BMWG
Internet-Draft
Intended status: Informational
Expires: 27 April 2023

G. Fioccola
E. Vasilenko
P. Volpato
Huawei Technologies
L. Contreras
Telefonica

24 October 2022

Benchmarking Methodology for IPv6 Segment Routing
draft-vfv-bmwg-srv6-bench-meth-04

Abstract

This document defines a methodology for benchmarking Segment Routing (SR) performance for Segment Routing over IPv6 (SRv6). It builds upon [RFC2544], [RFC5180], [RFC5695] and [RFC8402].

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], RFC 8174 [RFC8174].

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Segment Routing (SR), defined in [RFC8402], leverages the source routing paradigm. The headend node steers a packet through an SR Policy [I-D.ietf-spring-segment-routing-policy], instantiated as an ordered list of segments. A segment, referred to by its Segment Identifier (SID), can have a semantic local to an SR node or global within an SR domain. SR supports per-flow explicit routing while

maintaining per-flow state only at the ingress nodes to the SR domain.

However, there is no standard method defined to compare and contrast the foundational SR packet forwarding capabilities of network devices. This document aims to extend the efforts of [RFC1242] and [RFC2544] to SR network.

The SR architecture can be instantiated on two data-plane: SR over MPLS (SR-MPLS) and SR over IPv6 (SRv6). This document is limited to SRv6.

This document is limited to Headend encapsulations (H.Encaps.xxx) and segment Endpoints (End, End.X). It is expected that future documents may cover the benchmarking of SRv6 applications with decapsulation (End.Dxxx), Binding (End.Bxxx), Fast ReRoute [I-D.ietf-rtgwg-segment-routing-ti-lfa], etc.

SR can be applied to the IPv6 architecture with a new type of routing header called the SR Header (SRH) [RFC8754]. An instruction is associated with a segment and encoded as an IPv6 address. An SRv6 segment is also called an SRv6 SID. An SR Policy is instantiated as an ordered list of SRv6 SIDs in the routing header. The active segment is indicated by the Destination Address (DA) of the packet.

SRv6 involves 3 types of forwarding plane operations:

- PUSH consists of the insertion of one or more segments on top of the incoming packet. It is the SRH attachment in the case of SRv6.

- NEXT consists of the inspection of the next segment. The active segment is completed and the next segment is activated. It is the copy of the next segment from the SRH to the destination address of the IPv6 header in SRv6.

- CONTINUE happens when the active segment is not completed; hence, it remains active. It is the plain IPv6 forwarding action of a regular IPv6 packet according to its destination address in SRv6.

SR list for PUSH operation is typically constructed by SR Policy in ingress node, see [I-D.ietf-spring-segment-routing-policy].

[RFC5695] describes a methodology specific to the benchmarking of MPLS forwarding devices, by considering the most common MPLS packet forwarding scenarios and corresponding performance measurements.

[RFC5180] provides benchmarking methodology recommendations that address IPv6-specific aspects, such as evaluating the forwarding performance of traffic containing extension headers.

The purpose of this document is to describe a methodology specific to the benchmarking of Segment Routing over IPv6. The methodology described is a complement for [RFC5180] and [RFC5695].

2. SRv6 Forwarding

An SRv6 SID is allocated in the form of an IPv6 address. For the IPv6 data plane, a new type of IPv6 Routing Extension Header, called Segment Routing Header (SRH) has been defined [RFC8754]. The SRH contains the Segment List as an ordered list of IPv6 addresses: each address in the list is a SID. A dedicated field, referred to as Segments Left, is used to maintain the pointer to the active SID of the Segment List.

There are three different categories of nodes that may be involved in segment routing networks.

The SR source node is the headend node and steers a packet into an SR Policy. It can be a host originating an IPv6 packet or an SR domain ingress router encapsulating a received packet into an outer IPv6 packet and inserts the SRH in the outer IPv6 header. It sets the first SID of the SR Policy as the IPv6 Destination Address of the packet.

The SR transit node forwards packets destined for a remote segment as a normal IPv6 packet based on the IPv6 destination address, because the IPv6 destination address does not locally match with a segment. Indeed, according to [RFC8200] the only node allowed to inspect the Routing Extension Header (and therefore the SRH) is the node corresponding to the destination address of the packet.

The SR segment endpoint node receives packets whose IPv6 destination address is locally configured as a segment. It creates Forwarding Information Base (FIB) entries for its local SIDs. For each SR packet, it inspects the SRH, may prepare some actions (like forwarding through a particular port), then replaces the IPv6 destination address with the new active segment.

The operations applied by the SRv6 packet processing are different at the SR source, transit, and SR segment endpoint nodes.

The processing of the SR source node corresponds to the sequence of insertion of the SRH, composed of SIDs stored in reverse order, and setting of the IPv6 Destination Address as the first SID of the SR Policy. It can be performed by encapsulating a packet into an outer IPv6 packet with an SRH.

The processing of the SR segment endpoint node corresponds to the detection of the new active segment, which is the next segment in the Segment List and the related modification of the IPv6 destination address of the outer IPv6 header. Then packets are forwarded on the basis of the IPv6 forwarding table.

The processing of the SR transit node corresponds to normal forwarding of the packets containing the SR header. In SRv6, the transit nodes do not need to be SRv6 aware, as every IPv6 router can act as an SRv6 transit node since any IPv6 node will maintain a plain IPv6 FIB entry for any prefix, no matter if the prefix represents a segment or not.

[I-D.ietf-spring-segment-routing-policy] specifies the concepts of SR Policy and steering into an SR Policy. The header of a packet steered in an SR Policy is augmented with the ordered list of segments associated with that SR Policy. SR Policy state is instantiated only on the headend node, which steers a flow into an SR Policy. Indeed intermediate and endpoint nodes do not require any state to be maintained. SR Policies can be instantiated on the headend dynamically and on demand basis. SR policy may be installed by PCEP [RFC8664], BGP [I-D.ietf-idr-segment-routing-te-policy], or via manual configuration on the router. PCEP and BGP signaling of SR Policies can be the case of a controller-based deployment.

In addition to the basic SRv6 packet processing, the SRv6 Network Programming model [RFC8986] describes a set of functions that can be associated to segments and executed in a given SRv6 node.

Examples of such functions are described in [RFC8986], but, in practice, any behavior and function can be associated with a local SID in a node, to apply any special processing on the packet. The definition of a standardized set of segment routing functions facilitates the deployment of SR domains with interoperable equipment from multiple vendors.

According to [RFC8986], 128 bit SID can be logically split into three fields and interpreted as LOCATOR:FUNCTION:ARGS (in short LOC:FUNCT:ARG) where LOC includes the L most significant bits, FUNCT the following F bits and ARG the remaining A bits, where L+F+A=128. The LOC corresponds to an IPv6 prefix (for example with a length of 48, 56, or 64 bits) that can be distributed by the routing protocols

and provides the reachability of a node that hosts some functions. All the different functions residing in a node have a different FUNCT code, so that their SIDs will be different. The ARG bits are used to provide information (arguments) to a function. From the routing point of view, the solution is scalable, as a single prefix is distributed for a node, which implements a potentially large number of functions and related arguments.

3. Test Methodology

3.1. Test Setup

The test setup in general is compliant with section 6 of [RFC2544] but augmented by the methodology specified in section 4 of [RFC5695] using many ports. It is needed to test the packet forwarding engine that may have different performance based on the number of ports served. The Device Under Test (DUT) may have oversubscribed ports, then traffic for such ports should be proportionally decreased according to the specific DUT oversubscription ratio. All ports served by a particular packet forwarding engine should be loaded in reverse proportion to the claimed oversubscription ratio. Tests SHOULD be done with bidirectional traffic that better reflects the real environment for SRv6 nodes. It is OPTIONAL to choose a non-equal proportion for upstream and downstream traffic for some specific aggregation nodes.

The RECOMMENDED topology for SRv6 Forwarding Benchmarking should be the same as MPLS and it is described in section 4 of [RFC5695]. Port numbers involved in the tests and their oversubscription ratio MUST be reported. This document is benchmarking only "source routing". Hence, SIDs represent only Headend encapsulation (H.Encaps.xxx) or segment Endpoint (End, End.X) that may be carried in IGP extensions. In general, Functions of the last SID (called "behavior" in [RFC8986]) may be used to encode services (similar to L2/L3 VPNs and much more) but it is out of the scope of this document.

It is OPTIONAL to test SRH in the combination with any other extension headers (fragmentation, hop-by-hop, destination options, etc.) but in all tests, the SRH header should be present for the test to be relevant for SRv6. It is RECOMMENDED to follow section 5.3 of [RFC5180] to introduce other extension headers in proportion 1%, 10%, 50% that may better reflect real use cases.

Segment Routing may also be implemented as a software network function in an NFV Infrastructure and, in this case, additional considerations should be done. [ETSI-GR-NFV-TST-007] describes test guidelines for NFV capabilities that require interactions between the components implementing NFV functionality.

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

3.2. Locator and Endpoint behaviors methods

As specified in [RFC8986], topological segments have the structure that consists of Locator and Endpoint behavior (End, End.X, etc), the latter may have a few different flavors (PSP, USP, USD).

It is RECOMMENDED that the DUT and test tool support at least one option for SID stack construction:

- IS-IS Extensions to Support Segment Routing over IPv6 Dataplane [I-D.ietf-lsr-isis-srv6-extensions]

- OSPFv3 Extensions for SRv6 [I-D.ietf-lsr-ospfv3-srv6-extensions]

- Segment Routing Policy Architecture [I-D.ietf-spring-segment-routing-policy].

A routing protocol (OSPF or ISIS) SHOULD be used for the construction of the simplest SRH with 1 SID. It is RECOMMENDED that SR policy should be used for the construction of SRH with 2 SIDs. It is possible to test longer SRH if there is an interest.

It is RECOMMENDED that the top SID on the list should have an End.X flavor type to emulate traffic engineering scenario. In all cases, SID stack configuration SHOULD happen before packet forwarding would be started. Control plane convergence speed is not the subject of the present tests.

The Locator and Endpoint construction method and SR policy construction method used MUST be reported according to Section 4.

3.3. Frame Formats and Sizes

SRv6 tests will use Frame characteristics similarly to section 4.1.5 of [RFC5695], except the need for a bigger MTU to accommodate SRH.

It is to be noted that [RFC5695] requires exactly a single entry in the MPLS label stack in an MPLS packet that is not enough to simulate a typical SR SID list. The number of entries in SRH MUST be reported.

According to section 4.1.4.2 of [RFC5695], the payload is RECOMMENDED to have an IP packet (IPv6 or IPv4 with UDP or TCP) to better represent the real environment.

It is assumed that the test would be for Ethernet media only. Other media is possible (see section 4.1.5.2 of [RFC5695] for the POS example). Some layer 2 technologies (like POS/PPP) have bit- or byte- stuffing then [RFC4814] may help to calculate real performance more accurately or else 1-2% error is expected. The most popular layer 2 technology for SR is Ethernet, it does not have stuffing.

RECOMMENDED frame sizes are presented below. Any other frame sizes may be added if suspected of abnormal behavior. For example, some architectures may allocate buffer memory in big fixed chunks that may drop performance if frame sizes are chosen just a few octet more than the fixed chunk size (the second chunk would have a very low memory utilization).

RECOMMENDED frame sizes are the following:

- Ethernet Minimal: 64+8+n*16

- DUT Minimal Wire Speed: 128-256 (it depends on the DUT specification)

- Ethernet Typical: 1518+8+n*16

- DUT Maximum Wire Speed: 9000

where n is the number of labels (SID Depth).

Note that 8 octets are added in the previous calculations for the SRH header itself. While n*16 octets are added to accommodate SID entries. The typical frame size values are listed above for the DUT minimal and maximum wire speed, but they can be modified according to the DUT characteristics. Indeed, the minimum wire speed frame size can be considered based on the DUT specification but, in some cases, many tests may be needed in the search for the real minimum wire speed frame size. VLAN tag may additionally increase the frame size. VLAN tag tests are OPTIONAL.

3.4. Protocol Addresses

IANA reserved an IPv6 address block 2001:0200::/48 for use with IPv6 benchmark testing (see section 8 of [RFC5180]). IPv6 source and destination addresses for the test streams SHOULD belong to the IPv6 range assigned by IANA. It is not principal what Locator blocks would be chosen for tests. It may be /52, /56, /64, or even bigger. It is possible to test a few different Locator blocks if there is a need.

As it is discussed in section 3.1, there is a need to load the whole forwarding engine (on all ports). [RFC4814] discusses the importance to have many flows with address randomization for acceptable hash-based load balancing that is implemented in all forwarding engines. In the context of this document, it may also be relevant for SIDs, because SIDs may be used for hash to choose the next link (depending on DUT default or desired configuration). It is important to check what exactly is used for the hash load balancing algorithm on the DUT to keep these numbers sufficiently random and at volume. It is very often that IP addresses and transport protocol ports are used instead of SIDs.

3.5. Trial Duration

The test portion of each trial must take into account the respective protocol configuration. IGP protocols typically have a shorter hold time, while some BGP default configurations may be up to 180 seconds. It is needed to check the default hold time of the DUT for the respective protocol used.

In general, the test portion of each trial SHOULD be no less than 250 seconds, which is a reasonable value based on common hold time values. But a test can also adapt to the real setup and select a different value if default configuration has been changed. The test portion of each trial can be chosen at least 10 seconds longer than the hold time to verify that the DUT can maintain a stable control plane when the data-forwarding plane is under stress.

3.6. Traffic Verification

Traffic verification is following section 4.1.8 of [RFC5695].

3.7. Buffer tests

Back-to-back test was initially discussed in section 26.4 [RFC2544] and later improved in [RFC9004] which is considered the comprehensive reference for Back-to-back test. Modern forwarding engines are typically flexible in the buffer distribution between different ports. Hence, like for all other benchmarking tests, it is important to stress the forwarding engine on all ports. It should be necessary to perform throughput tests first because only frame sizes that stress DUT below wire-speed can be used for back-to-back tests. Buffers would be filled with the rate equal to the difference between the theoretical maximum frame rate (wire-speed) and DUT measured throughput for the respective frame size.

The test time could be much shorter than recommended in [RFC9004] because typical SR DUT is hardware-based with claimed buffers between 30ms to 100ms. It is better to consult with the vendor to find a good starting search point. If DUT is software-based then [RFC9004] recommendation for 2-30 seconds is applied.

Queuing SHOULD NOT have weighted random early detection (WRED) or any other mechanism that may start dropping packets before the buffer is filled. Queuing SHOULD be configured for the tail drop which is typically a non-default configuration. Back-to-back test is rather complex and expensive (50 runs for every frame size). Hence, it is OPTIONAL for SRv6.

4. Reporting Format

There are a few parameters that must be changed in section 5 of [RFC5695] for SRv6 tests. New parameters that MUST be reported are:

* Port numbers involved in the tests and their respective oversubscription ratio.

* Upstream/downstream traffic proportion (equal bidirectional or some other split).

* SRv6 Forwarding Operations (PUSH/ NEXT/ CONTINUE).

* Number of Segments considered in the SRH and the type of behavior used (according to [RFC8986]).

* SR Policy construction method (PCEP, BGP, manual configuration).

* Type of the payload (IPv6/IPv4, UDP/TCP).

* Tested buffers size in frames with respective frame size (for the optional back-to-back test); it is possible to record calculated buffer time for wire-speed throughput.

Some parameters MAY be changed:

* Label Distribution protocol and IGP are the same in the context of SRv6. Hence, it is called "Locator and Endpoint behaviors methods".

* Port media type may be reported only one time for all tests if only Ethernet media would be tested

5. SRv6 Forwarding Benchmarking Tests

In general, tests are compliant with [RFC2544] but the important correction discussed in section 6 of [RFC2544] is applied: ports chosen for every test MUST stress all ports served by one forwarding engine. It is better to check the DUT specification for the relationship between ports and the forwarding engine to minimize the number of ports involved. But it is possible to understand the worst case by looking at the throughput and latency from the trial tests. If any doubt exists about how full is the offered load for the forwarding engine then it is better to stress all ports of the line card or all ports for the whole router with a centralized forwarding engine. A partial load on the forwarding engine would show optimistic results. Controllable traffic distribution between many ports (as specified in section 4 of [RFC5695]) would need separate SID announcements for separate ports.

As specified in section 6 of [RFC5695], the traffic is sent from test tool Tx port(s) to the DUT at a constant load for a fixed-time interval, and is received from the DUT on test tool Rx port(s). If any frame loss is detected, then a new iteration is needed where the offered load is decreased and the sender will transmit again. An iterative search algorithm MUST be used to determine the maximum offered frame rate with a zero frame loss (No-Drop Rate - NDR). Each iteration should involve varying the offered load of the traffic, while keeping the other parameters (test duration, number of ports, number of addresses, frame size, etc.) constant, until the maximum rate at which none of the offered frames are dropped is determined.

5.1. Throughput

This section contains a description of the tests that are related to the characterization of a DUT's SRv6 traffic forwarding throughput.

The list of segments for SRv6 is represented as a list of IPv6 addresses, included in the SRH. There are three distinct types of nodes that are involved in segment routing networks.

5.1.1. Throughput of a Source Node

Objective: To obtain the DUT's Throughput during the packet processing of a Source Node. It is when the Source SR node, which corresponds to the headend node, encapsulates a received packet into an outer IPv6 packet and inserts the SR Header (SRH) as a Routing Extension Header in the outer IPv6 header. The Segment List in the SRH is composed of SIDs and the Source SR node sets the first SID of the SR Policy as the IPv6 Destination Address of the packet.

Procedure: Similar to [RFC5695] with potential extension to test SID list longer than 1 SID (2 are RECOMMENDED, many are possible). The test tool must advertise and learn the IP prefix(es), as per Section 3.4, and must use one option for SID stack construction, as per Section 3.2, on its receive ports and transmit ports towards the DUT.

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.1.2. Throughput of a Segment Endpoint Node

Objective: To obtain the DUT's Throughput during the packet processing of a Segment Endpoint Node. It is when the SR Segment Endpoint node receives packets whose IPv6 destination address is locally configured as a segment. The SR Segment Endpoint node inspects the SR header: it detects the new active segment, i.e. the next segment in the Segment List, modifies the IPv6 destination address of the outer IPv6 header and forwards the packet on the basis of the IPv6 forwarding table.

Procedure: Similar to [RFC5695] with potential extension to test SID list longer than 1 SID (2 are RECOMMENDED, many are possible). The test tool must advertise and learn the IP prefix(es), as per Section 3.4, and must use one option for SID stack construction, as per Section 3.2, on its receive ports and transmit ports towards the DUT.

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.1.3. Throughput of a Transit Node

Objective: To obtain the DUT's Throughput during the packet processing of a Transit Node. It is when a Transit node forwards the packet containing the SR header as a normal IPv6 packet because the IPv6 destination address does not locally match with a segment.

Procedure: Similar to [RFC5695] with potential extension to test SID list longer than 1 SID (2 are RECOMMENDED, many are possible). The test tool must advertise and learn the IP prefix(es), as per Section 3.4, and must use one option for SID stack construction, as per Section 3.2, on its receive ports and transmit ports towards the DUT.

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.2. Buffers size

Back-to-back test is OPTIONAL and SHOULD be performed only after throughput tests because it SHOULD use only frame sizes that DUT is not capable to forward wire-speed, as further explained in a previous section.

Objective: To determine the buffer size as defined in section 6 of [RFC9004] for each of the SRv6 forwarding operations.

Procedure: Similar to [RFC9004] with 2 SIDs RECOMMENDED (many SIDs are possible).

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.3. Latency

Objective: To determine the latency as defined in section 6.2 of [RFC5695] for each of the SRv6 forwarding operations.

Procedure: Similar to [RFC5695] with potential extension to test SID list longer than 1 SID (2 are RECOMMENDED, many are possible). It is OPTIONAL to improve the procedure according to section 7.2 [RFC8219] measuring 500 frames in one test with calculations for typical and worst-case latency.

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.4. Frame Loss

Objective: To determine the frame-loss rate (as defined in section 6.3 of [RFC5695]) for each of the SRv6 forwarding operations of a DUT throughout the entire range of input data rates and frame sizes.

Procedure: Similar to [RFC5695] with potential extension to test SID list longer than 1 SID (2 are RECOMMENDED, many are possible).

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.5. System Recovery

Objective: To characterize the speed at which a DUT recovers from an overload condition for each of the SRv6 forwarding operations.

Procedure: Similar to section 6.4 of [RFC5695].

Reporting Format: Similar to [RFC5180] with the additional parameters specified in Section 4.

5.6. Reset

Objective: To characterize the speed at which a DUT recovers from a device or software reset for each of the SRv6 forwarding operations.

Procedure: Similar to section 4 of [RFC6201] where the types of resets considered are Hardware resets, Software resets and Power interruption. They have a different impact on the forwarding behavior of the device.

Reporting Format: Similar to [RFC6201] with the additional parameters specified in Section 4.

The Reset tests SHOULD be extended according to [RFC6201] in order to reset only part of the DUT: only line card reset, only process reset (for example ISIS), only one routing engine reset in the configuration with routing engine redundancy, full power interruption, partial power interruption, etc.

6. Security Considerations

Benchmarking methodologies are limited to technology characterization in a laboratory environment, with dedicated address space and constraints. 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 production networks. The benchmarking network topology is 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.

There are no specific security considerations within the scope of this document.

7. IANA Considerations

This document has no IANA actions.

8. Acknowledgements

The authors would like to thank Al Morton, Gabor Lencse, Boris Khasanov for the precious comments and suggestions.

9. References

9.1. Normative References

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

[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>.

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

[RFC4814] Newman, D. and T. Player, "Hash and Stuffing: Overlooked Factors in Network Device Benchmarking", RFC 4814, DOI 10.17487/RFC4814, March 2007, <https://www.rfc-editor.org/info/rfc4814>.

[RFC5180] Popoviciu, C., Hamza, A., Van de Velde, G., and D. Dugatkin, "IPv6 Benchmarking Methodology for Network Interconnect Devices", RFC 5180, DOI 10.17487/RFC5180, May 2008, <https://www.rfc-editor.org/info/rfc5180>.

[RFC5695] Akhter, A., Asati, R., and C. Pignataro, "MPLS Forwarding Benchmarking Methodology for IP Flows", RFC 5695, DOI 10.17487/RFC5695, November 2009, <https://www.rfc-editor.org/info/rfc5695>.

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

[RFC8219] Georgescu, M., Pislaru, L., and G. Lencse, "Benchmarking Methodology for IPv6 Transition Technologies", RFC 8219, DOI 10.17487/RFC8219, August 2017, <https://www.rfc-editor.org/info/rfc8219>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

9.2. Informative References

[ETSI-GR-NFV-TST-007] ETSI, "ETSI GR NFV-TST 007: Network Functions Virtualisation (NFV) Release 3; Testing; Guidelines on Interoperability Testing for MANO", 2020, <https://www.etsi.org/deliver/etsi_gr/NFV-TST/001_099/007/02.06.01_60/gr_nfv-tst007v020601p.pdf>.

[I-D.ietf-idr-segment-routing-te-policy] Previdi, S., Filsfils, C., Talaulikar, K., Mattes, P., Jain, D., and S. Lin, "Advertising Segment Routing Policies in BGP", Work in Progress, Internet-Draft, draft-ietf-idr-segment-routing-te-policy-20, 27 July 2022, <https://www.ietf.org/archive/id/draft-ietf-idr-segment-routing-te-policy-20.txt>.

[I-D.ietf-lsr-isis-srv6-extensions] Psenak, P., Filsfils, C., Bashandy, A., Decraene, B., and Z. Hu, "IS-IS Extensions to Support Segment Routing over IPv6 Dataplane", Work in Progress, Internet-Draft, draft-ietf-lsr-isis-srv6-extensions-18, 20 October 2021, <https://www.ietf.org/archive/id/draft-ietf-lsr-isis-srv6-extensions-18.txt>.

[I-D.ietf-lsr-ospfv3-srv6-extensions] Li, Z., Hu, Z., Talaulikar, K., and P. Psenak, "OSPFv3 Extensions for SRv6", Work in Progress, Internet-Draft, draft-ietf-lsr-ospfv3-srv6-extensions-08, 14 September 2022, <https://www.ietf.org/archive/id/draft-ietf-lsr-ospfv3-srv6-extensions-08.txt>.

[I-D.ietf-rtgwg-segment-routing-ti-lfa]
Litkowski, S., Bashandy, A., Filsfils, C., Francois, P., Decraene, B., and D. Voyer, "Topology Independent Fast Reroute using Segment Routing", Work in Progress, Internet-Draft, draft-ietf-rtgwg-segment-routing-ti-lfa-08, 21 January 2022, <https://www.ietf.org/archive/id/draft-ietf-rtgwg-segment-routing-ti-lfa-08.txt>.

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", Work in Progress, Internet-Draft, draft-ietf-spring-segment-routing-policy-22, 22 March 2022, <https://www.ietf.org/archive/id/draft-ietf-spring-segment-routing-policy-22.txt>.

[RFC6201] Asati, R., Pignataro, C., Calabria, F., and C. Olvera, "Device Reset Characterization", RFC 6201, DOI 10.17487/RFC6201, March 2011, <https://www.rfc-editor.org/info/rfc6201>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <https://www.rfc-editor.org/info/rfc8664>.

[RFC9004] Morton, A., "Updates for the Back-to-Back Frame Benchmark in RFC 2544", RFC 9004, DOI 10.17487/RFC9004, May 2021, <https://www.rfc-editor.org/info/rfc9004>.

Authors' Addresses

Giuseppe Fioccola
Huawei Technologies
Riesstrasse, 25
80992 Munich
Germany
Email: giuseppe.fioccola@huawei.com

Eduard Vasilenko
Huawei Technologies
17/4 Krylatskaya str.
Moscow
Email: vasilenko.eduard@huawei.com

Paolo Volpato
Huawei Technologies
Via Lorenteggio, 240
20147 Milan
Italy
Email: paolo.volpato@huawei.com

Luis Miguel Contreras Murillo
Telefonica
Spain
Email: luismiguel.contrerasmurillo@telefonica.com