

IRTF
Internet-Draft
Intended status: Informational
Expires: 6 August 2022

M. Boucadair
Orange
D. Trossen
Huawei
A. Farrel
Old Dog Consulting
2 February 2022

Considerations for the use of SDN in Semantic Routing Networks
draft-boucadair-irtf-sdn-and-semantic-routing-00

Abstract

Semantic Routing is the process of making routing and forwarding decisions based, not only on the destination IP address, but on other information carried in an IP packet. The intent is to facilitate enhanced routing decisions based on this information in order to provide differentiated forwarding paths for specific packet flows.

Software Defined Networking (SDN) places control of network elements (including all or some of their forwarding decisions) within external software components called controllers and orchestrators. This approach differs from conventional approaches that solely rely upon distributed routing protocols for the delivery of advanced connectivity services. By doing so, SDN aims to enable network elements to be simplified while still performing (some high level) forwarding function.

This document examines the applicability of SDN techniques to Semantic Routing and provides considerations for the development of Semantic Routing solutions in the context of SDN.

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."

Internet-Draft

SDN and Semantic Routing

February 2022

This Internet-Draft will expire on 6 August 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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 Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

1.	Introduction	2
2.	Software Defined Networking (SDN): An Overview	3
3.	Semantic Routing: Summary of Required Technical Elements . .	5
4.	Programmable Forwarding	5
4.1.	Motivation	6
4.2.	SDN for Semantic Routing: The Intended Behavior	8
5.	Policy-Based Semantic Routing	10
6.	Network-Wide Coordination	10
7.	Applying Semantic Information to Packets	10
8.	Benefits and Concerns with the Use of SDN for Semantic Routing	11
9.	Security Considerations	12
10.	IANA Considerations	12
11.	Acknowledgements	12
12.	Contributors	12
13.	Informative References	12
	Authors' Addresses	15

[1.](#) Introduction

Service differentiation in the network can be enforced by manipulating a set of parameters that belong to distinct dimensions (e.g., forwarding, routing, traffic classification, resource partitioning). Among the techniques to achieve such differentiation, this document focuses on Semantic Routing, which refers to a process

that is meant to provide differentiated forwarding paths for specific packet flows distinct from simple shortest path first routing and, thus, satisfy specific service/application requirements.

More concretely, Semantic Routing is the process of making routing and forwarding decisions based, not only on the destination IP address of a packet, but also by taking into account other information that is carried in the packet such as (but not limited to):

- * Other fields of the IP header, e.g., DSCP/Traffic Class.
- * The transport header, e.g., transport port numbers [[RFC7597](#)] or subflows [[RFC8803](#)].
- * Specific transport encapsulation shims, e.g., [[RFC8926](#)].
- * Specific service headers, e.g., [[RFC8300](#)].
- * Specific metadata.

[Section 3](#) provides more details about Semantic Routing.

Software Defined Networking (SDN) places (partial or full) control of network elements and their forwarding decisions within dedicated software components called controllers and orchestrators. This approach differs from those that solely rely upon distributed routing protocols. An ambition of SDN is to enable network elements to be simplified while the network is optimized to deliver value-added connectivity services. Refer to [Section 2](#) for an overview of SDN.

This document examines the applicability of SDN to Semantic Routing ([Section 4](#)) and provides considerations for the development of Semantic Routing solutions in the context of SDN.

This version of the document does not elaborate on specific SDN protocols.

[2.](#) Software Defined Networking (SDN): An Overview

SDN refers to an approach for network programmability, that is, the capacity to initialize, control, and manage network behavior dynamically via open interfaces. Such programmability should facilitate the delivery of services in a deterministic, dynamic, and scalable manner.

SDN emphasizes the role of software in running networks by endorsing the separation between data and control planes. Even if such a separation has been adopted by most routing processes for decades ([Section 2.1 of \[RFC7149\]](#)), SDN focuses more on the power of "central" controllers to optimize route computation within a network before populating the Forwarding Information Base (FIB) of involved network elements.

The separation between control and data planes allows faster innovation in both planes, and enables a dynamic and flexible approach to implementing new network behaviors and reacting to changes in network state and traffic demands.

SDN has been discussed in many places during the last decade. For example, within the IRTF, [\[RFC7426\]](#) provides a concise reference for the SDN research community to address the questions of what SDN is, what the layer structure of an SDN architecture is, and how layers interface with each other within that architecture. [\[RFC7149\]](#) (published in the IETF stream) offers a service provider's perspective of the SDN landscape by describing requirements, issues, and other considerations about SDN. In particular, [\[RFC7149\]](#) classifies SDN techniques into the following functional domains:

- * Techniques for the dynamic discovery of network topology, devices, and capabilities, along with relevant information and data models that are meant to precisely document such topology, devices, and their capabilities.
- * Techniques for exposing network services and their characteristics

and for dynamically capturing the set of service parameters that will be used to measure the level of quality associated with the delivery of a given service or a combination thereof.

- * Techniques used by service-requirement-derived dynamic resource allocation and policy enforcement schemes, so that networks can be programmed accordingly.
- * Dynamic feedback mechanisms that are meant to assess how efficiently a set of policies are enforced from a service fulfillment and assurance perspective.

SDN can be deployed following a recursive model that involves dedicated interfaces for both network and service optimization. Indeed, [[RFC8597](#)] differentiates the control functions associated with transport from those related to services in an approach called Cooperating Layered Architecture for Software-Defined Networking (CLAS).

In an SDN context, domain-specific controllers can be deployed with specific interactions between them as discussed in [Section 4 of \[RFC8309\]](#).

[3.](#) Semantic Routing: Summary of Required Technical Elements

As described in [[I-D.farrel-irtf-introduction-to-semantic-routing](#)], Semantic Routing is the process of achieving enhanced routing decisions based on semantics added to IP headers to provide differentiated paths for different packet flows distinct from simple shortest path first routing. The additional information or "semantics" may be placed in existing header fields (such as the IPv6 Traffic Class field or the destination address) or may be achieved by adding fields to the header. Furthermore, it may be encoded in the payload or additional headers (such as in the port number fields or in an IPv6 Extension Header).

The application of Semantic Routing allows packets from different flows (even between the same applications on the same devices) to be marked for different treatment in the network. The packets may then be routed onto different paths according to the capabilities and states of the network links in order to meet the requirements of the

flows. For example, one flow may need low latency, while another may require ultra low jitter, and a third may demand very high bandwidth.

Three elements are needed to achieve Semantic Routing:

- * The capabilities and state of the network must be discovered.
- * The packets must be marked (with semantic information) according to their required delivery characteristics.
- * The routers must be programmed to forward the traffic according to how the packets are marked.

All these elements can be matched to the SDN functional domains listed in [Section 2](#). From that standpoint, this document provides more details on how SDN can be used to satisfy specific Semantic Routing needs.

[4.](#) Programmable Forwarding

Programmable Forwarding is the term applied to the use of control techniques to instruct network devices how to forward packets in a programmatic way.

[4.1.](#) Motivation

Modern networks are designed to carry traffic that belongs to a variety of services/applications that have distinct traffic performance requirements, reliability and robustness expectations, and service-specific needs [[RFC7665](#)][RFC8517]. Such expectations, and other forwarding requirements that can be captured in a Service Level Agreement (SLA) [[RFC7297](#)], can be considered by providers when designing their networks in order to be able to deliver differentiated forwarding behaviors. However, conventional routing and forwarding procedures do not always offer the required functionalities for such differentiated service delivery. Thus, additional means have to be enabled in these networks for the sake of innovative service delivery while minimizing the induced complexity to operate such networks. Also, these means should be tweaked to

ensure consistent forwarding behaviors network-wide.

The aforementioned means are not only extensions to routing protocols, but include other mechanisms that affect the forwarding behaviors within a network. An non-exhaustive list of sample capabilities that can be offered by appropriately controlling forwarding elements is provided below:

Resource Pooling: A network may host dedicated functions that implement resource pooling among many available paths or control which path is used to steer traffic as a function of the observed RTT (e.g., enable MPTCP converters [[RFC8803](#)] in specific network segments, including data centers as detailed in [Section 2.1 of \[RFC8041\]](#)).

There is a need to interact with the underlying forwarding elements to communicate a set forwarding policies that will ensure that such a differentiated service is provided to the specific flows. These forwarding policies include, for example, a set of rules that characterize the flows that are eligible to the resource pooling service or the scheduling policies (maximize link utilization, grab extra resources only when needed, etc.).

These policies are then enforced by programmable forwarders.

Performance-based Route Selection: Some applications may have strict traffic performance requirements (e.g., a low one-way delay [[RFC7679](#)]), however the underlying network elements may not support a mechanism to disseminate performance metrics associated with specific paths and/or perform performance-based route selection (e.g., [[I-D.ietf-idr-performance-routing](#)]).

As an alternative, an off-line Semantic Routing approach can be used to collect measurement data to reach a given content (e.g., one-way delay to reach specific data centers), perform route selection based on this data, and then program the appropriate forwarding elements accordingly.

Energy-efficient Forwarding: An important effort was made in the past to optimize the energy consumption of network elements.

However, such optimization is node-specific and no standard means are supported to optimize the energy consumption at the scale of the network. For example, many nodes (also, service cards) are deployed as backups.

A controller-based approach can be implemented so that the route selection process optimizes the overall energy consumption of a path. Such a process takes into account the current load, avoids waking nodes/cards for handling "few" traffic (i.e., minor portion of traffic), considers node-specific data (e.g., [\[RFC7460\]](#)), etc. This off-line Semantic Routing approach will transition specific cards/nodes to "idle", wake them, etc., without breaking service objectives. Moreover, such an approach will have to maintain an up-to-date topology even if a node is in an "idle" state (such nodes may be removed from adjacency tables if they don't participate to routing advertisements).

Network Partitioning: In order to rationalize the delivery of advanced connectivity services, a network may need to be partitioned in order to address specific forwarding requirements of groups of services/applications. Network slicing [\[I-D.ietf-teas-ietf-network-slices\]](#) can be considered to deliver these services. However, an intelligence is needed to decide the criteria to be used to partition the available resources, filter them, decide whether network extensions are needed, ensure whether/how resource preemption is adequately implemented, etc.

These tasks are better achieved using a central intelligence that has direct visibility into the intents of applications, underlying network capabilities, local policies and guidelines, etc. As an output of processing these various inputs, a set of node-specific policies are generated, and then pushed using available SDN interface.

Alternative Forwarding: The programmability of SDN in the form of forwarding actions defined on packet header fields allows for realizing forwarding techniques beyond the typical longest-prefix match used for IP-based reachability. Solutions like those in [\[ICC2016\]](#), for instance, use a binary representation of links in a network to realize a path-based forwarding action that purely acts

on node-local state, independent from the nature of the path or

the communications traversing over it. As discussed in [Section 7](#), the limitation of forwarding actions to only apply to defined (IP) packet header fields results, however, in issues that need special consideration when realizing such solutions in real-world deployments.

The next subsection further details which elements are needed when interacting with programmable forwarders in an SDN context.

[4.2](#). SDN for Semantic Routing: The Intended Behavior

SDN minimizes the required changes to legacy (interior) routing protocols. More concretely, SDN can be used to provide the intended Semantic Routing behavior, especially:

- * Identify the forwarding elements that can be safely involved in providing the intended Semantic Routing features.
- * Maintain abstract topologies that involve these elements and their capabilities.
- * Capture application-specific intents and derive the corresponding forwarding requirements and, then, forwarding policies.
- * Map these abstract topologies to (groups of) applications with specific Semantic Routing needs.
- * Program a subset of nodes (called boundary nodes) with the required classification and marking policies to bind flows with their intended Semantic Routing behavior.

In order to adequately process the application flows that require specific differentiated forwarding, SDN controllers maintain a table that allows to unambiguously identify such flows. The content of that table is used to derive the appropriate classification/match rules that are then communicated by an SDN controller to a set of forwarding elements.

When volatile data (e.g., dynamic IP addresses) is used to build such rules, it is the responsibility of the SDN controllers to update the rules whenever a new identifier is used. Failure to maintain "fresh" classification rules will lead to service failure/degradation.

-
- * Supply intermediate nodes (that is, nodes that are not boundary nodes) with the appropriate rules to locate and interpret the bits within the packet to proceed with forwarding actions that comprise Semantic Routing.
 - * Automatically adjust, if possible, the network MTU to accommodate the overhead that is required by any extra bits to signal semantic routing behavior.
 - * Instruct egress boundary nodes about the required actions such as stripping or setting any Semantic Routing bits.
 - * Interact with the underlying nodes to maintain, retrieve, and disseminate the appropriate data that is used for assuring that Semantic Routing policies are appropriately fulfilled.
 - * Configure OAM policies to measure the experience and adjust the forwarding behavior.
 - * Monitor the network and detect parts of the network where such policies are broken.
 - * Automate the overall procedure [[RFC8969](#)].

At least three approaches can be considered by an SDN controller to accomplish the above tasks:

- * Compute (centrally) the differentiated paths and install the required forwarding rules in involved nodes. Strict or loose paths may be installed. This approach has the merit of implementing new path selection algorithms without requiring them to be supported by every involved node.
- * Assign (centrally) differentiated link information and install the required forwarding rules in the involved nodes. End-to-end paths are constructed without involvement of the SDN controller, utilizing the link information to establish path identification information on which installed forwarding rules can act upon without additional path-specific knowledge being required. See [[ICC2016](#)] for an example of such approach.
- * Rely upon a distributed routing protocol to customize the route selection process ([[I-D.ietf-lsr-flex-algo](#)], for example). In such case, the SDN controller is responsible for communicating the parameters to be used for route selection process, select the nodes that will participate in a given topology, and configure any

tunnels to interconnect these nodes.

A hierarchical SDN design can also be considered, where specific controllers are enabled in each domain with dedicated interfaces to share data (e.g., radio bottleneck, expectations). These domains do not need to support the same technological implementation. The interaction between the SDN controllers eases the delivery of consistent Semantic Routing behaviors without requiring common domain configuration.

[5.](#) Policy-Based Semantic Routing

TBD

****SDN techniques as a whole are an instantiation of the policy-based management framework.****

[6.](#) Network-Wide Coordination

TBD

[7.](#) Applying Semantic Information to Packets

Given the focus of SDN is the use within IP networks, semantic information that can be used in SDN-based semantic routing is limited to those fields being defined in related SDN specifications; see [Section 2](#) for more information.

With this, SDN aligns with the concept of semantic routing [[I-D.farrel-irtf-introduction-to-semantic-routing](#)] in that it allows for range of packet header fields beyond mere IP addresses to be used in forwarding actions.

However, solutions have also been devised that "overwrite" existing protocol fields in order for them to be used in an SDN forwarding action outside their original semantics. [[POINT2015](#)][[POINT2016](#)] outline an example for such solution in which SDN is used for a path-based forwarding decision; while no "path" information is foreseen as an actionable packet header field in IPv6.

Here, the path is constructed by a path-computation element (PCE)

that matches a given service name against previously announced locations where said service name is located. The path is represented as a concatenation of individual link information, which in turn is used to SDN node locally forward the packet after arrival. Given the binary structure of the end-to-end path information, the SDN forwarding operation can be implemented in a standard-compliant manner with its realization described in [ICC2016] as a arbitrary wildcard matching operation.

However, the constraint of acting only on limited packet fields requires that the path information needs transfer in one of those standard-defined packet header fields; thereby overwriting any existing packet header field. As described in [POINT2016], the IPv6 address fields are used for this purpose, representing the longest continuous binary field in the IPv6 header (256 bit in total), thereby allowing for representing topologies with up to 256 links.

Given the approach chosen in [POINT2016], any IPv6 address information, if needed, is provided in the encapsulated payload, leading to repetitive encapsulation overhead by carrying two IP headers in a single packet, one used for path-based forwarding and one for the operations in arriving endpoint. Only newer forwarding plane architectures, such as P4, would allow for removing such overhead by placing the path information into another packet header field (or even the payload as an extended header of sort) to act upon.

8. Benefits and Concerns with the Use of SDN for Semantic Routing

The programmability of SDN provides a fertile ground for forwarding decision that go beyond the reachability information provided through IPv4/v6 addresses, e.g., by using other packet header fields. This not only allows for extending the simple reachability-driven forwarding decision with richer, e.g., policy-based, decisions (as discussed in [Section 5](#)), it may also enable new forwarding paradigms per se, such as those in [POINT2016], which in turn may realize forwarding behaviours like multicast at much lower cost points and higher efficiency (see [ICC2016]).

However, SDN specifications have limited capabilities when it comes to the additional packet header fields that may be used for

forwarding actions. As a consequence, "true" semantic routing on any semantic enhancement, which is included in the packet, is only possible in a manner limited to those fields.

Solutions such as those in [[POINT2016](#)], using methods outlined in [[ICC2016](#)], attempt to break this limitation albeit by overwriting standard-defined packet header fields, thereby changing the semantics of those fields within the realm of where the "re-defined" semantics are defined.

This limits any solution to a limited domain [[RFC8799](#)]. More importantly, the redefinition of packet fields poses the danger of exposing this (non standard compliant) semantic to elements outside the limited domain; semantic leakage may occur, requiring appropriate methods such as dedicated gateways for preventing such leakage. This can be seen in [[POINT2016](#)], where the boundaries to IP-compliant end

devices and other domains alike are delimited by dedicated gateway elements. Those gateways usually act at higher layers than the SDN forwarding layer, thereby incurring complexity and often delay.

See also [[I-D.king-irtf-challenges-in-routing](#)] for a discussion of issues and concerns that need to be examined when applying a new routing or forwarding paradigm to a self-contained network or Internet.

[9.](#) Security Considerations

SDN-related considerations are discussed in [Section 5 of \[RFC7149\]](#).

[10.](#) IANA Considerations

This document makes no requests for IANA action.

[11.](#) Acknowledgements

Thanks to George Polyzos for helpful comments on this work.

[12.](#) Contributors

George Xylomenos
Email: xgeorge@aueb.gr

13. Informative References

[I-D.farrel-irtf-introduction-to-semantic-routing]

Farrel, A. and D. King, "An Introduction to Semantic Routing", Work in Progress, Internet-Draft, [draft-farrel-irtf-introduction-to-semantic-routing-03](https://www.ietf.org/archive/id/draft-farrel-irtf-introduction-to-semantic-routing-03), 22 January 2022, <<https://www.ietf.org/archive/id/draft-farrel-irtf-introduction-to-semantic-routing-03.txt>>.

[I-D.ietf-idr-performance-routing]

Xu, X., Hegde, S., Talaulikar, K., Boucadair, M., and C. Jacquenet, "Performance-based BGP Routing Mechanism", Work in Progress, Internet-Draft, [draft-ietf-idr-performance-routing-03](https://www.ietf.org/archive/id/draft-ietf-idr-performance-routing-03), 22 December 2020, <<https://www.ietf.org/archive/id/draft-ietf-idr-performance-routing-03.txt>>.

[I-D.ietf-lsr-flex-algo]

Psenak, P., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", Work in Progress, Internet-Draft, [draft-ietf-lsr-flex-algo-18](https://www.ietf.org/archive/id/draft-ietf-lsr-flex-algo-18), 25 October 2021, <<https://www.ietf.org/archive/id/draft-ietf-lsr-flex-algo-18.txt>>.

[I-D.ietf-teas-ietf-network-slices]

Farrel, A., Gray, E., Drake, J., Rokui, R., Homma, S., Makhijani, K., Contreras, L. M., and J. Tantsura, "Framework for IETF Network Slices", Work in Progress, Internet-Draft, [draft-ietf-teas-ietf-network-slices-05](https://www.ietf.org/archive/id/draft-ietf-teas-ietf-network-slices-05), 25 October 2021, <<https://www.ietf.org/archive/id/draft-ietf-teas-ietf-network-slices-05.txt>>.

[I-D.king-irtf-challenges-in-routing]

King, D., Farrel, A., and C. Jacquenet, "Challenges for the Internet Routing Infrastructure Introduced by Semantic

Routing", Work in Progress, Internet-Draft, [draft-king-irtf-challenges-in-routing-06](https://www.ietf.org/archive/id/draft-king-irtf-challenges-in-routing-06), 22 January 2022, <<https://www.ietf.org/archive/id/draft-king-irtf-challenges-in-routing-06.txt>>.

[ICC2016] Reed, M., Al-Naday, M., Thomos, N., Trossen, D., Petropoulos, G., and S. Spirou, "Stateless multicast switching in software defined networks", Paper IEEE ICC 2016, 2016.

[POINT2015] Trossen, D., Reed, M., Riihijarvi, J., Georgiades, M., Xylomenos, G., and S. Fotiou, "IP Over ICN: The Better IP?", Paper EuCNC (European Conference on Networks and Communications), Paris, France, 2015.

[POINT2016] Kim, S.-Y., Robitzsch, S., Trossen, D., Reed, M., Al-Naday, M., and J. Riihijarvi, "Realizing IP-based Services over an Information-Centric Networking Transport Network", Paper Proceedings of the 3rd ACM Conference on Information-Centric Networking, Pages 215-216, 2016.

[RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", [RFC 7149](https://www.rfc-editor.org/info/rfc7149), DOI 10.17487/RFC7149, March 2014, <<https://www.rfc-editor.org/info/rfc7149>>.

[RFC7297] Boucadair, M., Jacquenet, C., and N. Wang, "IP Connectivity Provisioning Profile (CPP)", [RFC 7297](https://www.rfc-editor.org/info/rfc7297), DOI 10.17487/RFC7297, July 2014, <<https://www.rfc-editor.org/info/rfc7297>>.

[RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", [RFC 7426](https://www.rfc-editor.org/info/rfc7426), DOI 10.17487/RFC7426, January 2015, <<https://www.rfc-editor.org/info/rfc7426>>.

- [RFC7460] Chandramouli, M., Claise, B., Schoening, B., Quittek, J., and T. Dietz, "Monitoring and Control MIB for Power and Energy", [RFC 7460](#), DOI 10.17487/RFC7460, March 2015, <<https://www.rfc-editor.org/info/rfc7460>>.
- [RFC7597] Troan, O., Ed., Dec, W., Li, X., Bao, C., Matsushima, S., Murakami, T., and T. Taylor, Ed., "Mapping of Address and Port with Encapsulation (MAP-E)", [RFC 7597](#), DOI 10.17487/RFC7597, July 2015, <<https://www.rfc-editor.org/info/rfc7597>>.
- [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>>.
- [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>>.
- [RFC8041] Bonaventure, O., Paasch, C., and G. Detal, "Use Cases and Operational Experience with Multipath TCP", [RFC 8041](#), DOI 10.17487/RFC8041, January 2017, <<https://www.rfc-editor.org/info/rfc8041>>.
- [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>>.
- [RFC8309] Wu, Q., Liu, W., and A. Farrel, "Service Models Explained", [RFC 8309](#), DOI 10.17487/RFC8309, January 2018, <<https://www.rfc-editor.org/info/rfc8309>>.

- [RFC8517] Dolson, D., Ed., Snellman, J., Boucadair, M., Ed., and C. Jacquenet, "An Inventory of Transport-Centric Functions Provided by Middleboxes: An Operator Perspective", [RFC 8517](#), DOI 10.17487/RFC8517, February 2019,

<<https://www.rfc-editor.org/info/rfc8517>>.

- [RFC8597] Contreras, LM., Bernardos, CJ., Lopez, D., Boucadair, M., and P. Iovanna, "Cooperating Layered Architecture for Software-Defined Networking (CLAS)", [RFC 8597](#), DOI 10.17487/RFC8597, May 2019, <<https://www.rfc-editor.org/info/rfc8597>>.
- [RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", [RFC 8799](#), DOI 10.17487/RFC8799, July 2020, <<https://www.rfc-editor.org/info/rfc8799>>.
- [RFC8803] Bonaventure, O., Ed., Boucadair, M., Ed., Gundavelli, S., Seo, S., and B. Hesmans, "0-RTT TCP Convert Protocol", [RFC 8803](#), DOI 10.17487/RFC8803, July 2020, <<https://www.rfc-editor.org/info/rfc8803>>.
- [RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", [RFC 8926](#), DOI 10.17487/RFC8926, November 2020, <<https://www.rfc-editor.org/info/rfc8926>>.
- [RFC8969] Wu, Q., Ed., Boucadair, M., Ed., Lopez, D., Xie, C., and L. Geng, "A Framework for Automating Service and Network Management with YANG", [RFC 8969](#), DOI 10.17487/RFC8969, January 2021, <<https://www.rfc-editor.org/info/rfc8969>>.

Authors' Addresses

Mohamed Boucadair
Orange
Rennes
France

Email: mohamed.boucadair@orange.com

Dirk Trossen
Huawei
Munich
Germany

Email: dirk.trossen@huawei.com

Adrian Farrel
Old Dog Consulting
United Kingdom

Email: adrian@olddog.co.uk

