Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: May 14, 2022

K. Talaulikar, Ed.
P. Psenak
Cisco Systems
S. Zandi
G. Dawra
LinkedIn
November 10, 2021

Flexible Algorithm Definition Advertisement with BGP Link-State draft-ietf-idr-bgp-ls-flex-algo-08

Abstract

Flexible Algorithm is a solution that allows routing protocols (viz. OSPF and IS-IS) to compute paths over a network based on user-defined (and hence, flexible) constraints and metrics. The computation is performed by routers participating in the specific network in a distribute manner using a Flex Algorithm definition. This definition provisioned on one or more routers and propagated (viz. OSPF and IS-IS flooding) through the network.

BGP Link-State (BGP-LS) enables the collection of various topology information from the network. This draft defines extensions to BGP-LS address-family to advertise the Flexible Algorithm Definition as a part of the topology information from the network.

Status of This Memo

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

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

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

This Internet-Draft will expire on May 14, 2022.

Table of Contents

1. Introduction

IGP protocols (OSPF and IS-IS) traditionally compute best paths over the network based on the IGP metric assigned to the links. Many network deployments use RSVP-TE [RFC3209] based or Segment Routing (SR) Policy [RFC8402] based solutions to enforce traffic over a path that is computed using different metrics or constraints than the shortest IGP path. [I-D.ietf-lsr-flex-algo] defines the Flexible Algorithm solution that allows IGPs themselves to compute constraint based paths over the network.

Flexible Algorithm is called so as it allows a user the flexibility to define

- the type of calculation to be used (e.g. shortest path)
- the metric type to be used (e.g. IGP metric or TE metric)
- the set of constraints to be used (e.g. inclusion or exclusion of certain links using affinities)

The operations of the flexible algorithm solution are described in detail in [I-D.ietf-lsr-flex-algo] and a high-level summary of the same is described here for clarity. The network operator enables the participation of specific nodes in the network for a specific algorithm and then provisions the definition of that flexible algorithm on one or more of these nodes. The nodes where the flexible algorithm definition (FAD) is advertised then flood these definitions via respective IGP (IS-IS and OSPFv2/v3) mechanisms to all other nodes in the network. The nodes select the definition for each algorithm based on the flooded information in a deterministic manner and thus all nodes participating in a flexible algorithm computation arrive at a common understanding of the type of calculation that they need to use.

When using Segment Routing (SR) [RFC8402] MPLS forwarding plane [RFC8660], the result of a flex algorithm computation is the provisioning of the Prefix SIDs associated with that algorithm with paths based on the topology computed based on that algorithm's definition. When using SR over IPv6 (SRv6) [RFC8986], the result of a flex algorithm computation is the provisioning of the SRv6 Locators associated with that algorithm with paths based on the topology computed based on that algorithm. This flex algorithm computation is within an IGP area or level similar to the default shortest path tree (SPT) algorithm.

A flex algorithm specific metric MAY be advertised along with the prefix as described in [I-D.ietf-lsr-flex-algo] to enable end-to-end optimal path computation for prefixes across multiple areas/domains in the flex algorithm computation for the SR-MPLS forwarding plane.

The BGP-LS extensions for SR are defined in [RFC9085] and [I-D.ietf-idr-bgpls-srv6-ext]. They include the

- SR Algorithm TLV to indicate the participation of a node in a flex algorithm computation
- Prefix SID TLV to indicate the association of the Prefix-SIDs to a specific flex algorithm for SR-MPLS forwarding

- SRv6 Locator TLV to indicate the Locator for specific flex algorithm for SRv6 forwarding

Thus a controller or a Path Computation Engine (PCE) is aware of the IGP topology across multiple domains which includes the above information related to the flexible algorithm. This draft defines extensions to BGP-LS for carrying the FAD information so that it enables the controller/PCE to learn the mapping of the flex algorithm number to its definition in each area/domain of the underlying IGP. The controller/PCE also learns the type of computation used and the constraints for the same. This information can then be leveraged by it for setting up SR Policy paths end to end across domains by leveraging the appropriate Flex Algorithm specific SIDs in its Segment List [I-D.ietf-spring-segment-routing-policy]. e.g. picking the Flex Algorithm Prefix SID (in case of SR-MPLS) or End SID (in case of SRv6) of ABRs/ASBRs corresponding to a definition that optimizes on the delay metric enables the PCE/controller to build an end to end low latency path across IGP domains with minimal SIDs in the SID list.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. BGP-LS Extensions for Flex Algo

The BGP-LS [RFC7752] specifies the Node NLRI for the advertisement of nodes along with their attributes using the BGP-LS Attribute, the Link NLRI for the advertisement of links along with their attributes using the BGP-LS Attribute and the Prefix NLRI for the advertisement of prefixes along with their attributes using the BGP-LS Attribute.

The FAD advertised by a node is considered as its node level attributes and advertised as such.

Various link attributes like affinities and SRLGs used during the Flex-Algorithm path calculations in IS-IS and OSPF are advertised in those protocols using the Application Specific Link Attribute (ASLA) advertisements as described in [I-D.ietf-lsr-flex-algo]. The BGP-LS extensions for ASLA advertisements [I-D.ietf-idr-bgp-ls-app-specific-attr] MUST be used for the advertisement of these Flex-Algorithm application-specific link attributes from the underlying IGP protocols using the Flexible

Algorithm application specific bit defined in [I-D.ietf-lsr-flex-algo].

The Flexible Algorithm Prefix Metric (FAPM) are considered as prefix attributes and advertised as such.

3. Flexible Algorithm Definition

This document defines a new optional BGP-LS Attribute TLV associated with the Node NLRI called the Flexible Algorithm Definition (FAD) TLV and its format is as follows:

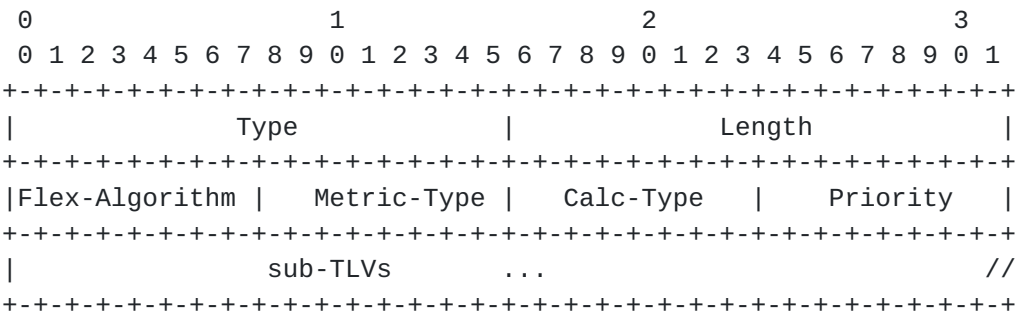


Figure 1: Flex Algorithm Definition TLV

where:

- Type: 1039
- Length: variable. Minimum of 4 octets.
- Flex-Algorithm : 1 octet value in the range between 128 and 255 inclusive which is the range defined for Flexible Algorithms in the IANA "IGP Parameters" registries under the "IGP Algorithm Types" registry [I-D.ietf-lsr-flex-algo].
- Metric-Type : 1 octet value indicating the type of metric used in the computation. Values allowed come from the IANA "IGP Parameters" registries under the "Flexible Algorithm Definition Metric-Type" registry [I-D.ietf-lsr-flex-algo].
- Calculation-Type : 1 octet value in the range between 0 and 127 inclusive which is the range defined for the standard algorithms in the IANA "IGP Parameters" registries under the "IGP Algorithm Types" registry [I-D.ietf-lsr-flex-algo].
- Priority : 1 octet value between 0 and 255 inclusive that specifies the priority of the FAD.

- sub-TLVs : zero or more sub-TLVs may be included as described further in this section.

The FAD TLV can only be added to the BGP-LS Attribute of the Node NLRI if the corresponding node originates the underlying IGP TLV/sub-TLV as described below. This information is derived from the protocol specific advertisements as below.

- IS-IS, as defined by the ISIS Flexible Algorithm Definition sub-TLV in [I-D.ietf-lsr-flex-algo].

- OSPFv2/OSPFv3, as defined by the OSPF Flexible Algorithm Definition TLV in [I-D.ietf-lsr-flex-algo].

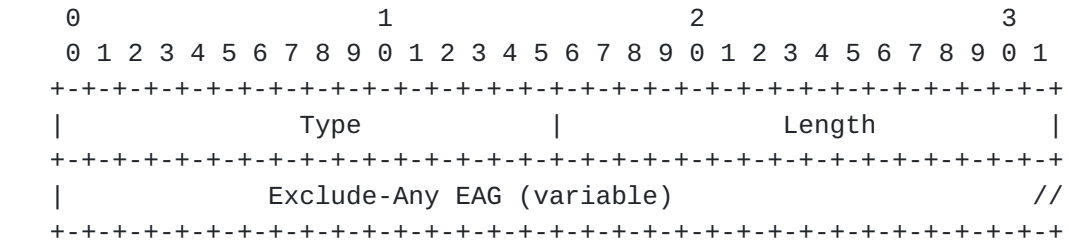
The BGP-LS Attribute associated with a Node NLRI MAY include one or more FAD TLVs corresponding to the FAD for each algorithm that the particular node is advertising.

The following sub-sections define the sub-TLVs for the FAD TLV.

3.1. Flex Algo Exclude Any Affinity

The Flex Algo Exclude Any Affinity sub-TLV is an optional sub-TLV that is used to carry the affinity constraints [RFC2702] associated with the FAD and enable the exclusion of links carrying any of the specified affinities from the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

- Type: 1040

- Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.

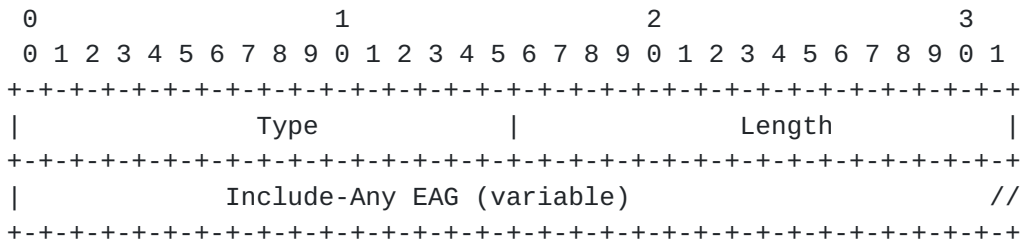
- Exclude-Any EAG : the bitmask used to represent the affinities to be excluded.

The information in the Flex Algo Exclude Any Affinity sub-TLV is derived from the IS-IS and OSPF protocol specific Flexible Algorithm Exclude Admin Group sub-TLV as defined in [I-D.ietf-lsr-flex-algo].

3.2. Flex Algo Include Any Affinity

The Flex Algo Include Any Affinity sub-TLV is an optional sub-TLV that is used to carry the affinity constraints [RFC2702] associated with the FAD and enable the inclusion of links carrying any of the specified affinities in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

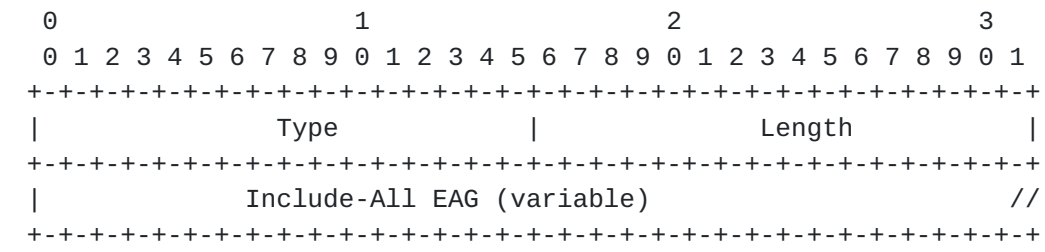
- Type: 1041
- Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.
- Include-Any EAG : the bitmask used to represent the affinities to be included.

The information in the Flex Algo Include Any Affinity sub-TLV is derived from the IS-IS and OSPF protocol specific Flexible Algorithm Include-Any Admin Group sub-TLV as defined in [I-D.ietf-lsr-flex-algo].

3.3. Flex Algo Include All Affinity

The Flex Algo Include All Affinity sub-TLV is an optional sub-TLV that is used to carry the affinity constraints [RFC2702] associated with the FAD and enable the inclusion of links carrying all of the specified affinities in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

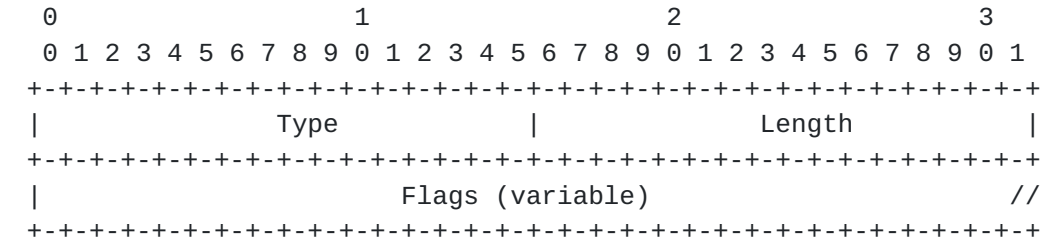
- Type: 1042
- Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.
- Include-All EAG : the bitmask used to represent the affinities to be included.

The information in the Flex Algo Include All Affinity sub-TLV is derived from the IS-IS and OSPF protocol specific Flexible Algorithm Include-All Admin Group sub-TLV as defined in [I-D.ietf-lsr-flex-algo].

3.4. Flex Algo Definition Flags

The Flex Algo Definition Flags sub-TLV is an optional sub-TLV that is used to carry the flags associated with the FAD that are used in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo].

The sub-TLV has the following format:



where:

- Type: 1043
- Length: variable. MUST be a multiple of 4 octets.
- Flags : the bitmask used to represent the flags for the FAD as introduced by [I-D.ietf-lsr-flex-algo] and listed in the "Flex-

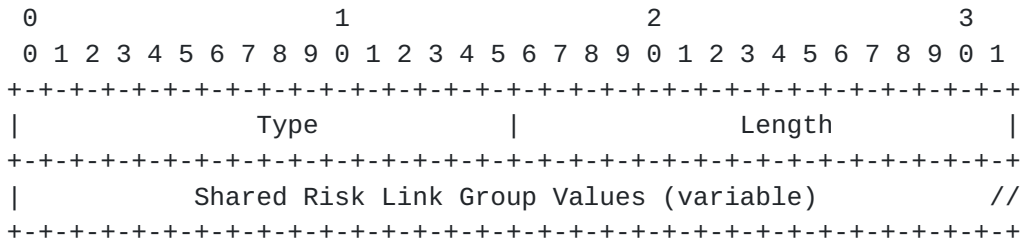
Algorithm Definition Flags" registry under the "Interior Gateway Protocol (IGP) Parameters" IANA registry.

The information in the Flex Algo Definition Flags sub-TLV is derived from the IS-IS and OSPF protocol specific Flexible Algorithm Definition Flags sub-TLV as defined in [I-D.ietf-lsr-flex-algo].

3.5. Flex Algo Exclude SRLG

The Flex Algo Exclude SRLG sub-TLV is an optional sub-TLV that is used to carry the shared risk link group (SRLG) [RFC4202] information associated with the FAD and enable the exclusion of links that are associated with any of the specified SRLG in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo]. The SRLGs associated with a link are carried in the BGP-LS Shared Link Risk Group (TLV 1096) [RFC7752].

The sub-TLV has the following format:



where:

- Type: 1045
- Length: variable, dependent on the number of SRLG values. MUST be a multiple of 4 octets.
- SRLG Values : One or more SRLG values, each of 4 octet size, as defined in [RFC4202].

The information in the Flex Algo SRLG Exclude sub-TLV is derived from the IS-IS and OSPF protocol specific Flexible Algorithm Exclude SRLG sub-TLV as defined in [I-D.ietf-lsr-flex-algo].

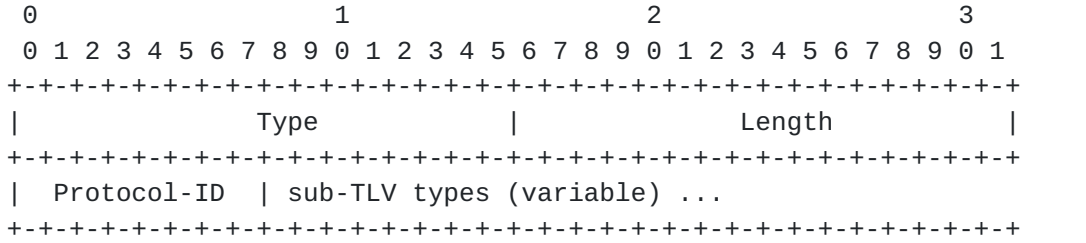
3.6. Flex Algo Unknown

The OSPF and ISIS signaling for FAD allows for extensions via new sub-TLVs under the respective IGP's Flex Algorithm Definition TLV. As specified in section 5.3 of [I-D.ietf-lsr-flex-algo], it is important that the entire FAD be understood by anyone using it for computation purposes. Therefore the FAD is different from most other

protocol extensions where the skipping or ignoring of unknown or unsupported sub-TLV information does not affect the base behavior.

The Flex Algo Unknown sub-TLV is an optional sub-TLV that is used to indicate the presence of unknown or unsupported FAD sub-TLVs. The need for this sub-TLV arises when the BGP-LS implementation on the advertising node does not support one or more of the FAD sub-TLVs present in the IGP advertisement.

The sub-TLV has the following format:



where:

- Type: TBD
- Length: variable
- Protocol-ID: Indicates the BGP-LS Protocol-ID of the protocol from which the FAD is being advertised via BGP-LS. The values are from the "BGP-LS Protocol-IDs" registry under the IANA BGP-LS Parameters registry.
- Sub-TLV Types : Zero or more sub-TLV types that are unknown or unsupported by the node originating the BGP-LS advertisement. The size of each sub-TLV type depends on the protocol indicated by the Protocol-ID field e.g., for ISIS each sub-TLV type would be of size 1 byte while for OSPF each sub-TLV type would be of size 2 bytes.

The discussion on the use of the FAD information by the consumers of the BGP-LS information is beyond the scope of this document. However, it is RECOMMENDED that the choice of the node used for originating the IGP topology information into BGP-LS be made such that the advertising node supports all the FAD extensions in use in its part of the network. This avoids the scenario where an incomplete FAD gets advertised via BGP-LS.

The node originating the advertisement SHOULD include the Flex Algo Unknown sub-TLV when it comes across an unsupported or unknown sub-TLV in the corresponding FAD in the IS-IS and OSPF advertisement.

This serves as an indication that the FAD information in BGP-LS is incomplete and is not usable for computation purposes. When advertising the Flex Algo Unknown sub-TLV, the protocol specific sub-TLV types that are unsupported or unknown SHOULD be included. This information serves as a diagnostic aid.

4. Flex Algorithm Prefix Metric

This document defines a new optional BGP-LS Attribute TLV associated with the Prefix NLRI called the Flexible Algorithm Prefix Metric (FAPM) TLV and its format is as follows:

where:

- Type: 1044
- Length: 8 octets.
- Flex-Algorithm : 1 octet value in the range between 128 and 255 inclusive which is the range defined for Flexible Algorithms in the IANA "IGP Parameters" registries under the "IGP Algorithm Types" registry [I-D.ietf-lsr-flex-algo].
- Flags: single octet value and only applicable for OSPF as defined in [I-D.ietf-lsr-flex-algo]. The value MUST be set to 0 for ISIS and ignored by the receiver.
- Reserved : 2 octet value that SHOULD be set to 0 by the originator and MUST be ignored by the receiver.
- Metric : 4 octets field to carry the metric information.

The FAPM TLV can be added to the BGP-LS Attribute of the Prefix NLRI originated by a node, only if the corresponding node originates the Prefix in along with the underlying IGP TLV/sub-TLV as described below. This information is derived from the protocol specific advertisements as below.

- IS-IS, as defined by the ISIS Flexible Algorithm Prefix Metric sub-TLV in [I-D.ietf-lsr-flex-algo].

- OSPFv2/OSPFv3, as defined by the OSPF Flexible Algorithm Prefix Metric sub-TLV in [I-D.ietf-lsr-flex-algo].

The BGP-LS Attribute associated with a Prefix NLRI MAY include one or more FAPM TLVs corresponding to the Flexible Algorithm Prefix Metric for each algorithm associated with that particular prefix.

5. IANA Considerations

This document requests assigning code-points from the registry "BGP-LS Node Descriptor, Link Descriptor, Prefix Descriptor, and Attribute TLVs" <https://www.iana.org/assignments/bgp-ls-parameters/bgp-ls-parameters.xhtml#node-descriptor-link-descriptor-prefix-descriptor-attribute-tlv> based on the table below which reflects the values assigned via the early allocation process. The column "IS-IS TLV/ Sub-TLV" defined in the registry does not require any value and should be left empty.

Code Point	Description	Length
1039	Flex Algorithm Definition TLV	variable
1040	Flex Algo Exclude Any Affinity sub-TLV	variable
1041	Flex Algo Include Any Affinity sub-TLV	variable
1042	Flex Algo Include All Affinity sub-TLV	variable
1043	Flex Algo Definition Flags sub-TLV	variable
1044	Flex Algorithm Prefix Metric TLV	variable
1045	Flex Algorithm Exclude SRLG sub-TLV	variable
TBD	Flex Algorithm Unknown sub-TLV	variable

6. Manageability Considerations

The new protocol extensions introduced in this document augment the existing IGP topology information that was distributed via [RFC7752]. Procedures and protocol extensions defined in this document do not affect the BGP protocol operations and management other than as discussed in the Manageability Considerations section of [RFC7752]. Specifically, the malformed NLRIs attribute tests in the Fault Management section of [RFC7752] now encompass the new TLVs for the BGP-LS NLRI in this document.

The extensions specified in this document do not specify any new configuration or monitoring aspects in BGP or BGP-LS. The

specification of BGP models is an ongoing work based on [I-D.ietf-idr-bgp-model].

7. Security Considerations

The procedures and protocol extensions defined in this document do not affect the BGP security model. See the "Security Considerations" section of [RFC4271] for a discussion of BGP security. Also, refer to [RFC4272] and [RFC6952] for analyses of security issues for BGP. Security considerations for acquiring and distributing BGP-LS information are discussed in [RFC7752]. The TLVs introduced in this document are used to propagate the IGP Flexible Algorithm extensions defined in [I-D.ietf-lsr-flex-algo]. It is assumed that the IGP instances originating these TLVs will support all the required security (as described in [I-D.ietf-lsr-flex-algo]) in order to prevent any security issues when propagating the TLVs into BGP-LS. The advertisement of the node and prefix attribute information defined in this document presents no significant additional risk beyond that associated with the existing node and prefix attribute information already supported in [RFC7752].

8. Acknowledgements

The authors would like to thank Les Ginsberg, Amalesh Maity and Y F Siu for their reviews and contributions to this work. The authors would also like to thanks Jie Dong for his shepherd review.

9. References

9.1. Normative References

[I-D.ietf-idr-bgp-ls-app-specific-attr]
Talaulikar, K., Psenak, P., and J. Tantsura, "Application-Specific Attributes Advertisement with BGP Link-State", draft-ietf-idr-bgp-ls-app-specific-attr-08 (work in progress), November 2021.

[I-D.ietf-lsr-flex-algo]
Psenak, P., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", draft-ietf-lsr-flex-algo-18 (work in progress), October 2021.

[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>.

[RFC4202] Kompella, K., Ed. and Y. Rekhter, Ed., "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4202, DOI 10.17487/RFC4202, October 2005, <https://www.rfc-editor.org/info/rfc4202>.

[RFC7308] Osborne, E., "Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)", RFC 7308, DOI 10.17487/RFC7308, July 2014, <https://www.rfc-editor.org/info/rfc7308>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[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>.

9.2. Informative References

[I-D.ietf-idr-bgp-model]
Jethanandani, M., Patel, K., Hares, S., and J. Haas, "BGP YANG Model for Service Provider Networks", draft-ietf-idr-bgp-model-12 (work in progress), October 2021.

[I-D.ietf-idr-bgpls-srv6-ext]
Dawra, G., Filsfils, C., Talaulikar, K., Chen, M., Bernier, D., and B. Decraene, "BGP Link State Extensions for SRv6", draft-ietf-idr-bgpls-srv6-ext-09 (work in progress), November 2021.

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-14 (work in progress), October 2021.

[RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, DOI 10.17487/RFC2702, September 1999, <https://www.rfc-editor.org/info/rfc2702>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, DOI 10.17487/RFC4272, January 2006, <https://www.rfc-editor.org/info/rfc4272>.

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[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>.

[RFC8660] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", RFC 8660, DOI 10.17487/RFC8660, December 2019, <https://www.rfc-editor.org/info/rfc8660>.

[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>.

[RFC9085] Previdi, S., Talaulikar, K., Ed., Filsfils, C., Gredler, H., and M. Chen, "Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing", RFC 9085, DOI 10.17487/RFC9085, August 2021, <https://www.rfc-editor.org/info/rfc9085>.

Authors' Addresses

Ketan Talaulikar (editor)
Cisco Systems
India

Email: ketant.ietf@gmail.com

Peter Psenak
Cisco Systems
Slovakia

Email: ppsenak@cisco.com

Shawn Zandi
LinkedIn
USA

Email: szandi@linkedin.com

Gaurav Dawra
LinkedIn
USA

Email: gdawra.ietf@gmail.com