Internet Engineering Task Force
INTERNET-DRAFT, Intended Status: Informational
Expires December 23,2017
June 21, 2017

L. Avramov
Google
J. Rapp
VMware

Data Center Benchmarking Methodology
draft-ietf-bmwg-dcbench-methodology-16

Abstract

The purpose of this informational document is to establish test and evaluation methodology and measurement techniques for physical network equipment in the data center. A pre-requisite to this publication is the terminology document [1]. Many of these terms and methods may be applicable beyond this publication's scope as the technologies originally applied in the data center are deployed elsewhere.

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."

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction

Traffic patterns in the data center are not uniform and are constantly changing. They are dictated by the nature and variety of applications utilized in the data center. It can be largely east-west traffic flows in one data center and north-south in another, while others may combine both. Traffic patterns can be bursty in nature and contain many-to-one, many-to-many, or one-to-many flows. Each flow may also be small and latency sensitive or large and throughput sensitive while containing a mix of UDP and TCP traffic. All of these can coexist in a single cluster and flow through a single network device simultaneously. Benchmarking of network devices have long used [RFC1242], [RFC2432], [RFC2544], [RFC2889] and [RFC3918] which have

largely been focused around various latency attributes and Throughput [RFC2889] of the Device Under Test (DUT) being benchmarked. These standards are good at measuring theoretical Throughput, forwarding rates and latency under testing conditions; however, they do not represent real traffic patterns that may affect these networking devices.

Currently, typical data center networking devices are characterized by:

-High port density (48 ports of more)

-High speed (up to 100 GB/s currently per port)

-High throughput (line rate on all ports for Layer 2 and/or Layer 3)

-Low latency (in the microsecond or nanosecond range)

-Low amount of buffer (in the MB range per networking device)

-Layer 2 and Layer 3 forwarding capability (Layer 3 not mandatory)

This document provides a methodology for benchmarking Data Center physical network equipment DUT including congestion scenarios, switch buffer analysis, microburst, head of line blocking, while also using a wide mix of traffic conditions. The terminology document [1] is a pre-requisite.

1.1. 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].

1.2. Methodology format and repeatability recommendation

The format used for each section of this document is the following:

-Objective

-Methodology

-Reporting Format: Additional interpretation of RFC2119 terms:

MUST: required metric or benchmark for the scenario described (minimum)

SHOULD or RECOMMENDED: strongly suggested metric for the scenario described

MAY: Optional metric for the scenario described

For each test methodology described, it is critical to obtain repeatability in the results. The recommendation is to perform enough iterations of the given test and to make sure the result is consistent. This is especially important for section 3, as the buffering testing has been historically the least reliable. The number of iterations SHOULD be explicitly reported. The relative standard deviation SHOULD be below 10%.

2. Line Rate Testing

2.1 Objective

Provide a maximum rate test for the performance values for Throughput, latency and jitter. It is meant to provide the tests to perform, and methodology to verify that a DUT is capable of forwarding packets at line rate under non-congested conditions.

2.2 Methodology

A traffic generator SHOULD be connected to all ports on the DUT. Two tests MUST be conducted: a port-pair test [RFC 2544/3918 section 15 compliant] and also in a full mesh type of DUT test [2889/3918

section 16 compliant].

For all tests, the test traffic generator sending rate MUST be less than or equal to 99.98% of the nominal value of Line Rate (with no further PPM adjustment to account for interface clock tolerances), to ensure stressing the DUT in reasonable worst case conditions (see RFC [1] section 5 for more details --note to RFC Editor, please replace all [1] references in this document with the future RFC number of that draft). Tests results at a lower rate MAY be provided for better understanding of performance increase in terms of latency and jitter when the rate is lower than 99.98%. The receiving rate of the traffic SHOULD be captured during this test in % of line rate.

The test MUST provide the statistics of minimum, average and maximum of the latency distribution, for the exact same iteration of the test.

The test MUST provide the statistics of minimum, average and maximum of the jitter distribution, for the exact same iteration of the test.

Alternatively when a traffic generator can not be connected to all ports on the DUT, a snake test MUST be used for line rate testing, excluding latency and jitter as those became then irrelevant. The snake test consists in the following method:

-connect the first and last port of the DUT to a traffic generator

-connect back to back sequentially all the ports in between: port 2 to 3, port 4 to 5 etc to port n-2 to port n-1; where n is the total number of ports of the DUT

-configure port 1 and 2 in the same vlan X, port 3 and 4 in the same vlan Y, etc. port n-1 and port n in the same vlan Z.

This snake test provides a capability to test line rate for Layer 2 and Layer 3 RFC 2544/3918 in instance where a traffic generator with only two ports is available. The latency and jitter are not to be considered with this test.

2.3 Reporting Format

The report MUST include:

-physical layer calibration information as defined into [1] section 4.

-number of ports used

-reading for "Throughput received in percentage of bandwidth", while sending 99.98% of nominal value of Line Rate on each port, for each packet size from 64 bytes to 9216 bytes. As guidance, an increment of 64 byte packet size between each iteration being ideal, a 256 byte and 512 bytes being are also often used. The most common packets sizes order for the report is:
64b,128b,256b,512b,1024b,1518b,4096,8000,9216b.

The pattern for testing can be expressed using [RFC 6985].

-Throughput needs to be expressed in % of total transmitted frames

-For packet drops, they MUST be expressed as a count of packets and SHOULD be expressed in % of line rate

-For latency and jitter, values expressed in unit of time [usually microsecond or nanosecond] reading across packet size from 64 bytes to 9216 bytes

-For latency and jitter, provide minimum, average and maximum values. If different iterations are done to gather the minimum, average and maximum, it SHOULD be specified in the report along with a justification on why the information could not have been gathered at the same test iteration

-For jitter, a histogram describing the population of packets measured per latency or latency buckets is RECOMMENDED

-The tests for Throughput, latency and jitter MAY be conducted as individual independent trials, with proper documentation in the report but SHOULD be conducted at the same time.

-The methodology makes an assumption that the DUT has at least nine ports, as certain methodologies require that number of ports or more.

3. Buffering Testing

3.1 Objective

To measure the size of the buffer of a DUT under typical|many|multiple conditions. Buffer architectures between multiple DUTs can differ and include egress buffering, shared egress

buffering SoC (Switch-on-Chip), ingress buffering or a combination. The test methodology covers the buffer measurement regardless of buffer architecture used in the DUT.

3.2 Methodology

A traffic generator MUST be connected to all ports on the DUT.

The methodology for measuring buffering for a data-center switch is based on using known congestion of known fixed packet size along with maximum latency value measurements. The maximum latency will increase until the first packet drop occurs. At this point, the maximum latency value will remain constant. This is the point of inflection of this maximum latency change to a constant value. There MUST be multiple ingress ports receiving known amount of frames at a known fixed size, destined for the same egress port in order to create a known congestion condition. The total amount of packets sent from the oversubscribed port minus one, multiplied by the packet size represents the maximum port buffer size at the measured inflection point.

1) Measure the highest buffer efficiency

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

First iteration: ingress port 1 sending line rate to egress port 2, while port 3 sending a known low amount of over-subscription traffic (1% recommended) with a packet size of 64 bytes to egress port 2. Measure the buffer size value of the number of frames sent from the port sending the oversubscribed traffic up to the inflection point multiplied by the frame size.

Second iteration: ingress port 1 sending line rate to egress port 2, while port 3 sending a known low amount of over-subscription traffic (1% recommended) with same packet size 65 bytes to egress port 2. Measure the buffer size value of the number of frames sent from the port sending the oversubscribed traffic up to the inflection point multiplied by the frame size.

Last iteration: ingress port 1 sending line rate to egress port 2, while port 3 sending a known low amount of over-subscription traffic (1% recommended) with same packet size B bytes to egress port 2. Measure the buffer size value of the number of frames sent from the port sending the oversubscribed traffic up to the inflection point

multiplied by the frame size.

When the B value is found to provide the largest buffer size, then size B allows the highest buffer efficiency.

2) Measure maximum port buffer size

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

At fixed packet size B determined in procedure 1), for a fixed default Differentiated Services Code Point (DSCP)/Class of Service (COS) value of 0 and for unicast traffic proceed with the following:

First iteration: ingress port 1 sending line rate to egress port 2, while port 3 sending a known low amount of over-subscription traffic (1% recommended) with same packet size to the egress port 2. Measure the buffer size value by multiplying the number of extra frames sent by the frame size.

Second iteration: ingress port 2 sending line rate to egress port 3, while port 4 sending a known low amount of over-subscription traffic (1% recommended) with same packet size to the egress port 3. Measure the buffer size value by multiplying the number of extra frames sent by the frame size.

Last iteration: ingress port N-2 sending line rate traffic to egress port N-1, while port N sending a known low amount of over-subscription traffic (1% recommended) with same packet size to the egress port N. Measure the buffer size value by multiplying the number of extra frames sent by the frame size.

This test series MAY be repeated using all different DSCP/COS values of traffic and then using Multicast type of traffic, in order to find if there is any DSCP/COS impact on the buffer size.

3) Measure maximum port pair buffer sizes

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

First iteration: ingress port 1 sending line rate to egress port 2;

ingress port 3 sending line rate to egress port 4 etc. Ingress port N-1 and N will respectively over subscribe at 1% of line rate egress port 2 and port 3. Measure the buffer size value by multiplying the number of extra frames sent by the frame size for each egress port.

Second iteration: ingress port 1 sending line rate to egress port 2; ingress port 3 sending line rate to egress port 4 etc. Ingress port N-1 and N will respectively over subscribe at 1% of line rate egress port 4 and port 5. Measure the buffer size value by multiplying the number of extra frames sent by the frame size for each egress port.

Last iteration: ingress port 1 sending line rate to egress port 2; ingress port 3 sending line rate to egress port 4 etc. Ingress port N-1 and N will respectively over subscribe at 1% of line rate egress port N-3 and port N-2. Measure the buffer size value by multiplying the number of extra frames sent by the frame size for each egress port.

This test series MAY be repeated using all different DSCP/COS values of traffic and then using Multicast type of traffic.

4) Measure maximum DUT buffer size with many to one ports

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

First iteration: ingress ports 1,2,... N-1 sending each [(1/[N-1])*99.98]+[1/[N-1]] % of line rate per port to the N egress port.

Second iteration: ingress ports 2,... N sending each [(1/[N-1])*99.98]+[1/[N-1]] % of line rate per port to the 1 egress port.

Last iteration: ingress ports N,1,2...N-2 sending each [(1/[N-1])*99.98]+[1/[N-1]] % of line rate per port to the N-1 egress port.

This test series MAY be repeated using all different COS values of traffic and then using Multicast type of traffic.

Unicast traffic and then Multicast traffic SHOULD be used in order to determine the proportion of buffer for documented selection of tests. Also the COS value for the packets SHOULD be provided for each test iteration as the buffer allocation size MAY differ per COS value. It is RECOMMENDED that the ingress and egress ports are varied in a random, but documented fashion in multiple tests to measure the buffer size for each port of the DUT.

3.3 Reporting format

The report MUST include:

- The packet size used for the most efficient buffer used, along with DSCP/COS value

- The maximum port buffer size for each port

- The maximum DUT buffer size

- The packet size used in the test

- The amount of over-subscription if different than 1%

- The number of ingress and egress ports along with their location on the DUT

- The repeatability of the test needs to be indicated: number of iterations of the same test and percentage of variation between results for each of the tests (min, max, avg)

The percentage of variation is a metric providing a sense of how big the difference between the measured value and the previous ones.

For example, for a latency test where the minimum latency is measured, the percentage of variation of the minimum latency will indicate by how much this value has varied between the current test executed and the previous one.

PV=((x2-x1)/x1)*100 where x2 is the minimum latency value in the current test and x1 is the minimum latency value obtained in the previous test.

The same formula is used for max and avg variations measured.

4 Microburst Testing

4.1 Objective

To find the maximum amount of packet bursts a DUT can sustain under various configurations.

This test provides additional methodology to the other RFC tests:

-All bursts should be send with 100% intensity. Note: intensity is defined in [1] section 6.1.1

-All ports of the DUT must be used for this test

-All ports are recommended to be testes simultaneously

4.2 Methodology

A traffic generator MUST be connected to all ports on the DUT. In order to cause congestion, two or more ingress ports MUST send bursts of packets destined for the same egress port. The simplest of the setups would be two ingress ports and one egress port (2-to-1).

The burst MUST be sent with an intensity of 100% (intensity is defined in [1] section 6.1.1), meaning the burst of packets will be sent with a minimum inter-packet gap. The amount of packet contained in the burst will be trial variable and increase until there is a non-zero packet loss measured. The aggregate amount of packets from all the senders will be used to calculate the maximum amount of microburst the DUT can sustain.

It is RECOMMENDED that the ingress and egress ports are varied in multiple tests to measure the maximum microburst capacity.

The intensity of a microburst MAY be varied in order to obtain the microburst capacity at various ingress rates. Intensity of microburst is defined in [1].

It is RECOMMENDED that all ports on the DUT will be tested simultaneously and in various configurations in order to understand all the combinations of ingress ports, egress ports and intensities.

An example would be:

First Iteration: N-1 Ingress ports sending to 1 Egress Ports

Second Iterations: N-2 Ingress ports sending to 2 Egress Ports

Last Iterations: 2 Ingress ports sending to N-2 Egress Ports

4.3 Reporting Format

The report MUST include:

- The maximum number of packets received per ingress port with the maximum burst size obtained with zero packet loss

- The packet size used in the test

- The number of ingress and egress ports along with their location on the DUT

- The repeatability of the test needs to be indicated: number of iterations of the same test and percentage of variation between results (min, max, avg)

5. Head of Line Blocking

5.1 Objective

Head-of-line blocking (HOLB) is a performance-limiting phenomenon that occurs when packets are held-up by the first packet ahead waiting to be transmitted to a different output port. This is defined in RFC 2889 section 5.5, Congestion Control. This section expands on RFC 2889 in the context of Data Center Benchmarking.

The objective of this test is to understand the DUT behavior under head of line blocking scenario and measure the packet loss.

Here are the differences between this HOLB test and RFC 2889:

-This HOLB starts with 8 ports in two groups of 4, instead of 4 RFC 2889

-This HOLB shifts all the port numbers by one in a second iteration of the test, this is new compared to RFC 2889. The shifting port numbers continue until all ports are the first in the group. The purpose is to make sure to have tested all permutations to cover differences of behavior in the SoC of the DUT

-Another test in this HOLB expands the group of ports, such that traffic is divided among 4 ports instead of two (25% instead of 50% per port)

-Section 5.3 adds additional reporting requirements from Congestion Control in RFC 2889

5.2 Methodology

In order to cause congestion in the form of head of line blocking, groups of four ports are used. A group has 2 ingress and 2 egress ports. The first ingress port MUST have two flows configured each going to a different egress port. The second ingress port will congest the second egress port by sending line rate. The goal is to

measure if there is loss on the flow for the first egress port which is not over-subscribed.

A traffic generator MUST be connected to at least eight ports on the DUT and SHOULD be connected using all the DUT ports.

1) Measure two groups with eight DUT ports

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

First iteration: measure the packet loss for two groups with consecutive ports

The first group is composed of: ingress port 1 is sending 50% of traffic to egress port 3 and ingress port 1 is sending 50% of traffic to egress port 4. Ingress port 2 is sending line rate to egress port 4. Measure the amount of traffic loss for the traffic from ingress port 1 to egress port 3.

The second group is composed of: ingress port 5 is sending 50% of traffic to egress port 7 and ingress port 5 is sending 50% of traffic to egress port 8. Ingress port 6 is sending line rate to egress port 8. Measure the amount of traffic loss for the traffic from ingress port 5 to egress port 7.

Second iteration: repeat the first iteration by shifting all the ports from N to N+1.

The first group is composed of: ingress port 2 is sending 50% of traffic to egress port 4 and ingress port 2 is sending 50% of traffic to egress port 5. Ingress port 3 is sending line rate to egress port 5. Measure the amount of traffic loss for the traffic from ingress port 2 to egress port 4.

The second group is composed of: ingress port 6 is sending 50% of traffic to egress port 8 and ingress port 6 is sending 50% of traffic to egress port 9. Ingress port 7 is sending line rate to egress port 9. Measure the amount of traffic loss for the traffic from ingress port 6 to egress port 8.

Last iteration: when the first port of the first group is connected on the last DUT port and the last port of the second group is connected to the seventh port of the DUT.

Measure the amount of traffic loss for the traffic from ingress port N to egress port 2 and from ingress port 4 to egress port 6.

2) Measure with N/4 groups with N DUT ports

The tests described in this section have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

The traffic from ingress split across 4 egress ports (100/4=25%).

First iteration: Expand to fully utilize all the DUT ports in increments of four. Repeat the methodology of 1) with all the group of ports possible to achieve on the device and measure for each port group the amount of traffic loss.

Second iteration: Shift by +1 the start of each consecutive ports of groups

Last iteration: Shift by N-1 the start of each consecutive ports of groups and measure the traffic loss for each port group.

5.3 Reporting Format

For each test the report MUST include:

- The port configuration including the number and location of ingress and egress ports located on the DUT

- If HOLB was observed in accordance with the HOLB test in section 5

- Percent of traffic loss

- The repeatability of the test needs to be indicated: number of iteration of the same test and percentage of variation between results (min, max, avg)

6. Incast Stateful and Stateless Traffic

6.1 Objective

The objective of this test is to measure the values for TCP Goodput [4] and latency with a mix of large and small flows. The test is

designed to simulate a mixed environment of stateful flows that require high rates of goodput and stateless flows that require low latency. Stateful flows are created by generating TCP traffic and, stateless flows are created using UDP type of traffic.

6.2 Methodology

In order to simulate the effects of stateless and stateful traffic on the DUT, there MUST be multiple ingress ports receiving traffic destined for the same egress port. There also MAY be a mix of stateful and stateless traffic arriving on a single ingress port. The simplest setup would be 2 ingress ports receiving traffic destined to the same egress port.

One ingress port MUST be maintaining a TCP connection trough the ingress port to a receiver connected to an egress port. Traffic in the TCP stream MUST be sent at the maximum rate allowed by the traffic generator. At the same time, the TCP traffic is flowing through the DUT the stateless traffic is sent destined to a receiver on the same egress port. The stateless traffic MUST be a microburst of 100% intensity.

It is RECOMMENDED that the ingress and egress ports are varied in multiple tests to measure the maximum microburst capacity.

The intensity of a microburst MAY be varied in order to obtain the microburst capacity at various ingress rates.

It is RECOMMENDED that all ports on the DUT be used in the test.

The tests described bellow have iterations called "first iteration", "second iteration" and, "last iteration". The idea is to show the first two iterations so the reader understands the logic on how to keep incrementing the iterations. The last iteration shows the end state of the variables.

For example:

Stateful Traffic port variation (TCP traffic):

TCP traffic needs to be generated in this section. During Iterations number of Egress ports MAY vary as well.

First Iteration: 1 Ingress port receiving stateful TCP traffic and 1 Ingress port receiving stateless traffic destined to 1 Egress Port

Second Iteration: 2 Ingress port receiving stateful TCP traffic and 1 Ingress port receiving stateless traffic destined to 1 Egress Port

Last Iteration: N-2 Ingress port receiving stateful TCP traffic and 1 Ingress port receiving stateless traffic destined to 1 Egress Port

Stateless Traffic port variation (UDP traffic):

UDP traffic needs to be generated for this test. During Iterations, the number of Egress ports MAY vary as well.

First Iteration: 1 Ingress port receiving stateful TCP traffic and 1 Ingress port receiving stateless traffic destined to 1 Egress Port

Second Iteration: 1 Ingress port receiving stateful TCP traffic and 2 Ingress port receiving stateless traffic destined to 1 Egress Port

Last Iteration: 1 Ingress port receiving stateful TCP traffic and N-2 Ingress port receiving stateless traffic destined to 1 Egress Port

6.3 Reporting Format

The report MUST include the following:

- Number of ingress and egress ports along with designation of stateful or stateless flow assignment.

- Stateful flow goodput

- Stateless flow latency

- The repeatability of the test needs to be indicated: number of iterations of the same test and percentage of variation between results (min, max, avg)

7. Security Considerations

Benchmarking activities as described in this memo are limited to technology characterization using controlled stimuli in a laboratory environment, with dedicated address space and the constraints specified in the sections above.

The benchmarking network topology will be an independent test setup and MUST NOT be connected to devices that may forward the test traffic into a production network, or misroute traffic to the test management network.

Further, benchmarking is performed on a "black-box" basis, relying solely on measurements observable external to the DUT/SUT.

Special capabilities SHOULD NOT exist in the DUT/SUT specifically for

benchmarking purposes. Any implications for network security arising from the DUT/SUT SHOULD be identical in the lab and in production networks.

8. IANA Considerations

NO IANA Action is requested at this time.

9. References

9.1. Normative References

[RFC1242] Bradner, S. "Benchmarking Terminology for Network Interconnection Devices", BCP 14, RFC 1242, DOI 10.17487/RFC1242, July 1991, <http://www.rfc-editor.org/info/rfc1242>

[RFC2544] Bradner, S. and J. McQuaid, "Benchmarking Methodology for Network Interconnect Devices", BCP 14, RFC 2544, DOI 10.17487/RFC2544, March 1999, <http://www.rfc-editor.org/info/rfc2544>

9.2. Informative References

[1] Avramov L. and Rapp J., "Data Center Benchmarking Terminology", April 2017.

[RFC2889] Mandeville R. and Perser J., "Benchmarking Methodology for LAN Switching Devices", RFC 2889, August 2000, <http://www.rfc-editor.org/info/rfc2889>

[RFC3918] Stopp D. and Hickman B., "Methodology for IP Multicast Benchmarking", RFC 3918, October 2004, <http://www.rfc-editor.org/info/rfc3918>

[RFC 6985] A. Morton, "IMIX Genome: Specification of Variable Packet Sizes for Additional Testing", RFC 6985, July 2013, <http://www.rfc-editor.org/info/rfc6985>

[4] Yanpei Chen, Rean Griffith, Junda Liu, Randy H. Katz, Anthony D. Joseph, "Understanding TCP Incast Throughput Collapse in Datacenter Networks, "http://yanpeichen.com/professional/usenixLoginIncastReady.pdf"

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

[RFC2432] Dubray, K., "Terminology for IP Multicast Benchmarking", BCP 14, RFC 2432, DOI 10.17487/RFC2432, October 1998, <http://www.rfc-editor.org/info/rfc2432>

9.2. Acknowledgements

The authors would like to thank Alfred Morton and Scott Bradner for their reviews and feedback.

Authors' Addresses

Lucien Avramov
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
United States
Phone: +1 408 774 9077
Email: lucien.avramov@gmail.com

Jacob Rapp
VMware
3401 Hillview Ave
Palo Alto, CA
United States
Phone: +1 650 857 3367
Email: jrapp@vmware.com