Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: September 26, 2019

S. Aldrin
Google
C. Pignataro, Ed.
N. Kumar, Ed.
Cisco
N. Akiya
Big Switch Networks
R. Krishnan
A. Ghanwani
Dell

March 25, 2019

Service Function Chaining (SFC) Operation, Administration and Maintenance (OAM) Framework draft-ietf-sfc-oam-framework-06

Abstract

This document provides a reference framework for Operations, Administration and Maintenance (OAM) for Service Function Chaining (SFC).

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 September 26, 2019.

Table of Contents

1. Introduction

Service Function Chaining (SFC) enables the creation of composite services that consist of an ordered set of Service Functions (SF) that are to be applied to packets and/or frames selected as a result of classification [RFC7665]. Service Function Chaining is a concept that provides for more than just the application of an ordered set of SFs to selected traffic; rather, it describes a method for deploying SFs in a way that enables dynamic ordering and topological independence of those SFs as well as the exchange of metadata between participating entities. The foundations of SFC are described in the following documents:

- SFC Problem Statement [RFC7498]
- SFC Architecture [RFC7665]

The reader is assumed to be familiar with the material in these documents.

This document provides a reference framework for Operations, Administration and Maintenance (OAM, [RFC6291]) of SFC. Specifically, this document provides:

- In Section 2, an SFC layering model;
- In Section 3, aspects monitored by SFC OAM;
- In Section 4, functional requirements for SFC OAM;
- In Section 5, a gap analysis for SFC OAM.

1.1. Document Scope

The focus of this document is to provide an architectural framework for SFC OAM, particularly focused on the aspect of the Operations component within OAM. Actual solutions and mechanisms are outside the scope of this document.

2. SFC Layering Model

Multiple layers come into play for implementing the SFC. These include the service layer and the underlying layers (Network, Link etc)

- The service layer in Figure 1, consists of SFC data plane elements that includes classifiers, Service Functions (SF), Service Function Forwarders (SFF), SFC Proxy. This layer uses the overlay network for ensuring connectivity between SFC data plane elements.

- The overlay network layer in Figure 1, leverages various overlay network technologies interconnecting SFC data plane elements and allows establishing service function paths (SFPs). This layer is mostly transparent to the SFC data plane elements.

- The underlay network layer in Figure 1, is dictated by the networking technology deployed within a network (e.g., IP, MPLS)

- The link layer in Figure 1, is dependent upon the physical technology used. Ethernet is a popular choice for this layer, but other alternatives are deployed (e.g. POS, DWDM etc...). The same or distinct link layer technologies may be used in each leg shown in figure 1.

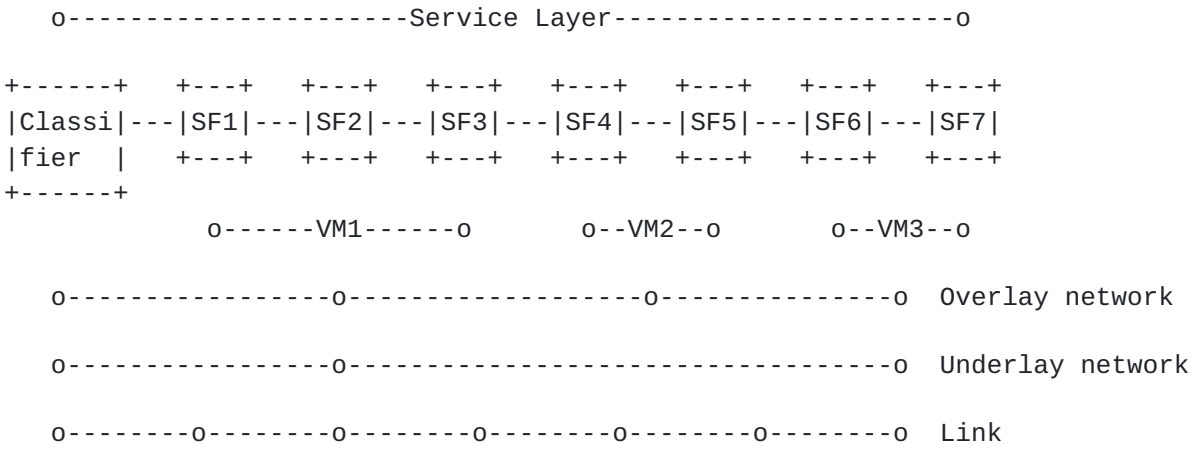

Figure 1: SFC Layering Example

While Figure 1 depicts a sample example where SFs are enabled as virtual entities, the SFC architecture does not make any assumptions on how SFC data plane elements are deployed. The SFC architecture is flexible to accomodate physical or virtual entity deployment. SFC OAM adheres to this flexibility and accordingly it is applicable whether SFC data plane elements are deployed directly on physical

hardware, as one or more Virtual Machines, or any combination thereof.

3. SFC OAM Components

The SFC operates at the service layer. For the purpose of defining the OAM framework, the service layer is broken up into three distinct components.

1. SF component: OAM solutions for this component include testing the service functions from any SFC-aware network devices (i.e. classifiers, controllers, other service nodes).

2. SFC component: OAM solutions for this component include testing the service function chains and the SFPs, validate the correlation between a Service Function Chain and the actual forwarding path followed by a packet matching that SFC, etc.

3. Classifier component: OAM solutions for this component include testing the validity of the classification rules and detecting any incoherence among the rules installed in different classifiers.

Below figure illustrates an example where OAM for the three defined components are used within the SFC environment.

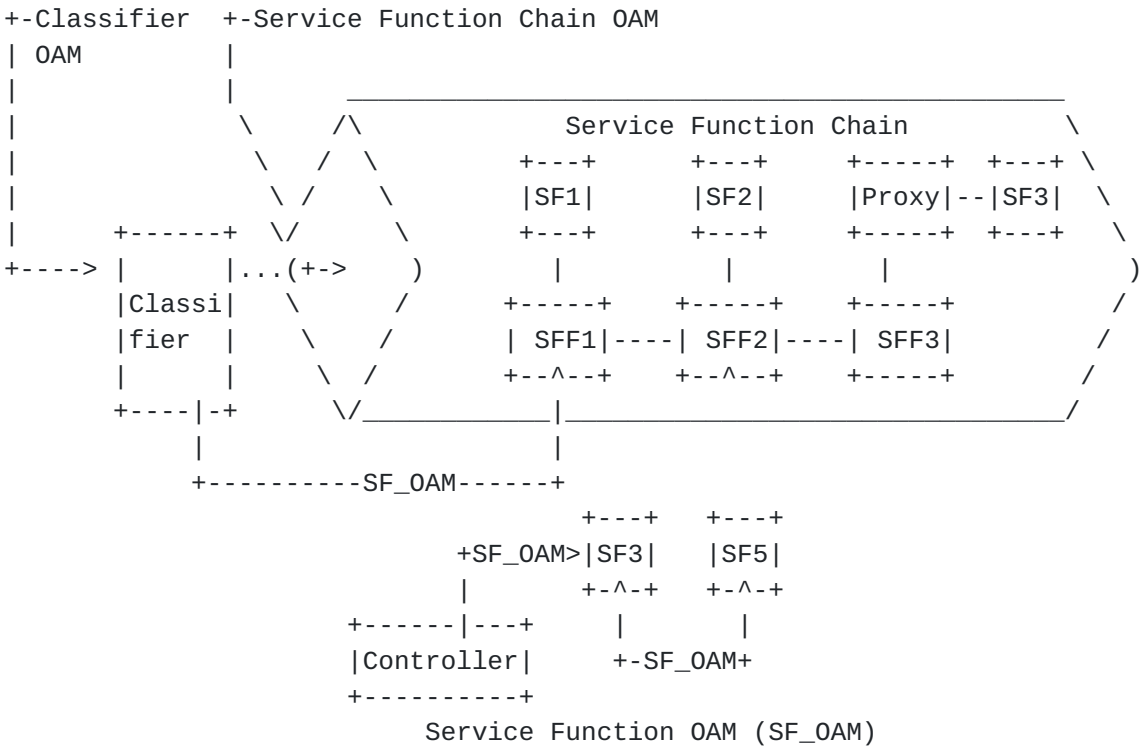

Figure 2: SFC OAM for Three Components

It is expected that multiple SFC OAM solutions will be defined, many targeting one specific component of the service layer. However, it is critical that SFC OAM solutions together provide the coverage of all three SFC OAM components: the service function component, the service function chain component and the classifier component.

3.1. Service Function Component

3.1.1. Service Function Availability

One SFC OAM requirement for the service function component is to allow an SFC-aware network device to check the availability to a specific service function, located on the same or different network devices. Service function availability is an aspect which raises an interesting question. How to determine that a service function is available?. On one end of the spectrum, one might argue that a service function is sufficiently available if the service node (physical or virtual) hosting the service function is available and is functional. On the other end of the spectrum, one might argue that the service function availability can only be concluded if the packet, after passing through the service function, was examined and verified that the packet got expected service applied.

The former approach will likely not provide sufficient confidence to the actual service function availability, i.e. a service node and a service function are two different entities. The latter approach is capable of providing an extensive verification, but comes with a cost. Some service functions make direct modifications to packets, while other service functions do not make any modifications to packets. Additionally, purpose of some service functions is to, conditionally, drop packets intentionally. In such case, packets will not be coming out from the service function. The fact is that there are many flavors of service functions available, and many more flavors of service functions will likely be introduced in future. Even a given service function may introduce a new functionality within a service function (e.g., a new signature in a firewall). The cost of this approach is that verifier functions will need to be continuously modified to "keep up" with new services coming out: lack of extendibility.

This framework document provides a RECOMMENDED architectural model where generalized approach is taken to verify that a service function is sufficiently available. More specifics on the mechanism to characterize SF-specific OAM to validate the service offering is outside the scope of this document. Those mechanism are implementation and deployment specific.

3.1.2. Service Function Performance Measurement

Second SFC OAM requirement for the service function component is to allow an SFC aware network device to check the loss and delay induced by a specific service function. The performance can be a passive measurement by using live traffic or can be active measurement by using synthetic probe packets.

On one hand, the performance of any specific service function can be measured by measuring the loss and delay metric of the traffic from service node to the respective service function, while on the other hand, the performance can be measured by leveraging the loss and delay metrics from the respective service functions. The latter requires service function involvement to perform the measurement while the former does not.

3.2. Service Function Chain Component

3.2.1. Service Function Chain Availability

Verifying an SFC is a complicated process as the SFC could be comprised of varying SF's. Thus, SFC requires the OAM layer to perform validation and verification of SF's within an SFP, as well as connectivity and fault isolation.

In order to perform service connectivity verification of an SFC, the OAM could be initiated from any SFC aware network devices for end-to-end paths or partial path terminating on a specific SF within the SFC. The goal of this OAM function is to ensure the SF's chained together has connectivity as it is intended to when SFC was established. Necessary return code should be defined to be sent back in the response to OAM packet, in order to qualify the verification.

When ECMP is in use at the service layer for any given SFC, there must be the ability to discover and traverse all available paths.

Detailed explanation on the mechanism is outside the scope of this document and will be expected to be included in the actual solution document.

3.2.2. Service Function Chain Performance Measurement

Any SFC-aware network device must have the ability to perform loss and delay measurements over the service function chain as a unit (i.e. end-to-end) or to a specific segment of service function through the SFC.

3.3. Classifier Component

A classifier maintains the classification rules that maps a flow to a specific SFC. It is vital that the classifier is correctly configured with updated classification rules and functioning accordingly. The SFC OAM must be able to validate the classification rules by assessing whether a flow is appropriately mapped to the relevant SFC. Sample OAM packets can be presented to the classifiers to assess the behavior with regards to a given classification entry.

4. SFC OAM Functions

Section 3 describes SFC OAM operations that is required on each SFC component. This section explores the same from the OAM functionality point of view, which many will be applicable to multiple SFC components.

Various SFC OAM requirements listed in Section 3, provides the need for various OAM functions at different layers. Many of the OAM functions at different layers are already defined and in existence. In order to apply such OAM functions at service layer, they have to be enhanced to operate a single SF/SFF to multiple SFs/SFFs in an SFC and also in multiple SFCs.

4.1. Connectivity Functions

Connectivity is mainly an on-demand function to verify that the connectivity exists between network elements and the availability exists to service functions. Ping is a common tool used to perform this function. OAM messages SHOULD be encapsulated with necessary SFC header and with OAM markings when testing the service function chain component. OAM messages MAY be encapsulated with necessary SFC header and with OAM markings when testing the service function component. Some of the OAM functions performed by connectivity functions are as follows:

- Verify the Path MTU from a source to the destination SF or through the SFC. This requires the ability for OAM packet to take variable length packet size.

- Verify the packet re-ordering and corruption.

- Verify the policy of an SFC or SF using OAM packet.

- Verification and validating forwarding paths.

- Proactively test alternate or protected paths to ensure reliability of network configurations.

4.2. Continuity Functions

Continuity is a model where OAM messages are sent periodically to validate or verify the reachability to a given SF or through a given SFC. This allows monitor network device to quickly detect failures like link failures, network failures, service function outages or service function chain outages. BFD is one such function which helps in detecting failures quickly. OAM functions supported by continuity check are as follows:

- Ability to provision continuity check to a given SF or through a given SFC.

- Notifying the failure upon failure detection for other OAM functions to take appropriate action.

4.3. Trace Functions

Tracing is an important OAM function that allows the operation to trigger an action (e.g., response generation) from every transit device (e.g., SFF, SF, SFC Proxy etc) on the tested layer. This function is typically useful to gather information from every transit

devices or to isolate the failure point towards an SF or through an SFC. Some of the OAM functions supported by trace functions are:

- Ability to trigger action from every transit device on the tested layer towards an SF or through an SFC, using TTL or other means.
- Ability to trigger every transit device to generate response with OAM code(s) on the tested layer towards an SF or through an SFC, using TTL or other means.
- Ability to discover and traverse ECMP paths within an SFC.
- Ability to skip un-supported SFs while tracing SFs in an SFC.

4.4. Performance Measurement Function

Performance management functions involve measuring of packet loss, delay, delay variance, etc. These measurements could be measured pro-actively and on-demand.

SFC OAM framework should provide the ability to perform packet loss for an SFC. Measuring packet loss is very important function. Using on-demand function, the packet loss could be measured using statistical means. Using OAM packets, the approximation of packet loss for a given SFC could be measured.

Delay within an SFC could be measured from the time it takes for a packet to traverse the SFC from ingress SFC node to egress SFF. As the SFCs are generally unidirectional in nature, measurement of one-way delay [RFC7679] is important. In order to measure one-way delay, time synchronization must be supported by means of NTP, PTP, GPS, etc.

One-way delay variation [RFC3393] could also be measured by sending OAM packets and measuring the jitter between the packets passing through an SFC.

Some of the OAM functions supported by the performance measurement functions are:

- Ability to measure the packet processing delay induced by a service function or the one-way delay to traverse a service function path along an SFC.
- Ability to measure the packet loss [RFC7680] within a service function or a service function path bound to a given SFC.

5. Gap Analysis

This section identifies various OAM functions available at different levels. It also identifies various gaps, if not all, existing within the existing toolset, to perform OAM function required for SFC.

5.1. Existing OAM Functions

There are various OAM tool sets available to perform OAM functions within various layers. These OAM functions could validate some of the underlay and overlay networks. Tools like ping and trace are in existence to perform connectivity check and tracing intermediate hops in a network. These tools support different network types like IP, MPLS, TRILL etc. There is also an effort to extend the tool set to provide connectivity and continuity checks within overlay networks. BFD is another tool which helps in detecting data forwarding failures. The following table is not exhaustive.

Layer	Connectivity	Continuity	Trace	Performance
Underlay N/w	Ping	E-OAM, BFD	Trace	IPPM, MPLS
Overlay N/w	Ping	BFD, NVo3	Trace	IPPM
SF	None	+ None	+ None	+ None
SFC	None	+ None	+ None	+ None

Table 3: OAM Tool GAP Analysis

Layer	Configuration	Orchestration	Topology	Notification
Underlay N/w	CLI, Netconf	CLI, Netconf	SNMP	SNMP, Syslog
Overlay N/w	CLI, Netconf	CLI, Netconf	SNMP	SNMP, Syslog
SF	CLI, Netconf	+ CLI	+ None	+ None
SFC	CLI, Netconf	+ CLI	+ None	+ None

Table 4: OAM Tool GAP Analysis (contd.)

5.2. Missing OAM Functions

As shown in Table 3, OAM functions for SFC are not standardized yet. Hence, there are no standard based tools available to verify SF and SFC.

5.3. Required OAM Functions

Primary OAM functions exist for underlying layers. Tools like ping, trace, BFD, etc., exist in order to perform these OAM functions. Configuration, orchestration and manageability of SF and SFC could be performed using CLI, NETCONF, etc.

As depicted in Table 3 and 4, for configuration, manageability and orchestration, providing data and information models for SFC is very much needed. With virtualized SF and SFC, manageability of these functions has to be done programmatically.

6. SFC OAM Model

This section describes the operational aspects of SFC OAM at the Service layer to perform the SFC OAM function defined in Section 4 and analyze the applicability of various existing OAM toolsets in the service layer.

6.1. SFC OAM Packet Marker

SFC OAM function described in Section 4 performed at the service layer or overlay network layer must mark the packet as OAM packet so that relevant nodes can differentiate an OAM packet from data packets. The base header defined in Section 2.2 of [RFC8300] assigns a bit to indicate OAM packets. When NSH encapsulation is used at the service layer, the O bit must be set to differentiate the OAM packet. Any other overlay encapsulations used in future must have a way to mark the packet as OAM packet.

6.2. OAM Packet Processing and Forwarding Semantic

Upon receiving OAM packet, an SFC-aware SFs may choose to discard the packet if it does not support OAM functionality or if the local policy prevent it from processing OAM packet. When SF supports OAM functionality, it is desired to process the packet and respond back accordingly that helps with end-to-end verification. To avoid hitting any performance impact, SFC-aware SFs can rate limit the number of OAM packets processed.

Service Function Forwarder (SFF) may choose not to forward the OAM packet to an SF if the SF does not support OAM function or if the

policy does not allow to forward OAM packet to an SF. SFF may choose to skip the SF, modify the header and forward to next SFC node in the chain. Although, skipping an SF might have implication on some OAM function (e.g., delay measurement may not be accurate). How SFF detects if the connected SF supports or allowed to process OAM packet is outside the scope of this document. It could be a configuration parameter instructed by the controller or can be a dynamic negotiation between SF and SFF.

If the SFF receiving the OAM packet bound to a given SFC is the last SFF in the chain, it must send a relevant response to the initiator of the OAM packet. Depending on the type of OAM solution and tool set used, the response could be a simple response (ICMP reply or BFD reply packet) or could include additional data from the received OAM packet (like stats data consolidated along the path). The proposed solution should detail it further.

Any SFC-aware node that initiates OAM packet must set the OAM marker in the overlay encapsulation.

6.3. OAM Function Types

As described in Section 4, there are different OAM functions that may require different OAM solutions. While the presence of OAM marker in the overlay header (e.g., O bit in the NSH header) indicates it as OAM packet, it is not sufficient to indicate what OAM function the packet is intended for. The Next Protocol field in NSH header may be used to indicate what OAM function is it intended to or what toolset is used.

6.4. OAM Toolset applicability

As described in Section 5.1, there are different tool sets available to perform OAM functions at different layers. This section describes the applicability of some of the available toolsets in the service layer.

6.4.1. ICMP Applicability

[RFC0792] and [RFC4443] describes the use of ICMP in IPv4 and IPv6 network respectively. It explains how ICMP messages can be used to test the network reachability between different end points and perform basic network diagnostics.

ICMP could be leveraged for basic OAM functions like SF availability or SFC availability. The Initiator can generate ICMP echo request message and control the service layer encapsulation header to get the response from relevant node. For example, a classifier initiating

OAM can generate ICMP echo request message, can set the TTL field in NSH header to 255 to get the response from last SFF and thereby test the SFC availability. Alternately, the initiator can set the TTL to other value to get the response from specific SFs and there by test partial SFC availability. Alternately, the initiator could send OAM packets with sequentially incrementing the TTL in NSH header to trace the SFP.

It could be observed that ICMP at its current stage may not be able to perform all required SFC OAM functions, but as explained above, it can be used for basic OAM functions.

6.4.2. Seamless BFD Applicability

[RFC5880] defines Bidirectional Forwarding Detection (BFD) mechanism for fast failure detection. [RFC5881] and [RFC5884] defines the applicability of BFD in IPv4, IPv6 and MPLS networks. [RFC7880] defines Seamless BFD (S-BFD), a simplified mechanism of using BFD. [RFC7881] explains its applicability in IPv4, IPv6 and MPLS network.

S-BFD could be leveraged to perform SF or SFC availability. An initiator could generate BFD control packet and set the "Your Discriminator" value as last SFF in the control packet. Upon receiving the control packet, last SFF will reply back with relevant DIAG code. We could also use the TTL field in the NSH header to perform partial SFC availability. For example, the initiator can set the "Your Discriminator" value to the SF that is intended to be tested and set the TTL field in NSH header in a way that it will be expired on the relevant SF. How the initiator gets the Discriminator value of the SF is outside the scope of this document.

6.4.3. In-Situ OAM

[I-D.ietf-sfc-proof-of-transit] defines a mechanism to perform proof of transit to securely verify if a packet traversed the relevant path or chain. While the mechanism is defined inband (i.e, it will be included in data packets), it can be used to perform various SFC OAM functions as well.

In-Situ OAM could be used with O bit set and perform SF availability, SFC availability of performance measurement.

6.4.4. SFC Traceroute

[I-D.penno-sfc-trace] defines a protocol that checks for path liveliness and trace the service hops in any SFP. Section 3 of [I-D.penno-sfc-trace] defines the SFC trace packet format while

section 4 and 5 of [I-D.penno-sfc-trace] defines the behavior of SF and SFF respectively.

An initiator can control the SIL in SFC trace packet to perform SF and SFC availability test.

6.5. Security Considerations

SFC and SF OAM must provide mechanisms for:

- Preventing usage of OAM channel for DDOS attacks.
- OAM packets meant for a given SFC should not get leaked beyond that SFC.
- Prevent OAM packets to leak the information of an SFC beyond its administrative domain.

6.6. IANA Considerations

No action is required by IANA for this document.

6.7. Acknowledgements

We would like to thank Mohamed Boucadair for his review and comments.

7. References

7.1. Normative References

[RFC0792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, DOI 10.17487/RFC0792, September 1981, <https://www.rfc-editor.org/info/rfc792>.

[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>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <https://www.rfc-editor.org/info/rfc4443>.

[RFC7498] Quinn, P., Ed. and T. Nadeau, Ed., "Problem Statement for Service Function Chaining", RFC 7498, DOI 10.17487/RFC7498, April 2015, <https://www.rfc-editor.org/info/rfc7498>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

7.2. Informative References

[I-D.ietf-sfc-proof-of-transit] Brockners, F., Bhandari, S., Dara, S., Pignataro, C., Leddy, J., Youell, S., Mozes, D., Mizrahi, T., Aguado, A., and D. Lopez, "Proof of Transit", draft-ietf-sfc-proof-of-transit-02 (work in progress), March 2019.

[I-D.penno-sfc-trace] Penno, R., Quinn, P., Pignataro, C., and D. Zhou, "Services Function Chaining Traceroute", draft-penno-sfc-trace-03 (work in progress), September 2015.

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

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC5881] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD) for IPv4 and IPv6 (Single Hop)", RFC 5881, DOI 10.17487/RFC5881, June 2010, <https://www.rfc-editor.org/info/rfc5881>.

[RFC5884] Aggarwal, R., Kompella, K., Nadeau, T., and G. Swallow, "Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)", RFC 5884, DOI 10.17487/RFC5884, June 2010, <https://www.rfc-editor.org/info/rfc5884>.

[RFC6291] Andersson, L., van Helvoort, H., Bonica, R., Romascanu, D., and S. Mansfield, "Guidelines for the Use of the "OAM" Acronym in the IETF", BCP 161, RFC 6291, DOI 10.17487/RFC6291, June 2011, <https://www.rfc-editor.org/info/rfc6291>.

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

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

[RFC7880] Pignataro, C., Ward, D., Akiya, N., Bhatia, M., and S. Pallagatti, "Seamless Bidirectional Forwarding Detection (S-BFD)", RFC 7880, DOI 10.17487/RFC7880, July 2016, <https://www.rfc-editor.org/info/rfc7880>.

[RFC7881] Pignataro, C., Ward, D., and N. Akiya, "Seamless Bidirectional Forwarding Detection (S-BFD) for IPv4, IPv6, and MPLS", RFC 7881, DOI 10.17487/RFC7881, July 2016, <https://www.rfc-editor.org/info/rfc7881>.

[RFC8029] Kompella, K., Swallow, G., Pignataro, C., Ed., Kumar, N., Aldrin, S., and M. Chen, "Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures", RFC 8029, DOI 10.17487/RFC8029, March 2017, <https://www.rfc-editor.org/info/rfc8029>.

Authors' Addresses

Sam K. Aldrin
Google

Email: aldrin.ietf@gmail.com

Carlos Pignataro (editor)
Cisco Systems, Inc.

Email: cpignata@cisco.com

Nagendra Kumar (editor)
Cisco Systems, Inc.

Email: naikumar@cisco.com

Nobo Akiya
Big Switch Networks

Email: nobo.akiya.dev@gmail.com

Ram Krishnan
Dell

Email: ramkri123@gmail.com

Anoop Ghanwani
Dell

Email: anoop@alumni.duke.edu