync. Furthermore, a measurement is valid only if no packet loss occurs and if packet misordering can be avoided; otherwise, the first packet of a block on R1 could be different from the first packet of the same block on R2 (for instance, if that packet is lost between R1 and R2 or it arrives after the next one).

The following table shows how timestamps can be used to calculate the delay between R1 and R2. The first column lists the sequence of blocks, while other columns contain the timestamp referring to the first packet of each block on R1 and R2. The delay is computed as a difference between timestamps. For the sake of simplicity, all the values are expressed in milliseconds.

Block	TS(A)R1	TS(B)R1	TS(A)R2	TS(B)R2	Delay R1-R2
1	12.483	-	15.591	-	3.108
2	-	6.263	-	9.288	3.025
3	27.556	-	30.512	-	2.956
	-	18.113	-	21.269	3.156
...	...	...	...	...	...
2n	77.463	-	80.501	-	3.038
2n+1	-	24.333	-	27.433	3.100

Table 2: Evaluation of Timestamps for Delay Measurements

The first row shows timestamps taken on R1 and R2, respectively, and refers to the first packet of block 1 (which is A-colored). Delay can be computed as a difference between the timestamp on R2 and the timestamp on R1. Similarly, the second row shows timestamps (in milliseconds) taken on R1 and R2 and refers to the first packet of block 2 (which is B-colored). By comparing timestamps taken on different nodes in the network and referring to the same packets (identified using the alternation of colors), it is possible to measure delay on different network segments.

For the sake of simplicity, in the above example, a single measurement is provided within a block, taking into account only the first packet of each block. The number of measurements can be easily increased by considering multiple packets in the block: for instance, a timestamp could be taken every N packets, thus generating multiple delay measurements. Taking this to the limit, in principle, the delay could be measured for each packet by taking and comparing the corresponding timestamps (possible but impractical from an implementation point of view).

3.3.1.1. Mean Delay

As mentioned before, the method previously exposed for measuring the delay is sensitive to out-of-order reception of packets. In order to overcome this problem, a different approach has been considered: it is based on the concept of mean delay. The mean delay is calculated by considering the average arrival time of the packets within a single block. The network device locally stores a timestamp for each packet received within a single block: summing all the timestamps and dividing by the total number of packets received, the average arrival time for that block of packets can be calculated. By subtracting the average arrival times of two adjacent devices, it is possible to calculate the mean delay between those nodes. When computing the mean delay, the measurement error could be augmented by accumulating the measurement error of a lot of packets. This method is robust to out-of-order packets and also to packet loss (only a small error is introduced). Moreover, it greatly reduces the number of timestamps (only one per block for each network device) that have to be collected by the management system. On the other hand, it only gives one measure for the duration of the block (for instance, 5 minutes), and it doesn't give the minimum, maximum, and median delay values [RFC6703]. This limitation could be overcome by reducing the duration of the block (for instance, from 5 minutes to a few seconds), which implicates a highly optimized implementation of the method.

3.3.2. Double-Marking Methodology

The Single-Marking methodology for one-way delay measurement is sensitive to out-of-order reception of packets. The first approach to overcome this problem has been described before and is based on the concept of mean delay. But the limitation of mean delay is that it doesn't give information about the delay value's distribution for the duration of the block. Additionally, it may be useful to have not only the mean delay but also the minimum, maximum, and median delay values and, in wider terms, to know more about the statistic distribution of delay values. So, in order to have more information about the delay and to overcome out-of-order issues, a different approach can be introduced; it is based on a Double-Marking methodology.

Basically, the idea is to use the first marking to create the alternate flow and, within this colored flow, a second marking to select the packets for measuring delay/jitter. The first marking is needed for packet loss and mean delay measurement. The second marking creates a new set of marked packets that are fully identified over the network, so that a network device can store the timestamps of these packets; these timestamps can be compared with the timestamps of the same packets on a second router to compute packet delay values for each packet. The number of measurements can be easily increased by changing the frequency of the second marking. But the frequency of the second marking must not be too high in order to avoid out-of-order issues. Between packets with the second marking, there should be a security time gap (e.g., this gap could be, at the minimum, the mean network delay calculated with the previous methodology) to avoid out-of-order issues and also to have a number of measurement packets that are rate independent. If a second-marking packet is lost, the delay measurement for the considered block is corrupted and should be discarded.

Mean delay is calculated on all the packets of a sample and is a simple computation to be performed for a Single-Marking Method. In some cases, the mean delay measure is not sufficient to characterize the sample, and more statistics of delay extent data are needed, e.g., percentiles, variance, and median delay values. The conventional range (maximum-minimum) should be avoided for several reasons, including stability of the maximum delay due to the influence by outliers. RFC 5481 [RFC5481], Section 6.5 highlights how the 99.9th percentile of delay and delay variation is more helpful to performance planners. To overcome this drawback, the idea is to couple the mean delay measure for the entire batch with a Double-Marking Method, where a subset of batch packets is selected for extensive delay calculation by using a second marking. In this way, it is possible to perform a detailed analysis on these double-marked packets. Please note that there are classic algorithms for median and variance calculation, but they are out of the scope of this document. The comparison between the mean delay for the entire batch and the mean delay on these double-marked packets gives useful information since it is possible to understand if the Double-Marking measurements are actually representative of the delay trends.

3.4. Delay Variation Measurement

Similar to one-way delay measurement (both for Single Marking and Double Marking), the method can also be used to measure the inter-arrival jitter. We refer to the definition in RFC 3393 [RFC3393]. The alternation of colors, for a Single-Marking Method, can be used as a time reference to measure delay variations. In case of Double Marking, the time reference is given by the second-marked packets. Considering the example depicted in Figure 2, R1 stores the timestamp TS(A)R1 whenever it sends the first packet of a block, and R2 stores the timestamp TS(B)R2 whenever it receives the first packet of a block. The inter-arrival jitter can be easily derived from one-way delay measurement, by evaluating the delay variation of consecutive samples.

The concept of mean delay can also be applied to delay variation, by evaluating the average variation of the interval between consecutive packets of the flow from R1 to R2.

4. Considerations

This section highlights some considerations about the methodology.

4.1. Synchronization

The Alternate-Marking technique does not require a strong synchronization, especially for packet loss and two-way delay measurement. Only one-way delay measurement requires network devices to have synchronized clocks.

Color switching is the reference for all the network devices, and the only requirement to be achieved is that all network devices have to recognize the right batch along the path.

If the length of the measurement period is L time units, then all network devices must be synchronized to the same clock reference with an accuracy of +/- L/2 time units (without considering network delay). This level of accuracy guarantees that all network devices consistently match the color bit to the correct block. For example, if the color is toggled every second (L = 1 second), then clocks must be synchronized with an accuracy of +/- 0.5 second to a common time reference.

This synchronization requirement can be satisfied even with a relatively inaccurate synchronization method. This is true for packet loss and two-way delay measurement, but not for one-way delay measurement, where clock synchronization must be accurate.

Therefore, a system that uses only packet loss and two-way delay measurement does not require synchronization. This is because the value of the clocks of network devices does not affect the computation of the two-way delay measurement.

4.2. Data Correlation

Data correlation is the mechanism to compare counters and timestamps for packet loss, delay, and delay variation calculation. It could be performed in several ways depending on the Alternate-Marking application and use case. Some possibilities are to:

- use a centralized solution using NMS to correlate data; and

- define a protocol-based distributed solution by introducing a new protocol or by extending the existing protocols (e.g., see RFC 6374 [RFC6374] or the Two-Way Active Measurement Protocol (TWAMP) as defined in RFC 5357 [RFC5357] or the One-Way Active Measurement Protocol (OWAMP) as defined in RFC 4656 [RFC4656]) in order to communicate the counters and timestamps between nodes.

In the following paragraphs, an example data correlation mechanism is explained and could be used independently of the adopted solutions.

When data is collected on the upstream and downstream nodes, e.g., packet counts for packet loss measurement or timestamps for packet delay measurement, and is periodically reported to or pulled by other nodes or an NMS, a certain data correlation mechanism SHOULD be in use to help the nodes or NMS tell whether any two or more packet counts are related to the same block of markers or if any two timestamps are related to the same marked packet.

The Alternate-Marking Method described in this document literally splits the packets of the measured flow into different measurement blocks; in addition, a Block Number (BN) could be assigned to each such measurement block. The BN is generated each time a node reads the data (packet counts or timestamps) and is associated with each packet count and timestamp reported to or pulled by other nodes or NMSs. The value of a BN could be calculated as the modulo of the local time (when the data are read) and the interval of the marking time period.

When the nodes or NMS see, for example, the same BNs associated with two packet counts from an upstream and a downstream node, respectively, it considers that these two packet counts correspond to the same block, i.e., these two packet counts belong to the same block of markers from the upstream and downstream nodes. The assumption of this BN mechanism is that the measurement nodes are time synchronized. This requires the measurement nodes to have a certain time synchronization capability (e.g., the Network Time Protocol (NTP) [RFC5905] or the IEEE 1588 Precision Time Protocol (PTP) [IEEE-1588]). Synchronization aspects are further discussed in Section 4.1.

4.3. Packet Reordering

Due to ECMP, packet reordering is very common in an IP network. The accuracy of a marking-based PM, especially packet loss measurement, may be affected by packet reordering. Take a look at the following example:

```
Block   :    1    |    2    |    3    |    4    |    5    |...
--------|---------|---------|---------|---------|---------|---
Node R1 : AAAAAAA | BBBBBBB | AAAAAAA | BBBBBBB | AAAAAAA |...
Node R2 : AAAAABB | AABBBBA | AAABAAA | BBBBBBA | ABAAABA |...
```

Figure 5: Packet Reordering

In Figure 5, the packet stream for Node R1 isn’t being reordered and can be safely assigned to interval blocks, but the packet stream for Node R2 is being reordered; so, looking at the packet with the marker of "B" in block 3, there is no safe way to tell whether the packet belongs to block 2 or block 4.

In general, there is the need to assign packets with the marker of "B" or "A" to the right interval blocks. Most of the packet reordering occurs at the edge of adjacent blocks, and they are easy to handle if the interval of each block is sufficiently large. Then, it can be assumed that the packets with different markers belong to the block that they are closer to. If the interval is small, it is difficult and sometimes impossible to determine to which block a packet belongs.

To choose a proper interval is important, and how to choose a proper interval is out of the scope of this document. But an implementation SHOULD provide a way to configure the interval and allow a certain degree of packet reordering.

5. Applications, Implementation, and Deployment

The methodology described in the previous sections can be applied in various situations. Basically, the Alternate-Marking technique could be used in many cases for performance measurement. The only requirement is to select and mark the flow to be monitored; in this way, packets are batched by the sender, and each batch is alternately marked such that it can be easily recognized by the receiver.

Some recent Alternate-Marking Method applications are listed below:

- IP Flow Performance Measurement (IPFPM): this application of the marking method is described in [COLORING]. As an example, in this document, the last reserved bit of the Flag field of the IPv4 header is proposed to be used for marking, while a solution for IPv6 could be to leverage the IPv6 extension header for marking.

- OAM Passive Performance Measurement: In [RFC8296], two OAM bits from the Bit Index Explicit Replication (BIER) header are reserved for the Passive performance measurement marking method. [PM-MM-BIER] details the measurement for multicast service over the BIER domain. In addition, the Alternate-Marking Method could also be used in a Service Function Chaining (SFC) domain. Lastly, the application of the marking method to Network Virtualization over Layer 3 (NVO3) protocols is considered by [NVO3-ENCAPS].

- MPLS Performance Measurement: RFC 6374 [RFC6374] uses the Loss Measurement (LM) packet as the packet accounting demarcation point. Unfortunately, this gives rise to a number of problems that may lead to significant packet accounting errors in certain situations. [MPLS-FLOW] discusses the desired capabilities for

MPLS flow identification in order to perform a better in-band performance monitoring of user data packets. A method of accomplishing identification is Synonymous Flow Labels (SFLs) introduced in [SFL-FRAMEWORK], while [SYN-FLOW-LABELS] describes performance measurements in RFC 6374 with SFL.

- Active Performance Measurement: [ALT-MM-AMP] describes how to extend the existing Active Measurement Protocol, in order to implement the Alternate-Marking methodology. [ALT-MM-SLA] describes an extension to the Cisco SLA Protocol Measurement-Type UDP-Measurement.

An example of implementation and deployment is explained in the next section, just to clarify how the method can work.

5.1. Report on the Operational Experiment

The method described in this document, also called Packet Network Performance Monitoring (PNPM), has been invented and engineered in Telecom Italia.

It is important to highlight that the general description of the methodology in this document is a consequence of the operational experiment. The fundamental elements of the technique have been tested, and the lessons learned from the operational experiment inspired the formalization of the Alternate-Marking Method as detailed in the previous sections.

The methodology has been used experimentally in Telecom Italia’s network and is applied to multicast IPTV channels or other specific traffic flows with high QoS requirements (i.e., Mobile Backhauling traffic realized with a VPN MPLS).

This technology has been employed by leveraging functions and tools available on IP routers, and it’s currently being used to monitor packet loss in some portions of Telecom Italia’s network. The application of this method for delay measurement has also been evaluated in Telecom Italia’s labs.

This section describes how the experiment has been executed, particularly, how the features currently available on existing routing platforms can be used to apply the method, in order to give an example of implementation and deployment.

The operational test, described herein, uses the flow-based strategy, as defined in Section 3. Instead, the link-based strategy could be applied to a physical link or a logical link (e.g., an Ethernet VLAN or an MPLS Pseudowire (PW)).

The implementation of the method leverages the available router functions, since the experiment has been done by a Service Provider (as Telecom Italia is) on its own network. So, with current router implementations, only QoS-related fields and features offer the required flexibility to set bits in the packet header. In case a Service Provider only uses the three most-significant bits of the DSCP field (corresponding to IP Precedence) for QoS classification and queuing, it is possible to use the two least-significant bits of the DSCP field (bit 0 and bit 1) to implement the method without affecting QoS policies. That is the approach used for the experiment. One of the two bits (bit 0) could be used to identify flows subject to traffic monitoring (set to 1 if the flow is under monitoring, otherwise, it is set to 0), while the second (bit 1) can be used for coloring the traffic (switching between values 0 and 1, corresponding to colors A and B) and creating the blocks.

The experiment considers a flow as all the packets sharing the same source IP address or the same destination IP address, depending on the direction. In practice, once the flow has been defined, traffic coloring using the DSCP field can be implemented by configuring an access-list on the router output interface. The access-list intercepts the flow(s) to be monitored and applies a policy to them that sets the DSCP field accordingly. Since traffic coloring has to be switched between the two values over time, the policy needs to be modified periodically. An automatic script is used to perform this task on the basis of a fixed timer. The automatic script is loaded on board of the router and automatizes the basic operations that are needed to realize the methodology.

After the traffic is colored using the DSCP field, all the routers on the path can perform the counting. For this purpose, an access-list that matches specific DSCP values can be used to count the packets of the flow(s) being monitored. The same access-list can be installed on all the routers of the path. In addition, network flow monitoring, such as provided by IPFIX [RFC7011], can be used to recognize timestamps of the first/last packet of a batch in order to enable one of the alternatives to measure the delay as detailed in Section 3.3.

In Telecom Italia’s experiment, the timer is set to 5 minutes, so the sequence of actions of the script is also executed every 5 minutes. This value has shown to be a good compromise between measurement frequency and stability of the measurement (i.e., the possibility of collecting all the measures referring to the same block).

For this experiment, both counters and any other data are collected by using the automatic script that sends these out to an NMS. The NMS is responsible for packet loss calculation, performed by comparing the values of counters from the routers along the flow path(s). A 5-minute timer for color switching is a safe choice for reading the counters and is also coherent with the reporting window of the NMS.

Note that the use of the DSCP field for marking implies that the method in this case works reliably only within a single management and operation domain.

Lastly, the Telecom Italia experiment scales up to 1000 flows monitored together on a single router, while an implementation on dedicated hardware scales more, but it was tested only in labs for now.

5.1.1. Metric Transparency

Since a Service Provider application is described here, the method can be applied to end-to-end services supplied to customers. So it is important to highlight that the method MUST be transparent outside the Service Provider domain.

In Telecom Italia’s implementation, the source node colors the packets with a policy that is modified periodically via an automatic script in order to alternate the DSCP field of the packets. The nodes between source and destination (included) have to use an access-list to count the colored packets that they receive and forward.

Moreover, the destination node has an important role: the colored packets are intercepted and a policy restores and sets the DSCP field of all the packets to the initial value. In this way, the metric is transparent because outside the section of the network under monitoring, the traffic flow is unchanged.

In such a case, thanks to this restoring technique, network elements outside the Alternate-Marking monitoring domain (e.g., the two Provider Edge nodes of the Mobile Backhauling VPN MPLS) are totally unaware that packets were marked. So this restoring technique makes Alternate Marking completely transparent outside its monitoring domain.

6. Hybrid Measurement

The method has been explicitly designed for Passive measurements, but it can also be used with Active measurements. In order to have both end-to-end measurements and intermediate measurements (Hybrid measurements), two endpoints can exchange artificial traffic flows and apply Alternate Marking over these flows. In the intermediate points, artificial traffic is managed in the same way as real traffic and measured as specified before. So the application of the marking method can also simplify the Active measurement, as explained in [ALT-MM-AMP].

7. Compliance with Guidelines from RFC 6390

RFC 6390 [RFC6390] defines a framework and a process for developing Performance Metrics for protocols above and below the IP layer (such as IP-based applications that operate over reliable or datagram transport protocols).

This document doesn’t aim to propose a new Performance Metric but rather a new Method of Measurement for a few Performance Metrics that have already been standardized. Nevertheless, it’s worth applying guidelines from [RFC6390] to the present document, in order to provide a more complete and coherent description of the proposed method. We used a combination of the Performance Metric Definition template defined in Section 5.4 of [RFC6390] and the Dependencies laid out in Section 5.5 of that document.

- Metric Name / Metric Description: as already stated, this document doesn’t propose any new Performance Metrics. On the contrary, it describes a novel method for measuring packet loss [RFC7680]. The same concept, with small differences, can also be used to measure delay [RFC7679] and jitter [RFC3393]. The document mainly describes the applicability to packet loss measurement.

- Method of Measurement or Calculation: according to the method described in the previous sections, the number of packets lost is calculated by subtracting the value of the counter on the source node from the value of the counter on the destination node. Both counters must refer to the same color. The calculation is performed when the value of the counters is in a steady state. The steady state is an intrinsic characteristic of the marking method counters because the alternation of color makes the counters associated with each color still one at a time for the duration of a marking period.

- Units of Measurement: the method calculates and reports the exact number of packets sent by the source node and not received by the destination node.

- Measurement Point(s) with Potential Measurement Domain: the measurement can be performed between adjacent nodes, on a per-link basis, or along a multi-hop path, provided that the traffic under measurement follows that path. In case of a multi-hop path, the measurements can be performed both end-to-end and hop-by-hop.

- Measurement Timing: the method has a constraint on the frequency of measurements. This is detailed in Section 3.2, where it is specified that the marking period and the guard band interval are strictly related each other to avoid out-of-order issues. That is because, in order to perform a measurement, the counter must be in a steady state, and this happens when the traffic is being colored with the alternate color. As an example, in the experiment of the method, the time interval is set to 5 minutes, while other optimized implementations can also use a marking period of a few seconds.

- Implementation: the experiment of the method uses two encodings of the DSCP field to color the packets; this enables the use of policy configurations on the router to color the packets and accordingly configure the counter for each color. The path followed by traffic being measured should be known in advance in order to configure the counters along the path and be able to compare the correct values.

- Verification: both in the lab and in the operational network, the methodology has been tested and experimented for packet loss and delay measurements by using traffic generators together with precision test instruments and network emulators.

- Use and Applications: the method can be used to measure packet loss with high precision on live traffic; moreover, by combining end-to-end and per-link measurements, the method is useful to pinpoint the single link that is experiencing loss events.

- Reporting Model: the value of the counters has to be sent to a centralized management system that performs the calculations; such samples must contain a reference to the time interval they refer to, so that the management system can perform the correct correlation; the samples have to be sent while the corresponding counter is in a steady state (within a time interval); otherwise, the value of the sample should be stored locally.

- Dependencies: the values of the counters have to be correlated to the time interval they refer to; moreover, because the experiment of the method is based on DSCP values, there are significant dependencies on the usage of the DSCP field: it must be possible to rely on unused DSCP values without affecting QoS-related configuration and behavior; moreover, the intermediate nodes must not change the value of the DSCP field not to alter the measurement.

- Organization of Results: the Method of Measurement produces singletons.

- Parameters: currently, the main parameter of the method is the time interval used to alternate the colors and read the counters.

8. IANA Considerations

This document has no IANA actions.

9. Security Considerations

This document specifies a method to perform measurements in the context of a Service Provider’s network and has not been developed to conduct Internet measurements, so it does not directly affect Internet security nor applications that run on the Internet. However, implementation of this method must be mindful of security and privacy concerns.

There are two types of security concerns: potential harm caused by the measurements and potential harm to the measurements.

- Harm caused by the measurement: the measurements described in this document are Passive, so there are no new packets injected into the network causing potential harm to the network itself and to data traffic. Nevertheless, the method implies modifications on the fly to a header or encapsulation of the data packets: this must be performed in a way that doesn’t alter the quality of service experienced by packets subject to measurements and that preserves stability and performance of routers doing the measurements. One of the main security threats in OAM protocols is network reconnaissance; an attacker can gather information about the network performance by passively eavesdropping on OAM messages. The advantage of the methods described in this document is that the marking bits are the only information that is exchanged between the network devices. Therefore, Passive eavesdropping on data-plane traffic does not allow attackers to gain information about the network performance.

- Harm to the Measurement: the measurements could be harmed by routers altering the marking of the packets or by an attacker injecting artificial traffic. Authentication techniques, such as digital signatures, may be used where appropriate to guard against injected traffic attacks. Since the measurement itself may be affected by routers (or other network devices) along the path of IP packets intentionally altering the value of marking bits of packets, as mentioned above, the mechanism specified in this document can be applied just in the context of a controlled domain; thus, the routers (or other network devices) are locally administered and this type of attack can be avoided. In addition, an attacker can’t gain information about network performance from a single monitoring point; it must use synchronized monitoring points at multiple points on the path, because they have to do the same kind of measurement and aggregation that Service Providers using Alternate Marking must do.

The privacy concerns of network measurement are limited because the method only relies on information contained in the header or encapsulation without any release of user data. Although information in the header or encapsulation is metadata that can be used to compromise the privacy of users, the limited marking technique in this document seems unlikely to substantially increase the existing privacy risks from header or encapsulation metadata. It might be theoretically possible to modulate the marking to serve as a covert channel, but it would have a very low data rate if it is to avoid adversely affecting the measurement systems that monitor the marking.

Delay attacks are another potential threat in the context of this document. Delay measurement is performed using a specific packet in each block, marked by a dedicated color bit. Therefore, a man-in-the-middle attacker can selectively induce synthetic delay only to delay-colored packets, causing systematic error in the delay measurements. As discussed in previous sections, the methods described in this document rely on an underlying time synchronization protocol. Thus, by attacking the time protocol, an attacker can potentially compromise the integrity of the measurement. A detailed discussion about the threats against time protocols and how to mitigate them is presented in RFC 7384 [RFC7384].

10. References

10.1. Normative References

[IEEE-1588]
IEEE, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE Std 1588-2008.

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

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, DOI 10.17487/RFC3393, November 2002, <https://www.rfc-editor.org/info/rfc3393>.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <https://www.rfc-editor.org/info/rfc5905>.

[RFC7679] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Delay Metric for IP Performance Metrics (IPPM)", STD 81, RFC 7679, DOI 10.17487/RFC7679, January 2016, <https://www.rfc-editor.org/info/rfc7679>.

[RFC7680] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Loss Metric for IP Performance Metrics (IPPM)", STD 82, RFC 7680, DOI 10.17487/RFC7680, January 2016, <https://www.rfc-editor.org/info/rfc7680>.

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

10.2. Informative References

[ALT-MM-AMP]
Fioccola, G., Clemm, A., Bryant, S., Cociglio, M., Chandramouli, M., and A. Capello, "Alternate Marking Extension to Active Measurement Protocol", Work in Progress, draft-fioccola-ippm-alt-mark-active-01, March 2017.

[ALT-MM-SLA]
Fioccola, G., Clemm, A., Cociglio, M., Chandramouli, M., and A. Capello, "Alternate Marking Extension to Cisco SLA Protocol RFC6812", Work in Progress, draft-fioccola-ippm-rfc6812-alt-mark-ext-01, March 2016.

[COLORING] Chen, M., Zheng, L., Mirsky, G., Fioccola, G., and T. Mizrahi, "IP Flow Performance Measurement Framework", Work in Progress, draft-chen-ippm-coloring-based-ipfpm-framework-06, March 2016.

[IP-FLOW-REPORT]
Chen, M., Zheng, L., and G. Mirsky, "IP Flow Performance Measurement Report", Work in Progress, draft-chen-ippm-ipfpm-report-01, April 2016.

[IP-MULTICAST-PM]
Cociglio, M., Capello, A., Bonda, A., and L. Castaldelli, "A method for IP multicast performance monitoring", Work in Progress, draft-cociglio-mboned-multicast-pm-01, October 2010.

[MPLS-FLOW]
Bryant, S., Pignataro, C., Chen, M., Li, Z., and G. Mirsky, "MPLS Flow Identification Considerations", Work in Progress, draft-ietf-mpls-flow-ident-06, December 2017.

[MULTIPOINT-ALT-MM]
Fioccola, G., Cociglio, M., Sapio, A., and R. Sisto, "Multipoint Alternate Marking method for passive and hybrid performance monitoring", Work in Progress, draft-fioccola-ippm-multipoint-alt-mark-01, October 2017.

[NVO3-ENCAPS]
Boutros, S., Ganga, I., Garg, P., Manur, R., Mizrahi, T., Mozes, D., Nordmark, E., Smith, M., Aldrin, S., and I. Bagdonas, "NVO3 Encapsulation Considerations", Work in Progress, draft-ietf-nvo3-encap-01, October 2017.

[OPSAWG-P3M]
Capello, A., Cociglio, M., Castaldelli, L., and A. Bonda, "A packet based method for passive performance monitoring", Work in Progress, draft-tempia-opsawg-p3m-04, February 2014.

[PM-MM-BIER]
Mirsky, G., Zheng, L., Chen, M., and G. Fioccola, "Performance Measurement (PM) with Marking Method in Bit Index Explicit Replication (BIER) Layer", Work in Progress, draft-ietf-bier-pmmm-oam-03, October 2017.

[RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <https://www.rfc-editor.org/info/rfc4656>.

[RFC5357] Hedayat, K., Krzanowski, R., Morton, A., Yum, K., and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)", RFC 5357, DOI 10.17487/RFC5357, October 2008, <https://www.rfc-editor.org/info/rfc5357>.

[RFC5481] Morton, A. and B. Claise, "Packet Delay Variation Applicability Statement", RFC 5481, DOI 10.17487/RFC5481, March 2009, <https://www.rfc-editor.org/info/rfc5481>.

[RFC6374] Frost, D. and S. Bryant, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, DOI 10.17487/RFC6374, September 2011, <https://www.rfc-editor.org/info/rfc6374>.

[RFC6390] Clark, A. and B. Claise, "Guidelines for Considering New Performance Metric Development", BCP 170, RFC 6390, DOI 10.17487/RFC6390, October 2011, <https://www.rfc-editor.org/info/rfc6390>.

[RFC6703] Morton, A., Ramachandran, G., and G. Maguluri, "Reporting IP Network Performance Metrics: Different Points of View", RFC 6703, DOI 10.17487/RFC6703, August 2012, <https://www.rfc-editor.org/info/rfc6703>.

[RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <https://www.rfc-editor.org/info/rfc7011>.

[RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", RFC 7276, DOI 10.17487/RFC7276, June 2014, <https://www.rfc-editor.org/info/rfc7276>.

[RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/RFC7384, October 2014, <https://www.rfc-editor.org/info/rfc7384>.

[RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <https://www.rfc-editor.org/info/rfc7799>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

[SFL-FRAMEWORK]
Bryant, S., Chen, M., Li, Z., Swallow, G., Sivabalan, S., and G. Mirsky, "Synonymous Flow Label Framework", Work in Progress, draft-ietf-mpls-sfl-framework-00, August 2017.

[SYN-FLOW-LABELS]
Bryant, S., Chen, M., Li, Z., Swallow, G., Sivabalan, S., Mirsky, G., and G. Fioccola, "RFC6374 Synonymous Flow Labels", Work in Progress, draft-ietf-mpls-rfc6374-sfl-01, December 2017.

Acknowledgements

The previous IETF specifications describing this technique were: [IP-MULTICAST-PM] and [OPSAWG-P3M].

The authors would like to thank Alberto Tempia Bonda, Domenico Laforgia, Daniele Accetta, and Mario Bianchetti for their contribution to the definition and the implementation of the method.

The authors would also thank Spencer Dawkins, Carlos Pignataro, Brian Haberman, and Eric Vyncke for their assistance and their detailed and precious reviews.

Authors' Addresses

Giuseppe Fioccola (editor)
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: giuseppe.fioccola@telecomitalia.it

Alessandro Capello
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: alessandro.capello@telecomitalia.it

Mauro Cociglio
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: mauro.cociglio@telecomitalia.it

Luca Castaldelli
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: luca.castaldelli@telecomitalia.it

Mach(Guoyi) Chen
Huawei Technologies

Email: mach.chen@huawei.com

Lianshu Zheng
Huawei Technologies

Email: vero.zheng@huawei.com

Greg Mirsky
ZTE
United States of America

Email: gregimirsky@gmail.com

Tal Mizrahi
Marvell
6 Hamada St.
Yokneam
Israel

Email: talmi@marvell.com