

Inter-Domain Routing
Internet-Draft
Intended status: Standards Track
Expires: August 26, 2019

K. Talaulikar
P. Psenak
Cisco Systems
S. Zandi
G. Dawra
LinkedIn
February 22, 2019

Flexible Algorithm Definition Advertisement with BGP Link-State
draft-ketant-idr-bgp-ls-flex-algo-01

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.

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.

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 August 26, 2019.

Copyright Notice

Copyright (c) 2019 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. BGP-LS Extensions for Flex Algo Definition	4
2.1. Flex Algo Exclude Any Affinity	5
2.2. Flex Algo Include Any Affinity	6
2.3. Flex Algo Include All Affinity	7
3. IANA Considerations	7
4. Manageability Considerations	8
4.1. Operational Considerations	8
4.2. Management Considerations	8
5. Security Considerations	8
6. Acknowledgements	8
7. References	9
7.1. Normative References	9
7.2. Informative References	9
Authors' Addresses	10

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 [I-D.ietf-spring-segment-routing-policy] 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 IGP's themselves to compute constraint based paths over the network.

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

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

The operations of the flexible algorithm solution is 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 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] forwarding plane, 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. This flex algorithm computation is within an IGP area or level similar to the default shortest path tree (SPT) algorithm.

The BGP-LS extensions for SR are defined in [I-D.ietf-idr-bgp-ls-segment-routing-ext] and includes the

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

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 Flexible Algorithm Definition 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 Prefix SIDs in its Segment List [I-D.ietf-spring-segment-routing-policy]. e.g. picking the Flex Algorithm Prefix SID or 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 Prefix-SIDs in the SID list.

2. BGP-LS Extensions for Flex Algo Definition

The BGP-LS [RFC7752] specifies the Node NLRI for advertisement of nodes and their attributes using the BGP-LS Attribute. The Flexible Algorithm Definition (FAD) advertised by a node are considered as its node level attributes and advertised as such.

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

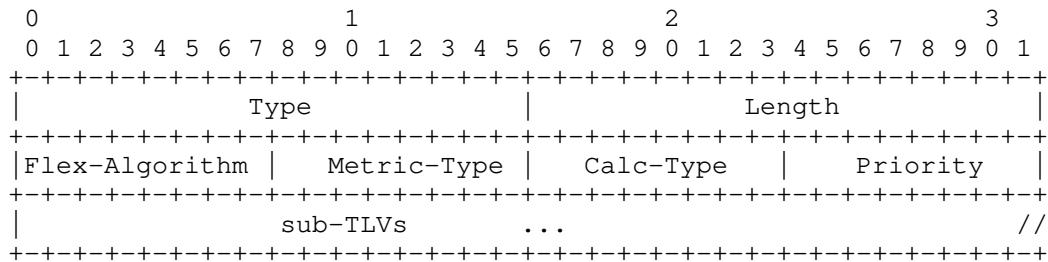


Figure 1: Flex Algorithm Definition TLV

where:

- o Type: TBD (see IANA Considerations Section 3)
- o Length: variable. Minimum of 8 octets.
- o 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].
- o Metric-Type : 1 octet value indicating the type of the 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].

- o 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].
- o Priority : 1 octet value between 0 and 255 inclusive that specifies the priority of the FAD.
- o 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..

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

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

2.1. 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 flex algo definition 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 TLV has the following format:

```

0               1               2               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Type                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Exclude-Any EAG (variable)          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
//
```

where:

- o Type: TBD (see IANA Considerations Section 3)
- o Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.
- o 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].

2.2. 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 flex algo definition 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 TLV has the following format:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     |                                     |
|               Type                 |               Length                 |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     |                                     |
|               Include-Any EAG (variable)                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

where:

- o Type: TBD (see IANA Considerations Section 3)
- o Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.
- o 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].

2.3. 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 flex algo definition 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 TLV has the following format:

```

      0                               1                               2                               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               |                               |
|                               Type                               Length
|                               |                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Include-All EAG (variable)                               //
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

where:

- o Type: TBD (see IANA Considerations Section 3)
- o Length: variable, dependent on the size of the Extended Admin Group. MUST be a multiple of 4 octets.
- o 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. IANA Considerations

This document requests assigning code-points from the registry "BGP-LS Node Descriptor, Link Descriptor, Prefix Descriptor, and Attribute TLVs" based on table below. 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
TBD	Flex Algorithm Definition TLV	variable
TBD	Flex Algo Exclude Any Affinity sub-TLV	variable
TBD	Flex Algo Include Any Affinity sub-TLV	variable
TBD	Flex Algo Include All Affinity sub-TLV	variable

4. Manageability Considerations

This section is structured as recommended in [RFC5706].

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 NLRI's attribute tests in the Fault Management section of [RFC7752] now encompass the new TLVs for the BGP-LS NLRI in this document.

4.1. Operational Considerations

No additional operation considerations are defined in this document.

4.2. Management Considerations

No additional management considerations are defined in this document.

5. Security 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 security model other than as discussed in the Security Considerations section of [RFC7752].

6. Acknowledgements

The authors would like to thank Les Ginsberg for his reviews and contributions to this work.

7. References

7.1. Normative References

- [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-01 (work in progress), November 2018.
- [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>>.

7.2. Informative References

- [I-D.ietf-idr-bgp-ls-segment-routing-ext]
Previdi, S., Talaulikar, K., Filsfils, C., Gredler, H., and M. Chen, "BGP Link-State extensions for Segment Routing", draft-ietf-idr-bgp-ls-segment-routing-ext-11 (work in progress), October 2018.
- [I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d., bogdanov@google.com, b., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-02 (work in progress), October 2018.
- [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>>.

- [RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", RFC 5706, DOI 10.17487/RFC5706, November 2009, <<https://www.rfc-editor.org/info/rfc5706>>.
- [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>>.
- [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>>.

Authors' Addresses

Ketan Talaulikar
Cisco Systems

Email: ketant@cisco.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