Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 2, 2017

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

March 1, 2017

Alternate Marking method for passive performance monitoring
draft-ietf-ippm-alt-mark-04

Abstract

This document describes a passive method to perform packet loss, delay and jitter measurements on live traffic. This method is based on Alternate Marking (Coloring) technique. A report on the operational experiment done at Telecom Italia is explained in order to give an example and show the method applicability. This technique can be applied in various situations as detailed in this document.

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 September 2, 2017.

Table of Contents

1. Introduction

Nowadays, most of the traffic in Service Providers' networks carries real time content. These contents are highly sensitive to packet loss [RFC2680], while interactive contents are sensitive to delay [RFC2679], and jitter [RFC3393].

In view of this scenario, Service Providers need methodologies and tools to monitor and measure network performances 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 OAM, that includes also performance monitoring techniques, has been done by Standards Developing Organizations(SDOs):: [RFC7276] provides a good overview of existing OAM mechanisms defined in IETF, ITU-T and IEEE. Considering IETF, a lot of work has been done on fault detection and connectivity verification, while a minor effort has been dedicated so far 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, possibly 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 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 a much better accuracy, especially for packet loss measurements.

This document is organized as follows:

- Section 2 gives an overview of the method, including a comparison with different measurement strategies;
- Section 3 describes the method in detail;
- Section 4 reports considerations about synchronization, data correlation and packet re-ordering;
- Section 5 reports examples of implementation and deployment of the method. Furthermore the operational experiment done at Telecom Italia is described;
- Section 8 includes some security aspects;
- Section 9 finally summarizes some concluding remarks.

2. Overview of the method

In order to perform packet loss measurements on a live 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 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 process it in real time could overload the device.

An alternate approach is to count the number of packets sent on one end, the number of packets received on the other end, and to compare the two values. This operation is much simpler to implement, but requires that the devices performing the measurement are 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 could 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 inserting periodically 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 "colors" 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.

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.

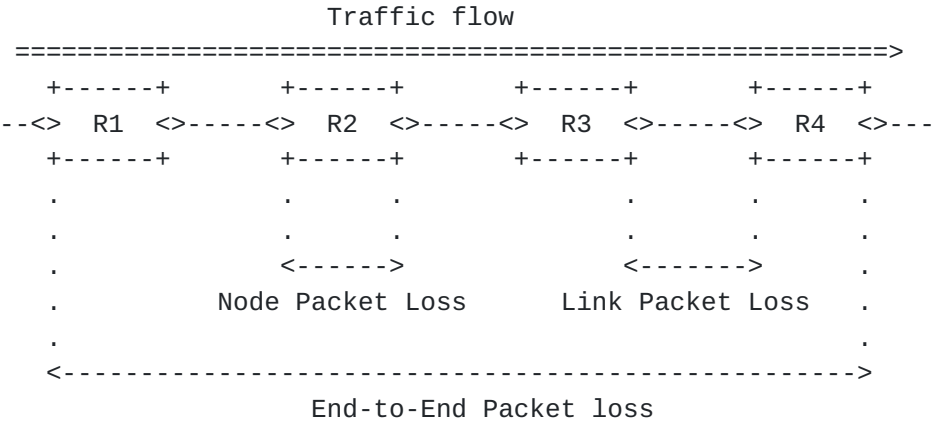

Figure 1: Available measurements

3. Detailed description of the method

This section describes in detail how the method operate. A special emphasis is given to the measurement of packet loss, that 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: 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); but 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

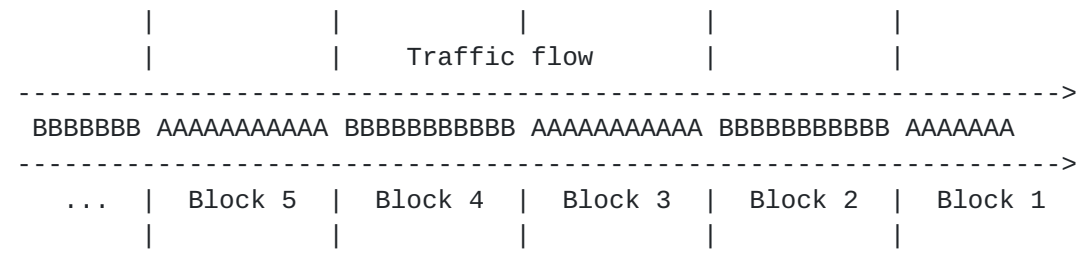

Figure 2: Traffic coloring

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.

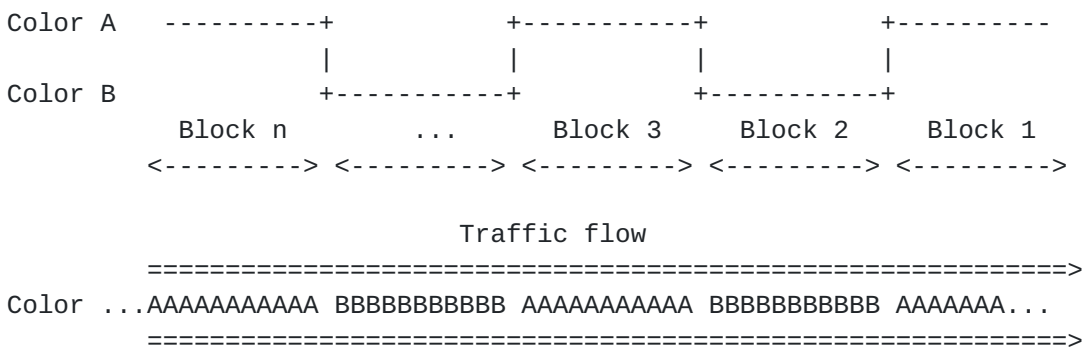

Figure 3: Computation of link packet loss

Traffic coloring could be done by R1 itself or by an upward router. 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 A, only counter C(A)R1 will be incremented, while C(B)R1 is not incremented; vice versa, 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 color 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 R2 egress interface it is possible to count the packets sent out of 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 a 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 to 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 to read 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, that 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 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
...	...	...	...	...	...
n	0	387	0	387	0
n+1	379	0	377	0	2

Table 1: Evaluation of counters for packet loss measurements

During an A block (blocks 1, 3 and n+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. Vice versa, during a B block (blocks 2, 4 and n), 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 and it is possible to read them, compare the values measured on router 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 blocks three and four, 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.

3.1.1. Timing aspects

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

By considering the clock error between network devices R1 and R2, they must be synchronized to the same clock reference with an accuracy of +/- L/2 time units, where L is the 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 belonging to different batches is L time units.

In practice, there are also out of order at batch boundaries, strictly related to the delay between measurement points. 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 out of order because of network delay.

The following figure eplains both issues.

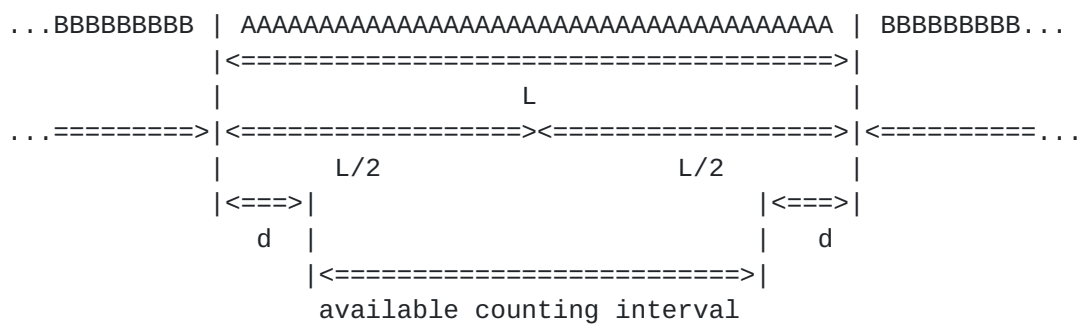

Figure 4: Timing aspects

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 guardband 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.

3.2. 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.

3.2.1. Single marking methodology

The alternation of colors can be used as a time reference to calculate the delay. Whenever the color changes (that 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. Considering Figure 2, R1 stores a timestamp TS(A1)R1 when it sends the first packet of block 1 (A-colored), a 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 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 (f.i. 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
...	...	...	...	...	...
n	77.463	-	80.501	-	3.038
n+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 referring 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 referring to the first packet of block 2 (which is B-colored). 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).

3.2.1.1. 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. 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 (f.i. 5 minutes), and it doesn't give the minimum, maximum and median delay values (RFC 6703 [RFC6703]). This limitation could be overcome by reducing the duration of the block (f.i. from 5 minutes to a few seconds), that implicates an highly optimized implementation of the method.

By summing the mean delays of the two directions of a path, it is also possible to measure the two-way mean delay (round-trip delay).

3.2.2. 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 is 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 values 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 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 be not 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 is 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 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 double marking method, where a subset of batch packets are 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 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 an useful information since it is possible to understand if the double marking measurements are actually representative of the delay trends.

3.3. Delay variation measurement

Similarly 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 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 a timestamp TS(A)R1 whenever it sends the first packet of a block and R2 stores a 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.

The 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 toggeled 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, instead, for one-way delay measurement 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.

- A possibility is to use a centralized solution using Network Management System (NMS) to correlate data;

- Another possibility is to define a protocol based distributed solution, by defining a new protocol or by extending the existing protocols (e.g. RFC6374, TWAMP, OWAMP) in order to communicate the counters and timestamps between nodes.

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

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

The alternate marking method described in this document literally split the packets of the measured flow into different measurement blocks, in addition a Block Number could be assigned to each of 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 NMS. The value of 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, same BNs associated with two packet counts from an upstream and a downstream node respectively, it considers that these two packet counts corresponding to the same block, i.e. that these two packet counts belong to the same block of markers from the upstream and downstream node. 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) [IEEE1588]). Synchronization aspects are further discussed in Section 4.

4.3. Packet Re-ordering

Due to ECMP, packet re-ordering is very common in IP network. The accuracy of marking based PM, especially packet loss measurement, may be affected by packet re-ordering. 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 the following paragraphs an example of data correlation mechanism is explained and could be use independently of the adopted solutions.

Most of the packet re-ordering occur at the edge of adjacent blocks, and they are easy to handle if the interval of each block is sufficient large. Then, it can assume that the packets with different marker belong to the block that they are more close to. If the interval is small, it is difficult and sometime impossible to determine to which block a packet belongs. See above example, 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.

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 re-ordering.

5. Implementation and deployment

The methodology described in the previous sections can be applied in various situations. Basically 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 can be easily recognized by the receiver.

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 at Telecom Italia

The method described in this document, also called PNPM (Packet Network Performance Monitoring), has been invented and engineered in Telecom Italia and it's currently being used in Telecom Italia's network. The methodology has been applied 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 the method to delay measurement is currently being evaluated in Telecom Italia's labs. This section describes 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 fundamental steps for this implementation of the method can be summarized in the following items:

- coloring the packets;
- counting the packets;
- collecting data and calculating the packet loss.
- metric transparency.

Before going deeper into the implementation details, 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. This could be the case, for example, of IPTV channels or other specific applications traffic with high QoS requirements (i.e. Mobile Backhauling traffic). 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, especially for IPTV traffic where static joins are frequently used to force traffic forwarding and replication. Another application is on Mobile Backhauling, implemented with a VPN MPLS in Telecom Italia's network; in this case the problem with unicast traffic is overcome by monitoring just the two Provider Edge nodes of the VPN MPLS.

- o 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 (for instance an Ethernet VLAN or a MPLS PW). 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 is needed for each class.

The current implementation in Telecom Italia uses the first strategy. 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, a Service Provider should be aware that, if a measurement is performed by grouping many flows, it is not possible to determine exactly which flow was affected by packets 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 have been identified, it is necessary to configure the monitoring on the proper nodes. Configuring the monitoring means configuring the policy 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 the 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, counters on other interfaces will be null.

5.1.1. 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, it is desirable, in general, to have a single coloring node because it is easier to manage and doesn't rise any risk of conflict (consider the case where two nodes color the same flow). Thus it is necessary to color the flow as close as possible to the source. In addition, coloring a flow close to the source allows an end-to-end measure if a measurement point is enabled on the last-hop router as well. The only requirement is that the coloring must change periodically and every node along the path

must be able to identify unambiguously the colored packets. 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. 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 less significant bits of the DSCP field (bit 0 and bit 1) to implement the method without affecting QoS policies. 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 color A and B) and creating the blocks.

In practice, coloring the traffic using the DSCP field can be implemented by configuring on the router output interface an access list that intercepts the flow(s) to be monitored and applies to them a policy 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 ca be used perform this task on the basis of a fixed timer. In Telecom Italia's implementation this timer is set to 5 minutes: this value showed to be a good compromise between measurement frequency and stability of the measurement (i.e. possibility to collect all the measures referring to the same block).

5.1.2. Counting the packets

Assuming that the coloring of the packets is performed only by the source node, 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, 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 od counters is needed. For example, in order to monitor separately 3 flows on a router with 4 interfaces involved, 24 counters are needed (2 counters for each of the 3 flows on each of the 4 interfaces). If traffic is colored using the DSCP field, as in Telecom Italia's implementation, an access-list that matches specific DSCP values can be used to count the packets of the flow(s) being monitored.

In case of link-based measurements the behaviour 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 to read the counters periodically, many times during a block duration, and to compare these successive readings: when the counter stops incrementing 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, if each block is 5 minutes long, reading the counter for color A every 5 minute 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. The choice of a 5 minutes timer for colore switching was also inspired by these considerations.

5.1.3. 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) is required 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 also where 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 (5 minutes).

If link-based measurement is used, it would 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 similar approach could be complicated if applied to a flow-based measurement.

5.1.4. Metric transparency

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 count with an access-list 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 anaware that packets were marked. So this restoring technique makes Alternate Marking completely transparent outside its monitoring domain.

5.2. IP flow performance measurement (IPFPM)

This application of marking method is described in [I-D.chen-ippm-coloring-based-ipfpm-framework].

5.3. Performance Measurement Marking Method in BIER Domain

In [I-D.ietf-bier-mpls-encapsulation] two OAM bits from Bit Index Explicit Replication (BIER) Header are reserved for the passive performance measurement marking method. [I-D.ietf-bier-pmmm-oam] details the measurement for multicast service over BIER domain.

5.4. Overlay OAM Passive Performance Measurement

The Overlay OAM Design Team is considering the preliminary OAM requirements from NVO3, BIER, and SFC. Marking Method is the preferred passive method to measure performance.

[I-D.ooamdt-rtgwg-ooam-requirement] and [I-D.ooamdt-rtgwg-oam-gap-analysis] explain in deep this item.

5.5. RFC6374 Use Case

RFC6374 [RFC6374] uses the 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. [I-D.ietf-mpls-flow-ident] 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 (SFL) introduced in [I-D.bryant-mpls-sfl-framework], while [I-D.bryant-mpls-rfc6374-sfl] describes RFC6374 performance measurements with SFL.

5.6. Application to active performance measurement

[I-D.fioccola-ippm-alt-mark-active] describes how to extend the existing Active Measurement Protocol, in order to implement alternate marking methodology. [I-D.fioccola-ippm-rfc6812-alt-mark-ext] describes an extension to the Cisco SLA Protocol Measurement-Type UDP-Measurement.

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 end points can exchanges 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 marking method can simplify also the active measurement, as explained in [I-D.fioccola-ippm-alt-mark-active].

7. Compliance with RFC6390 guidelines

RFC6390 [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 a new method of measurement for a few Performance Metrics that have already been standardized. Nevertheless, it's worth applying [RFC6390] guidelines to the present document, in order to provide a more complete and coherent description of the proposed method. We

used a subset of the Performance Metric Definition template defined by [RFC6390].

- Metric name and description: as already stated, this document doesn't propose any new Performance Metric. On the contrary, it describes a novel method for measuring packet loss [RFC2680]. The same concept, with small differences, can also be used to measure delay [RFC2679], 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.

- 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 Points: 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 have a constraint on the frequency of measurements. In order to perform a measure, the counter must be in a steady state: this happens when the traffic is being colored with the alternate color; for example in the Telecom Italia application of the method the time interval is set to 5 minutes.

- Implementation: the Telecom Italia application 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.

- 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 perform 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, as far the Telecom Italia application 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. 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 which 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. For what concerns the first point, the measurements described in this document are passive, so there are no 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 the IP header of 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 preserve stability and performance of routers doing the measurements. The measurements themselves 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.

The privacy concerns of network measurement are limited because the method only relies on information contained in the IP header without any release of user data.

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 is just in the context of one Service Provider's network, and thus the routers (or other network devices) are locally administered and this type of attack can be avoided.

One of the main security threats in OAM protocols is network reconnaissance; an attacker can gather information about the network performance by passively eavesdropping to 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 to data plane traffic does not allow attackers to gain information about the network performance.

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].

9. Conclusions

The advantages of the method described in this document are:

- easy implementation: it can be implemented using features already available on major routing platforms;
- 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/frame -based traffic: Ethernet, IP, MPLS, etc., 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;

- no interoperability issues: the features required to implement the method are available on all current routing platforms.

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.

10. IANA Considerations

There are no IANA actions required.

11. Acknowledgements

The previous IETF drafts about this technique were: [I-D.cociglio-mboned-multicast-pm] and [I-D.tempia-opsawg-p3m]. There are some references to this methodology in other IETF works (e.g. [I-D.ietf-mpls-flow-ident], [I-D.bryant-mpls-sfl-framework] [I-D.bryant-mpls-rfc6374-sfl], [I-D.ietf-bier-mpls-encapsulation], [I-D.ietf-bier-pmmm-oam] [I-D.chen-ippm-coloring-based-ipfpm-framework]).

In addition the authors would like to thank Domenico Laforgia, Daniele Accetta and Mario Bianchetti for their contribution to the definition and the implementation of the method.

12. References

12.1. Normative References

[RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", RFC 2679, DOI 10.17487/RFC2679, September 1999, <http://www.rfc-editor.org/info/rfc2679>.

[RFC2680] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Packet Loss Metric for IPPM", RFC 2680, DOI 10.17487/RFC2680, September 1999, <http://www.rfc-editor.org/info/rfc2680>.

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

12.2. Informative References

[I-D.bryant-mpls-rfc6374-sfl]
Bryant, S., Chen, M., Li, Z., Swallow, G., Sivabalan, S., Mirsky, G., and G. Fioccola, "RFC6374 Synonymous Flow Labels", draft-bryant-mpls-rfc6374-sfl-03 (work in progress), October 2016.

[I-D.bryant-mpls-sfl-framework]
Bryant, S., Chen, M., Li, Z., Swallow, G., Sivabalan, S., and G. Mirsky, "Synonymous Flow Label Framework", draft-bryant-mpls-sfl-framework-02 (work in progress), October 2016.

[I-D.chen-ippm-coloring-based-ipfpm-framework]
Chen, M., Zheng, L., Mirsky, G., Fioccola, G., and T. Mizrahi, "IP Flow Performance Measurement Framework", draft-chen-ippm-coloring-based-ipfpm-framework-06 (work in progress), March 2016.

[I-D.cociglio-mboned-multicast-pm]
Cociglio, M., Capello, A., Bonda, A., and L. Castaldelli, "A method for IP multicast performance monitoring", draft-cociglio-mboned-multicast-pm-01 (work in progress), October 2010.

[I-D.fioccola-ippm-alt-mark-active]
Fioccola, G., Clemm, A., Cociglio, M., Chandramouli, M., and A. Capello, "Alternate Marking Extension to Active Measurement Protocol", draft-fioccola-ippm-alt-mark-active-00 (work in progress), July 2016.

[I-D.fioccola-ippm-rfc6812-alt-mark-ext]
Fioccola, G., Clemm, A., Cociglio, M., Chandramouli, M., and A. Capello, "Alternate Marking Extension to Cisco SLA Protocol RFC6812", draft-fioccola-ippm-rfc6812-alt-mark-ext-01 (work in progress), March 2016.

[I-D.ietf-bier-mpls-encapsulation]
Wijnands, I., Rosen, E., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication in MPLS and non-MPLS Networks", draft-ietf-bier-mpls-encapsulation-06 (work in progress), December 2016.

[I-D.ietf-bier-pmmm-oam]
Mirsky, G., Zheng, L., Chen, M., and G. Fioccola, "Performance Measurement (PM) with Marking Method in Bit Index Explicit Replication (BIER) Layer", draft-ietf-bier-pmmm-oam-01 (work in progress), January 2017.

[I-D.ietf-mpls-flow-ident]
Bryant, S., Pignataro, C., Chen, M., Li, Z., and G. Mirsky, "MPLS Flow Identification Considerations", draft-ietf-mpls-flow-ident-04 (work in progress), February 2017.

[I-D.ooamdt-rtgwg-oam-gap-analysis]
Mirsky, G., Nordmark, E., Pignataro, C., Kumar, N., Kumar, D., Chen, M., Yizhou, L., Mozes, D., Networks, J., and I. Bagdonas, "Operations, Administration and Maintenance (OAM) for Overlay Networks: Gap Analysis", draft-ooamdt-rtgwg-oam-gap-analysis-02 (work in progress), July 2016.

[I-D.ooamdt-rtgwg-ooam-requirement]
Kumar, N., Pignataro, C., Kumar, D., Mirsky, G., Chen, M., Nordmark, E., Networks, J., and D. Mozes, "Overlay OAM Requirements", draft-ooamdt-rtgwg-ooam-requirement-02 (work in progress), January 2017.

[I-D.tempia-opsawg-p3m]
Capello, A., Cociglio, M., Castaldelli, L., and A. Bonda, "A packet based method for passive performance monitoring", draft-tempia-opsawg-p3m-04 (work in progress), February 2014.

[RFC5481] Morton, A. and B. Claise, "Packet Delay Variation Applicability Statement", RFC 5481, DOI 10.17487/RFC5481, March 2009, <http://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, <http://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, <http://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, <http://www.rfc-editor.org/info/rfc6703>.

[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, <http://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, <http://www.rfc-editor.org/info/rfc7384>.

Authors' Addresses

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

Email: giuseppe.fioccola@telecomitalia.it

Alessandro Capello (editor)
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 (editor)
Huawei Technologies

Email: mach.chen@huawei.com

Lianshu Zheng (editor)
Huawei Technologies

Email: vero.zheng@huawei.com

Greg Mirsky (editor)
ZTE
USA

Email: gregimirsky@gmail.com

Tal Mizrahi (editor)
Marvell
6 Hamada st.
Yokneam
Israel

Email: talmi@marvell.com