IPPM Working Group
Internet-Draft
Intended status: Experimental
Expires: May 3, 2021

M. Cociglio
Telecom Italia
C. Corbo
Politecnico di Torino
G. Fioccola
Huawei Technologies
M. Nilo
Telecom Italia
R. Sisto
Politecnico di Torino

October 30, 2020

The Big Data Approach for Multipoint Alternate Marking method draft-c2f-ippm-big-data-alt-mark-01

Abstract

This document introduces a new approach for the Alternate Marking method. It is called Big Data Multipoint Alternate Marking method and, starting from the methodology described in RFC 8321 and RFC 8889, it explains how to implement performance measurement analytics on the Network Management System by analysing the raw data of the network nodes.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 3, 2021.

Table of Contents

1. Introduction 2
2. Marking Methods Classification 3
3. Scenario and Background 4
4. Methodology 6
4.1. Data collecting 6
4.2. Sending data 8
4.3. Preprocessing 8
4.4. Results 9
5. Security Considerations 10
6. Acknowledgements 10
7. IANA Considerations 10
8. References 10
8.1. Normative References 10
8.2. Informative References 11
Authors' Addresses 11

1. Introduction

This document describes a scenario and a methodology that can be used to get performance details from a monitored network. The approach is inspired by the concepts illustrated in the Alternate Marking Method (RFC 8321 [RFC8321]), Multipoint Alternate Marking Method (RFC 8889 [RFC8889]), and Hash Sampling (RFC 5474 [RFC5474] and RFC 5475 [RFC5475]).

In general the performance measurement results are based on a posteriori calculation and the method is called Big Data Multipoint Alternate Marking performance measurement.

The kinds of measurements are specified on the Network Management System (NMS) and they can be split into two main categories: per cluster and end-to-end.

- The per cluster approach includes all the details that refer to each single cluster and provides a list of parameters that characterize it (packet loss, mean delay).

- The end-to-end approach provides more general information about the entire path (packet loss, mean delay).

The results can be provided on demand, in a non real-time processing environment, and each one of them refers to a single monitoring period, even if it is possible to broaden the search to more periods.

The basic mechanism of the Big data approach here introduced is the Packet sampling. Packet sampling, which is performed through Hashing Sampling technique (RFC 5474 [RFC5474] and RFC 5475 [RFC5475]) applied on all incoming traffic, without any flow distinction. Nevertheless, thanks to data postprocessing, results are split by flow afterwards, since the storage system memorizes the fields of the packet headers that identify flows. The NMS, in fact, requires, as input parameters, the flow identification fields as well as the timestamp.

The use of hash sampling improves packet tracking performance and thus overall performance. It allows to track the path followed by each packet without further efforts by the NMS.

2. Marking Methods Classification

[I-D.mizrahi-ippm-compact-alternate-marking] presents a summary of the alternate marking methods, and discusses the trade-offs among them.

The methodologies are classified as follows:

- Double Marking,

- Single Marking,

- Hash-based Marking.

Double Marking and Single Marking are described in RFC 8321 [RFC8321] and RFC 8889 [RFC8889].

While, Hash-based selection can be leveraged as a marking method, allowing a zero-bit marking approach. As defined in RFC 5475 [RFC5475]:

> A Hash Function h maps the Packet Content c, or some portion of it, onto a Hash Range R. The packet is selected if h(c) is an element of S, which is a subset of R called the Hash Selection Range.

The Hash-Based marking requires the hash function and the set S to be configured consistently across the measurement points. It is worth mentioning that the duration between sampled packets depends only on the hash value.

The single marking approach can be combined with hash-based sampling as described in [I-D.mizrahi-ippm-compact-alternate-marking]: a single marking bit is used for the loss measurement, while the hash-based sampling is used to trigger delay measurement. In the same way, the hash-based sampling can be used in multipoint network, and this is explained in RFC 8889 [RFC8889].

3. Scenario and Background

The service provider's network is made up of a main backbone network surrounded by routers that handle customers traffic input and output. The proposed methodology requires that the traffic is marked before entering the backbone network, by means of the Alternate Marking technique. The marking process can be made by the edge routers or by the customers itself, keeping in mind that it requires that the markers are synchronized.

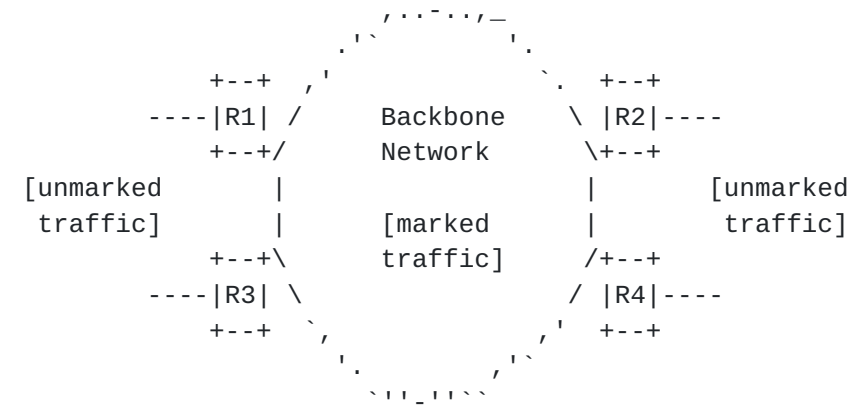

Figure 1: Backbone Network

Only the marked traffic can be monitored. In case of one marking bit, all the traffic must be marked. Instead, in case of two marking bits, it is possible to mark the traffic partially, therefore the results will not be affected by unmarked packets and will refer only to the marked ones.

In order to apply the Alternate Marking methodology a time reference period and a marking method must be fixed at the beginning. The time reference period must consider the misalignment between the marking source routers, clock error between network devices and the interval we need to wait to avoid packets being out of order because of network delay, as described in RFC 8321 [RFC8321] and RFC 8889 [RFC8889].

A possible marking method could use two bits of the header and set them to 0x01, to identify a period, and to 0x10 to identify the next one. This allows to distinguish between marked traffic and unmarked traffic, instead of using just one bit, which can can generate misunderstanding between the unmarked traffic (that has the marked bit set to 0 by default) and the marked traffic (that alternates between 0x0 and 0x1, with 0x0 as marker value and not as default). As an alternative, it is possible to use just one marking bit and utilize a filter based on IP subnets to exclude the flows from monitoring. The flows that do not have to be monitored are those internal to the network (that usually have private IP addresses).

To enable the Big Data approach for monitoring, the network nodes require a packet collector, that is the agent installed on board of the network node that collects measurements, based on the configured Packet sampling criteria.

The portion of network to be monitored must be delimited by routers with packet collector installed on. The rest of the network cannot be monitored even if the traffic is marked. So, the size of the monitored network depends on the network devices placement. However, the size of the network surrounded by packet collectors must be less than or equal to the size of the network with marked traffic.

It is worth highlighting that, if one marking bit is used, the requirement is that all the ingress traffic from the boundary nodes must be marked. While, if two marking bits are used, the marking is applied even on flow basis by the boundary nodes delimiting the monitored network and it is easy to recognize the marked traffic within the network. Moreover, both in the case of one and two marking bits, we need to ensure that all the marked traffic, both ingress and egress, comes through the monitoring nodes in order to guarantee to properly monitor the network.

4. Methodology

The method described here consists of the following steps:

1. Data collecting;

2. Sending data;

3. Preprocessing;

4. Results.

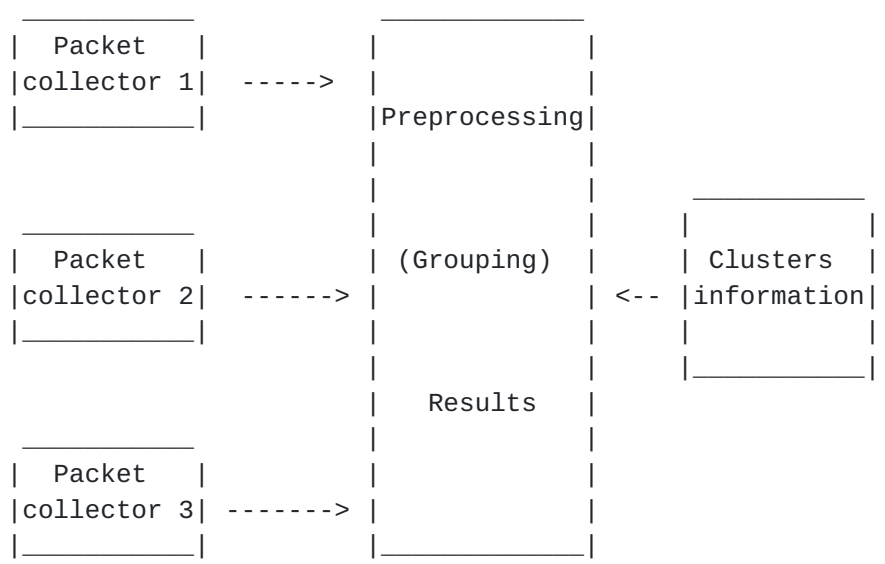

Figure 2: Outline of the Methodology

4.1. Data collecting

The Data collecting phase implies that, on board of the network nodes, the packet collector analyses data passing through a network interface. A packet collector needs to be placed into each router interface we want to monitor.

The agent is configured by setting:

- the reference hash,

- the maximum number of packets to store,

- the alternate marking period duration,

- the one or the two alternate marking bits that identify the marked flow,
- the interface to monitor,
- the flows to exclude from monitoring (i.e identified by header IP fields): to be used only in case of one marking bit.

In general, if one alternate marking bit is available it is always possible to identify the flow. In this case it must be used a filter that excludes from monitoring the traffic flows that are not marked in the network (e.g. IP addresses in use in the transit network that often use private addresses) and, at the same time, it is needed to make sure that all the ingress traffic is marked. This is a little more complicated but helps to address the case of IPv4 where only one bit could be available (i.e. so called unused bit). In this regard, the flows to exclude from monitoring are needed only for the case of one marking bit in order to identify the marked traffic. On the other hand, these are not necessary in case of two marking bits because it is already easy to identify the marked traffic and monitor only this portion.

The Data collection is based on Hashing sampling described in RFC 5474 [RFC5474] and RFC 5475 [RFC5475].

The process is recursive. Each incoming packet is hashed, compared with the reference hash, and recorded if the number of bits that are matching are the same of those required by the packet collector . When the number of matched packets exceed the maximum number requested in configuration, the number of bits to match is increased by one. At this point all the previous stored packets could be potentially discarded and must be rechecked. So data are stored temporally and are subject to changes, discards and additions. After the period ends, previous data are still subject to change, but after a guard band (reasonably L/2 if L is the period duration) data are stored permanently and ready to be sent.

Note that the packet collector (carried out with probe) selects the packets based on the configured parameters, so it works with every incoming packet, without distinction. This greatly increases the probe configuration easiness. Otherwise the probe should save all possible flows (potentially too many), which would be too expensive for the device, and need to be reconfigured if a new flow is available for performance monitoring. On the other hand, this increases the amount of data collected.

Stored data include two kind of details: one refers to each single packet and the other one is about aggregate measures.

The first set of data includes the fields that identify the flow (IP header fields), packet hash, timestamp when the packets come in, period to which data refer.

The second set of data reports network interface identification, total counted packets, total hashed packets, mean timestamp based on all the timestamp of all packets that passed through the interface, period.

4.2. Sending data

The Sending data phase is separated from the previous one. Once the data has been stored and collected as logs by the network device following the provisions of the theoretical model, the sending system has only the task of carrying data safely and reliably. It is possible to use a synchronous mechanism, in which the sending system periodically checks the availability of new data, or an asynchronous mechanism. In the last case when a new batch of data is ready, an alert wakes up the sending system that carries them to the destination.

4.3. Preprocessing

The Preprocessing phase has two main goals:

- aggregate input data to produce a new record that is ready to be postprocessed and that makes it easier to obtain performance parameters;

- decrease the total amount of data to store.

Although this step is not mandatory, it is recommended to speed up subsequent operations and to give a better shape to the stored data in order to fit well with the last queries.

Preprocessing can be done after data has been stored into the NMS in an iterative loop that parses that periodically or just before to be sent to NMS, through a consolidator, that collects data that comes from all network devices, parses them and then sends them to the NMS.

However, in this phase it is possible to group incoming data from all devices and determine the path followed by each sampled packet. In order to do that, if the data are grouped by hash and ordered them by timestamp, it is possible to outline the path.

After providing to the NMS the topology information and Clusters partition of the monitored network, it is also possible to track the crossed cluster for each couple of sorted data, by analyzing the

interface ID available in the stored record and comparing them with the edge that characterizes the clusters available in the monitored network.

4.4. Results

The Results phase involves the preprocessed records lay into database. When necessary the storage system can be queried, in a deferred time. The records are organized to fit well with the queries that care about timing and loss aspects.

Results are achieved by querying the storage system properly. Certainly, input parameters that identify which flow we are addressing are required. Additionally, time reference is needed to select only the packets of interest. The Big data system is aware of flow identification fields and performs packet flow grouping on the fly. The results described below can refer to different flows, depending on which parameters have been specified for the query.

It is possible to deduce the cluster mean delay D_i (mean delay referred to cluster i), by analyzing each record, computing delay d_j (delay referred to record j) as difference between the two available timestamps, that correspond to the input timestamp (when the packet has gone into the cluster) and the output timestamp (when the packet has gone out of that cluster), and summing it with all other delays; then the result is divided by the number of records that refer to the same cluster:

D_i = [d_0 + d_1 +...+ d_(N_i-1)] / N_i

Where D_i is the mean delay related to cluster i, d_j the delay related to record j, N_i the total number of records belonging to cluster i.

It could also be computed the end-to-end mean delay AD as the sum of all delays available in our database, and dividing it by all the records:

AD = [ad_0 + ad_1 +...+ ad_(M-1)] / M

Where AD is the end-to-end mean delay, ad_j the delay related to records j, and M the total number of records.

If necessary, after observing an unusual cluster delay, it could be possible to compute also max/avg/min link delay, by analyzing records again, and exploiting the difference between the two timestamps.

Additionally, also details about loss are available. Since the total packets are counted by each node, the sum of the input packets must be equal to the sum of the output packets inside each cluster. If their difference is greater than 0, then a loss has occurred, and the result is the total loss. The total packet loss per cluster:

PL_i = [p_(i,0) + p_(i,1) +...+ p_(i,K-1)] - [p_(o,0) + p_(o,1) +...+ p_(o,L-1)]

Considering cluster i with K input nodes and L output nodes, the calculation follows RFC 8889 [RFC8889].

In the same way it is possible to get the entire packet loss, as the sum of all the packet loss per cluster. The same measure can be obtained by using only the hashed packets, but in this case, we get an approximate measurement that might reflect or not the real one.

Notice that all these measurements refers to the flow we specify as input of the query and that the specified flow can include or not all the sampled packets (e.g. filter on ip_src=0.0.0.0/0, ip_dst=0.0.0.0/0, port_src=/, port_dst=/, type=tcp, outlines a flow that includes all the TCP packets in an IP network).

5. Security Considerations

This document specifies a method of performing measurements that does not directly affect Internet security or applications that run on the Internet. However, implementation of this method must be mindful of security and privacy concerns, as explained in RFC 8321 [RFC8321] and RFC 8889 [RFC8889].

6. Acknowledgements

The authors would like to thank Guido Marchetto for the precious contribution.

7. IANA Considerations

This document has no IANA actions

8. References

8.1. Normative References

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

8.2. Informative References

[I-D.mizrahi-ippm-compact-alternate-marking]
Mizrahi, T., Arad, C., Fioccola, G., Cociglio, M., Chen, M., Zheng, L., and G. Mirsky, "Compact Alternate Marking Methods for Passive and Hybrid Performance Monitoring", draft-mizrahi-ippm-compact-alternate-marking-05 (work in progress), July 2019.

[RFC5474] Duffield, N., Ed., Chiou, D., Claise, B., Greenberg, A., Grossglauser, M., and J. Rexford, "A Framework for Packet Selection and Reporting", RFC 5474, DOI 10.17487/RFC5474, March 2009, <https://www.rfc-editor.org/info/rfc5474>.

[RFC5475] Zseby, T., Molina, M., Duffield, N., Niccolini, S., and F. Raspall, "Sampling and Filtering Techniques for IP Packet Selection", RFC 5475, DOI 10.17487/RFC5475, March 2009, <https://www.rfc-editor.org/info/rfc5475>.

[RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <https://www.rfc-editor.org/info/rfc8321>.

[RFC8889] Fioccola, G., Ed., Cociglio, M., Sapio, A., and R. Sisto, "Multipoint Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8889, DOI 10.17487/RFC8889, August 2020, <https://www.rfc-editor.org/info/rfc8889>.

Authors' Addresses

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

Email: mauro.cociglio@telecomitalia.it

Calogero Corbo
Politecnico di Torino

Email: corbocalo94@gmail.com

Giuseppe Fioccola
Huawei Technologies
Riesstrasse, 25
Munich 80992
Germany

Email: giuseppe.fioccola@huawei.com

Massimo Nilo
Telecom Italia
Via Reiss Romoli, 274
Torino 10148
Italy

Email: massimo.nilo@telecomitalia.it

Riccardo Sisto
Politecnico di Torino
Corso Duca degli Abruzzi, 24
Torino 10129
Italy

Email: riccardo.sisto@polito.it