Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: 10 September 2023

K. Talaulikar
A. MahendraBabu
Cisco Systems
9 March 2023

Advertising Flexible Algorithm Extensions in BGP Link-State draft-aravindbabu-idr-bgp-ls-flex-algo-ext-00

Abstract

Flexible Algorithm is a solution that allows some routing protocols (e.g., 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 distributed manner using a Flexible Algorithm Definition. This Definition is provisioned on one or more routers and propagated through the network by OSPF and IS-IS flooding.

BGP Link-State (BGP-LS) enables the collection of various topology information from the network. RFC9351 introduced BGP-LS support for the advertisement of Flexible Algorithm Definition as a part of the topology information from the network. This document specifies the advertisement of further Flexible Algorithm related extensions in BGP-LS.

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 10 September 2023.

Copyright Notice

Table of Contents

1. Introduction 2
1.1. Requirements Language 3
2. Advertising IP Algorithm Participation 3
3. Advertising IP Algorithm Reachability 4
4. Advertising Generic Metric for Links 6
5. Advertising Flexible Algorithm Definition Extensions 7
5.1. FAD Exclude Minimum Bandwidth Sub-TLV 7
5.2. FAD Exclude Maximum Link Delay Sub-TLV 8
5.3. FAD Reference Bandwidth Sub-TLV 8
5.4. FAD Bandwidth Thresholds Sub-TLV 9
5.5. Flexible Algorithm Exclude-Any Reverse Affinity Sub-TLV 11
5.6. Flexible Algorithm Include-Any Reverse Affinity Sub-TLV 12
5.7. Flexible Algorithm Include-All Reverse Affinity Sub-TLV 12
6. IANA Considerations 13
7. Manageability Considerations 14
8. Security Considerations 14
9. Acknowledgements 14
10. References 14
10.1. Normative References 14
10.2. Informative References 15
Authors' Addresses 16

1. Introduction

Flexible Algorithm is a solution that allows some routing protocols (e.g., 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 distributed manner using a Flexible Algorithm Definition. This Definition is provisioned on one or more routers and propagated through the network by OSPF and IS-IS flooding. [RFC9350] defines the base Flexible Algorithm solution that allows IGPs themselves to compute constraint-based paths over the network.

The extensions to BGP-LS [RFC7752] for the advertisement of the Flexible Algorithm Definition (FAD) information to enable learning of the mapping of the flexible algorithm number to its Definition in each area/domain of the underlying IGPs are defined in [RFC9351].

This document defines further extensions to BGP-LS for Flexible Algorithm as below:

* The extensions to the Flexible Algorithm so that it can be used with the regular IPv4 and IPv6 forwarding as defined for IGPs in [I-D.ietf-lsr-ip-flexalgo].

* The Flexible Algorithm Definition that is used to exclude based on bandwidth and metric constraints and to automatically calculate metrics for use in SPF calculation as defined for IGPs in [I-D.ietf-lsr-flex-algo-bw-con].

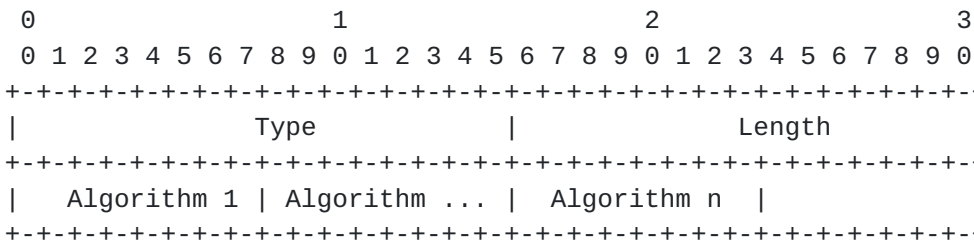
* The Flexible Algorithm Definition that is used to include/exclude links in the reverse direction of the traffic flow for SPF calculation as defined for IGPs in [I-D.ppsenak-lsr-igp-flex-algo-reverse-affinity].

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. Advertising IP Algorithm Participation

The IP Algorithm TLV is a BGP-LS Attribute TLV associated with the Node NLRI that is used for the algorithms associated with a given node. The format of this TLV is as follows:



where:

Figure 1: IP Algorithm TLV

* Type: TBA

* Length: Variable

* Algorithm: Single octet algorithm value between 128 and 255 inclusive.

The IP Algorithm TLV is derived from the following IGP protocol-specific advertisements:

* In the case of IS-IS, from the IS-IS IP Algorithm sub-TLV defined in [I-D.ietf-lsr-ip-flexalgo].

* In the case of OSPFv2/OSPFv3, from the OSPF IP Algorithm sub-TLV defined in [I-D.ietf-lsr-ip-flexalgo].

The IP Algorithm TLV is optional and it MUST NOT be advertised more than once in the BGP-LS Attribute. If multiple instances are present, then the first one MUST be considered valid, and the rest MUST be ignored.

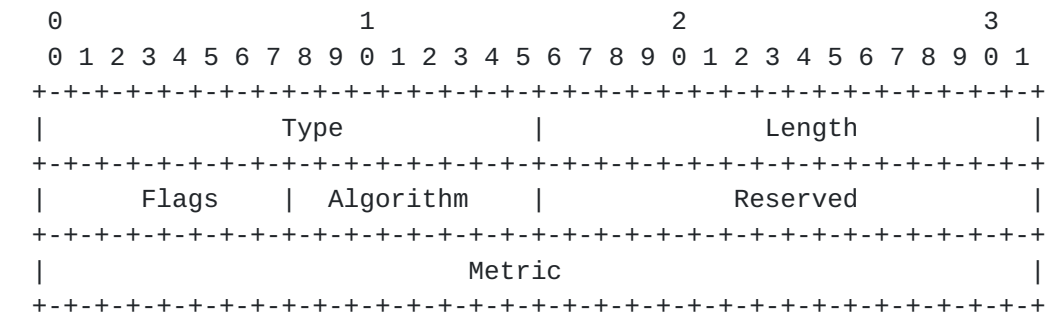
3. Advertising IP Algorithm Reachability

The normal or base (i.e., algorithm 0) prefix reachabilities are done using the BGP-LS IPv4/IPv6 Topology Prefix NLRIs defined in [RFC7752] along with its associated IGP metric carried within the IGP Metric TLV (TLV 1095) in the BGP-LS Attributed associated with the NLRI. The presence of IGP Metric TLV is what identifies the base/normal prefix reachability.

The IP algorithm-specific reachability advertisements are also done using the BGP-LS IPv4/IPv6 Topology Prefix NLRIs. However, these algorithm-specific advertisements MUST NOT carry an IGP Metric TLV along with them. Instead, the metric associated with the IP algorithm-specific prefix reachability is carried within the TLVs introduced in the following subsections.

A BGP-LS Consumer receiving an IPv4/IPv6 Topology Prefix NLRI advertisement that carries both an IGP Metric TLV along with any of the TLVs introduced in the following subsections, MUST consider it as a normal (i.e., algorithm 0) prefix reachability advertisement and MUST ignore all TLVs corresponding to algorithm-specific prefix reachability advertisements.

The IP Algorithm Prefix Reachability TLV is a BGP-LS Attribute TLV associated with the IPv4/IPv6 Topology Prefix NLRI that is used for the advertisement of the algorithm-specific prefix reachability from a given node. The format of this TLV is as follows:



where:

Figure 2: IP Algorithm Prefix Reachability TLV

* Type: TBA

* Length: 8

* Flags: 1 octet of flags. The flags are copied from the IS-IS IPv4/IPv6 Algorithm Prefix Reachability TLV [I-D.ietf-lsr-ip-flexalgo] or the OSPFv2/OSPFv3 IP Algorithm Prefix Reachability sub-TLV [I-D.ietf-lsr-ip-flexalgo] in the case of IS-IS or OSPFv2/OSPFv3 respectively.

* Algorithm: 1 octet value providing the Associated Algorithm from 128 to 255.

* Reserved: 2 octet value that MUST be set to 0 by the originator and ignored by the receiver.

* Metric: The algorithm-specific metric value.

The IP Algorithm Prefix Reachability TLV is derived from the following IGP protocol-specific advertisements:

* In the case of IS-IS, from the IS-IS IPv4/IPv6 Algorithm Prefix Reachability TLV defined in [I-D.ietf-lsr-ip-flexalgo]. The sub-TLVs are encoded using the BGP-LS Attribute TLVs defined for the IPv4/IPv6 Topology Prefix NLRI.

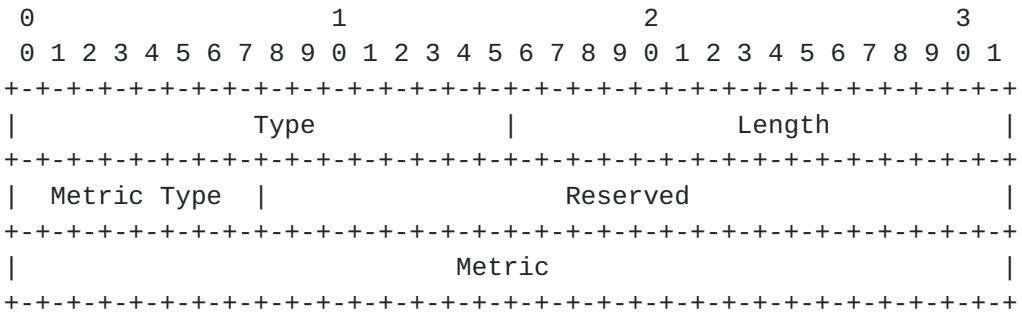
* In the case of OSPF, from the OSPFv2/OSPFv3 IP Algorithm Prefix Reachability sub-TLV defined in [I-D.ietf-lsr-ip-flexalgo].

The Multi-topology ID (MTID) associated with the underlying IGP advertisements is encoded using the Multi-Topology Identifier TLV (TLV 263) [RFC7752] as a Prefix Descriptor TLV when the advertisement is associated with a non-default topology. The IP Prefix value itself is encoded using the IP Reachability Information TLV (TLV 265) [RFC7752] as a Prefix Descriptor TLV.

The IP Algorithm Prefix Reachability TLV MUST NOT be advertised more than once in the BGP-LS Attribute. If multiple instances are present, then the first one MUST be considered valid, and the rest MUST be ignored.

4. Advertising Generic Metric for Links

The Generic Metric TLV is a BGP-LS Attribute TLV associated with the Link NLRI that is used for the advertisement of the generic metric(s) associated with a link. The format of this TLV is as follows:



where:

Figure 3: Generic Metric TLV

* Type: TBA

* Length: 8.

* Metric Type: 1 octet that carries a metric type from the IGP Metric Type registry

* Reserved: 3 octet value that MUST be set to 0 by the originator and ignored by the receiver.

* Metric: 4-octet field carrying the metric value. In the case of IS-IS, the value MUST be in the range of 0 - 16,777,215.

The Generic Metric TLV is derived from the following IGP protocol-specific advertisements:

* In the case of IS-IS, from the IS-IS Generic Metric sub-TLV defined in [I-D.ietf-lsr-flex-algo-bw-con].

* In the case of OSPF, from the OSPF Generic Metric sub-TLV defined in [I-D.ietf-lsr-flex-algo-bw-con].

The Generic Metric TLV MAY be advertised more than once in the BGP-LS Attribute, one for each metric type. If multiple instances are present for the same metric type, then the first one MUST be considered valid, and the rest MUST be ignored.

5. Advertising Flexible Algorithm Definition Extensions

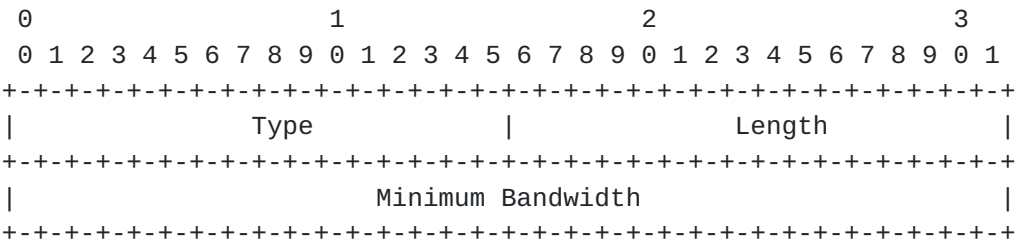
[RFC9351] introduced the Flexible Algorithm Definition (FAD) TLV that is advertised in the BGP-LS Attribute along with the Node NLRI for the advertisement of the Flexible Algorithm definition advertised by a given node in IGPs.

The following subsections define new sub-TLVs of the FAD TLV to cover further extensions of the IGP Flexible Algorithm solution.

5.1. FAD Exclude Minimum Bandwidth Sub-TLV

The FAD Exclude Minimum Bandwidth sub-TLV is an optional sub-TLV that is used to carry the minimum bandwidth associated with the FAD that are used in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo-bw-con].

The sub-TLV has the following format:



where:

Figure 4: FAD Exclude Minimum Bandwidth sub-TLV

* Type: TBA

* Length: 4 octets.

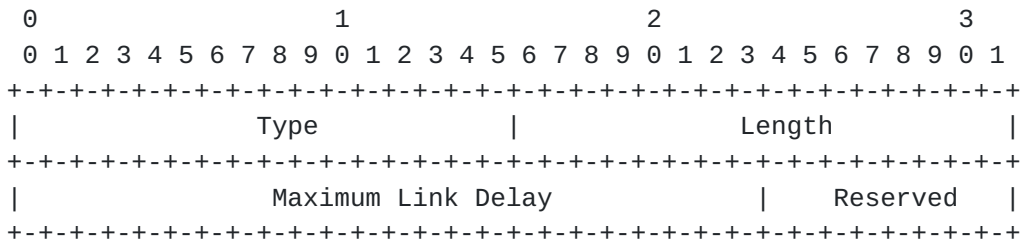
* Min Bandwidth: The minimum link bandwidth is encoded in 32 bits in IEEE floating point format. The units are bytes per second.

The information in the FAD Exclude Minimum Bandwidth sub-TLV is derived from the IS-IS and OSPF protocol-specific FAD Exclude Minimum Bandwidth sub-TLVs as defined in [I-D.ietf-lsr-flex-algo-bw-con].

5.2. FAD Exclude Maximum Link Delay Sub-TLV

The FAD Exclude Maximum Link Delay sub-TLV is an optional sub-TLV that is used to carry the maximum link delay information associated with the FAD that is used in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo-bw-con].

The sub-TLV has the following format:



where:

Figure 5: FAD Exclude Maximum Link Delay sub-TLV

* Type: TBA

* Length: 4 octets.

* Maximum Link Delay: The maximum link delay is encoded in microseconds.

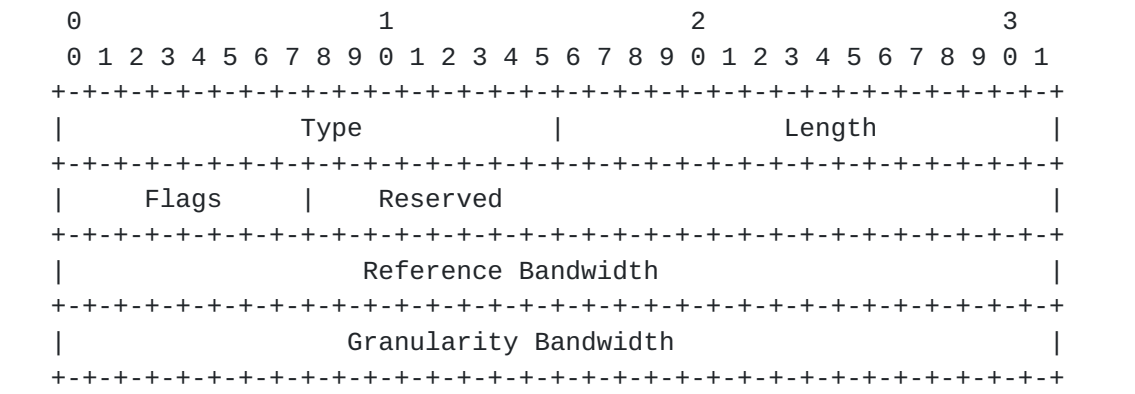
* Reserved: 1 octet field that MUST be set to 0 by the originator and ignored by the receiver.

The information in the FAD Exclude Maximum Link Delay sub-TLV is derived from the IS-IS and OSPF protocol-specific FAD Exclude Maximum Link Delay sub-TLVs as defined in [I-D.ietf-lsr-flex-algo-bw-con].

5.3. FAD Reference Bandwidth Sub-TLV

The FAD Reference Bandwidth sub-TLV is an optional sub-TLV that is used to carry the information needed for the reference bandwidth method of metric calculation associated with the FAD that is used in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo-bw-con].

The sub-TLV has the following format:



where:

Figure 6: FAD Reference Bandwidth sub-TLV

* Type: TBA

* Length: 12 octets.

* Flags: 1 octet of flags. The flags are copied from the IS-IS FAD Reference Bandwidth sub-TLV [I-D.ietf-lsr-flex-algo-bw-con] or the OSPF FAD Reference Bandwidth sub-TLV [I-D.ietf-lsr-flex-algo-bw-con] in the case of IS-IS or OSPF respectively.

* Reserved: 3 octet field that MUST be set to 0 by the originator and ignored by the receiver.

* Reference Bandwidth: The reference bandwidth is encoded in 32 bits in IEEE floating point format. The units are bytes per second.

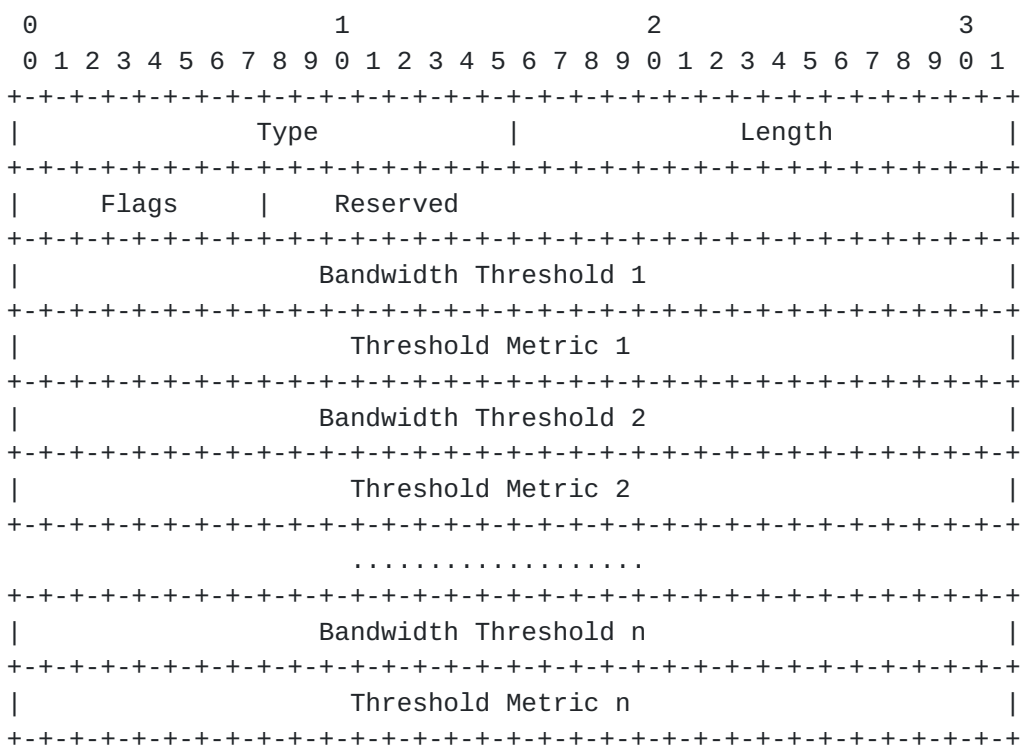
* Granularity Bandwidth: The granularity bandwidth is encoded in 32 bits in IEEE floating point format. The units are bytes per second.

The information in the FAD Reference Bandwidth sub-TLV is derived from the IS-IS and OSPF protocol-specific FAD Reference Bandwidth sub-TLV as defined in [I-D.ietf-lsr-flex-algo-bw-con].

5.4. FAD Bandwidth Thresholds Sub-TLV

The FAD Bandwidth Thresholds sub-TLV is an optional sub-TLV that is used to carry the information needed for bandwidth threshold method of metric calculation associated with the FAD that are used in the computation of the specific algorithm as described in [I-D.ietf-lsr-flex-algo-bw-con].

The sub-TLV has the following format:



where:

Figure 7: FAD Bandwidth Thresholds sub-TLV

* Type: TBA

* Length: 4 + (n * 8) octets. Here n is equal to the number of Threshold Metrics specified. n MUST be greater than or equal to 1.

* Flags: 1 octet of flags. The flags are copied from the IS-IS FAD Bandwidth Thresholds sub-TLV [I-D.ietf-lsr-flex-algo-bw-con] or the OSPF FAD Bandwidth Thresholds sub-TLV [I-D.ietf-lsr-flex-algo-bw-con] in the case of IS-IS or OSPF respectively.

* Reserved: 3 octet field that MUST be set to 0 by the originator and ignored by the receiver.

* Bandwidth Threshold (1 ... n): The bandwidth threshold is encoded in 32 bits in IEEE floating point format. The units are bytes per second.

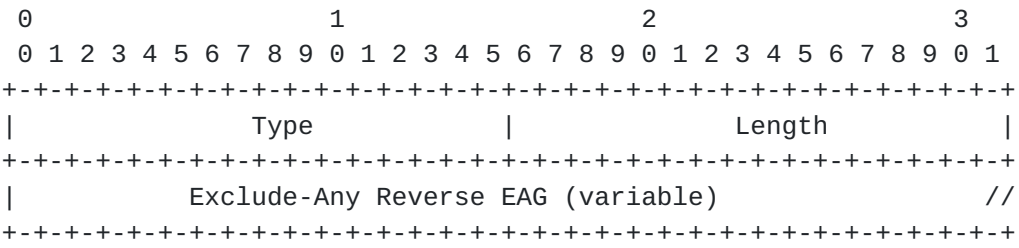
* Threshold Metric (1 ... n): 4 octet field carrying the threshold metric value. In the case of IS-IS, the value MUST be in the range of 0 - 16,777,215.

The information in the FAD Bandwidth Thresholds sub-TLV is derived from the IS-IS and OSPF protocol-specific FAD Bandwidth Thresholds sub-TLV as defined in [I-D.ietf-lsr-flex-algo-bw-con].

5.5. Flexible Algorithm Exclude-Any Reverse Affinity Sub-TLV

The Flexible Algorithm Exclude-Any Reverse Affinity sub-TLV is an optional sub-TLV that is used to carry the reverse affinity constraints 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.ppsenak-lsr-igp-flex-algo-reverse-affinity]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

Figure 8: Flexible Algorithm Exclude-Any Reverse Affinity sub-TLV

* Type: TBA

* Length: The total length of the value field in octets dependent on the size of the EAG. It MUST be a non-zero value and a multiple of 4.

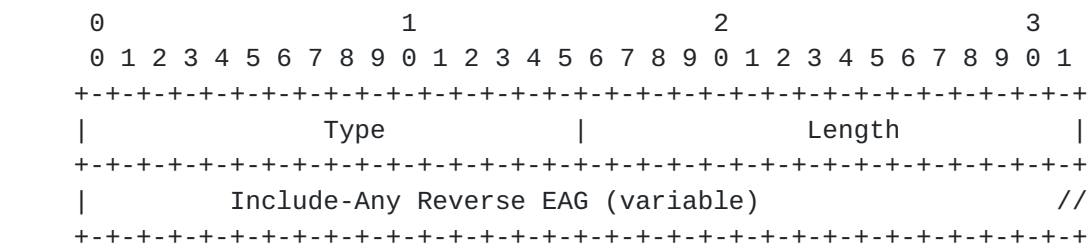
* Exclude-Any Reverse EAG: the EAG value.

The information in the Flexible Algorithm Exclude Any Reverse 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.ppsenak-lsr-igp-flex-algo-reverse-affinity].

5.6. Flexible Algorithm Include-Any Reverse Affinity Sub-TLV

The Flexible Algorithm Include-Any Reverse Affinity sub-TLV is an optional sub-TLV that is used to carry the affinity constraints 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.ppsenak-lsr-igp-flex-algo-reverse-affinity]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

Figure 9: Flexible Algorithm Include-Any Reverse Affinity sub-TLV

* Type: TBA

* Length: The total length of the value field in octets dependent on the size of the EAG. It MUST be a non-zero value and a multiple of 4.

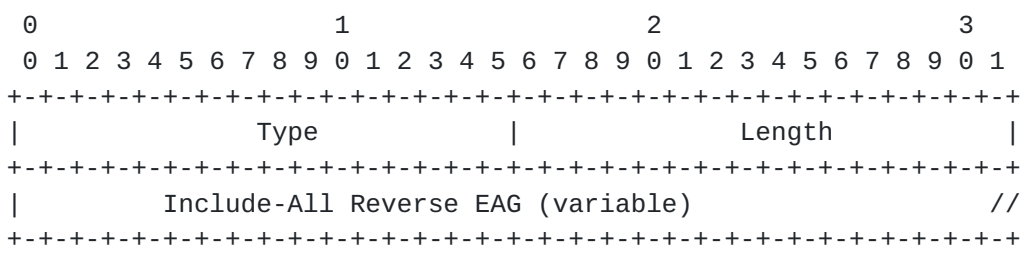
* Include-Any EAG: the EAG value.

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

5.7. Flexible Algorithm Include-All Reverse Affinity Sub-TLV

The Flexible Algorithm Include-All Reverse Affinity sub-TLV is an optional sub-TLV that is used to carry the affinity constraints 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.ppsenak-lsr-igp-flex-algo-reverse-affinity]. The affinity is expressed in terms of Extended Admin Group (EAG) as defined in [RFC7308].

The sub-TLV has the following format:



where:

Figure 10: Flexible Algorithm Include-All Reverse Affinity sub-TLV

* Type: TBA

* Length: The total length of the value field in octets dependent on the size of the EAG. It MUST be a non-zero value and a multiple of 4.

* Include-All EAG: the EAG value.

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

6. IANA Considerations

This document requests IANA to allocate code points from the "BGP-LS NLRI and Attribute TLVs" sub-registry of the "Border Gateway Protocol - Link-State (BGP-LS) Parameters" registry group.

Code Point	Description	Reference
TBA	IP Algorithm	this document
TBA	IP Algorithm Prefix Reachability	this document
TBA	Generic Metric	this document
TBA	Flexible Algorithm Exclude Minimum Bandwidth	this document
TBA	Flexible Algorithm Exclude Maximum Link Delay	this document
TBA	Flexible Algorithm Reference Bandwidth	this document
TBA	Flexible Algorithm Bandwidth Thresholds	this document
TBA	Flexible Algorithm Exclude Any Reverse Affinity	this document
TBA	Flexible Algorithm Include Any Reverse Affinity	this document
TBA	Flexible Algorithm Include All Reverse Affinity	this document

Figure 11: BGP-LS Flexible Algorithm Extensions Code Points

7. Manageability Considerations

This document does not introduce any new manageability considerations beyond those covered by [RFC9351].

8. Security Considerations

This document does not introduce any new security considerations beyond those covered by [RFC9351].

9. Acknowledgements

10. References

10.1. Normative References

[I-D.ietf-lsr-flex-algo-bw-con]
Hegde, S., Britto, W., Shetty, R., Decraene, B., Psenak, P., and T. Li, "Flexible Algorithms: Bandwidth, Delay, Metrics and Constraints", Work in Progress, Internet-Draft, draft-ietf-lsr-flex-algo-bw-con-05, 16 January 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-lsr-flex-algo-bw-con-05>.

[I-D.ietf-lsr-ip-flexalgo]
Britto, W., Hegde, S., Kaneriya, P., Shetty, R., Bonica, R., and P. Psenak, "IGP Flexible Algorithms (Flex-Algorithm) In IP Networks", Work in Progress, Internet-Draft, draft-ietf-lsr-ip-flexalgo-08, 19 December 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-lsr-ip-flexalgo-08>.

[I-D.ppsenak-lsr-igp-flex-algo-reverse-affinity]
Psenak, P., Horn, J., and A. Dhamija, "IGP Flexible Algorithms Reverse Affinity Constraint", Work in Progress, Internet-Draft, draft-ppsenak-lsr-igp-flex-algo-reverse-affinity-01, 4 January 2023, <https://datatracker.ietf.org/doc/html/draft-ppsenak-lsr-igp-flex-algo-reverse-affinity-01>.

[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>.

[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>.

[RFC9350] Psenak, P., Ed., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", RFC 9350, DOI 10.17487/RFC9350, February 2023, <https://www.rfc-editor.org/info/rfc9350>.

[RFC9351] Talaulikar, K., Ed., Psenak, P., Zandi, S., and G. Dawra, "Border Gateway Protocol - Link State (BGP-LS) Extensions for Flexible Algorithm Advertisement", RFC 9351, DOI 10.17487/RFC9351, February 2023, <https://www.rfc-editor.org/info/rfc9351>.

10.2. Informative References

[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>.

Authors' Addresses

Ketan Talaulikar
Cisco Systems
India
Email: ketant.ietf@gmail.com

Aravind Babu MahendraBabu
Cisco Systems
India
Email: aravindbabu.mahendrababu@gmail.com