

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 20, 2018

P. Psenak, Ed.
Cisco Systems
S. Hegde, Ed.
Juniper Networks, Inc.
C. Filsfils
Cisco Systems, Inc.
A. Gulko
Thomson Reuters
February 16, 2018

ISIS Segment Routing Flexible Algorithm
draft-hegdepsenak-isis-sr-flex-algo-02.txt

Abstract

IGP protocols traditionally compute best paths over the network based on the IGP metric assigned to the links. Many network deployments use RSVP-TE based or Segment Routing based Traffic Engineering to enforce traffic over a path that is computed using different metrics or constraints than the shortest IGP path. Various mechanisms are used to steer the traffic towards such traffic engineered paths. This document proposes a solution that allows IGPs themselves to compute constraint based paths over the network without the use of the above mentioned traffic engineering technologies.

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 20, 2018.

Copyright Notice

Copyright (c) 2018 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
1.1.	Requirements notation	3
2.	Flexible Algorithm	3
3.	Flexible Algorithm Advertisement	3
4.	Flexible Algorithm Definition Advertisement	4
4.1.	Flexible Algorithm Definition Sub-TLV	4
4.2.	Flexible Algorithm Exclude Admin Group Sub-TLV	7
4.3.	Flexible Algorithm Include Admin Group Sub-TLVs	7
5.	Calculation of Flexible Algorithm Paths	8
6.	Backward Compatibility	10
7.	Security Considerations	10
8.	IANA Considerations	10
8.1.	Sub TLVs for Type 242	10
8.2.	New Sub-Sub-TLV registry	10
8.2.1.	Flexible Algorithm Definition TLV Metric Registry	11
9.	Acknowledgments	11
10.	References	12
10.1.	Normative References	12
10.2.	Informative References	12
	Authors' Addresses	13

1. Introduction

IGP computed path based on the shortest IGP metric must often be replaced by traffic engineered path due to the traffic requirements which are not reflected in the IGP metric. Some networks engineer the IGP metric assignments in a way that the IGP Metric reflects the link bandwidth or delay. If, for example, the IGP metric is reflecting the bandwidth on the link and the application traffic is delay sensitive, the best IGP path may not reflect the best path from such application's perspective.

To overcome such IGP limitation, various sorts of traffic engineering has been deployed, including RSVP-TE or SR-TE, in which case the TE component is responsible for computing the path based on additional metrics and/or constraints. Such paths need to be installed in the

forwarding and replace the original paths computed by IGPs. Tunnels are often used to represent the engineered paths and mechanisms like one described in [RFC3906] are used to replace the native IGP paths with such tunnel paths.

Segment Routing (SR) allows a flexible definition of end-to-end paths within IGP topologies by encoding paths as sequences of topological sub-paths, called segments. It also defines an algorithm that defines how the paths are computed. It also provides a way to associate Prefix-SID with an algorithm. This allows IGPs to compute paths based on various algorithms and cause traffic to be forwarded on such paths using the algorithm specific segments.

This document describes the IS-IS extension to support Segment Routing Flexible Algorithm on an MPLS data-plane.

1.1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Flexible Algorithm

Many possible constraints may be used to compute a path over a network. Some networks are deployed as multiple planes. A simple form of constraint may be to use a particular plane. A more sophisticated form of constraint can include some extended metric as described in [RFC7810]. Constraints which restrict paths to links with specific affinities or avoid links with specific affinities are also possible. Combinations of these are also possible.

To provide maximum flexibility we do not want to provide a strict mapping between the set of constraints and the algorithm that is associated with it. We want the mapping between the algorithm value and it's meaning to be flexible and defined by the user. As far as all routers in the domain have the common understanding what the particular algorithm value represents, the computation for such algorithm is consistent and traffic is not subject to any looping.

Because the meaning of the algorithm is not defined by any standard, but is defined by the user, we call it Flex-Algorithm.

3. Flexible Algorithm Advertisement

[I-D.ietf-isis-segment-routing-extensions] defines an SR-Algorithm. This algorithm defines how the best path is computed by the IGP. Routers advertise the support for the algorithm as a node capability.

Prefix SIDs are also advertised with an algorithm value and as such are tightly coupled with the algorithm.

Existing advertisement of the SR-Algorithm is used for the Flex-Algorithm advertisements as defined in [I-D.ietf-isis-segment-routing-extensions].

SR-Algorithm is a one octet value. We propose to split the range of values as follows:

0-127 - standardised values assigned by IANA

128-255 - user defined values.

4. Flexible Algorithm Definition Advertisement

To guarantee the loop free forwarding for paths computed for a particular Flex-Algorithm, all routers in the flooding scope of the algorithm definition MUST agree on the definition of the Flex-Algorithm.

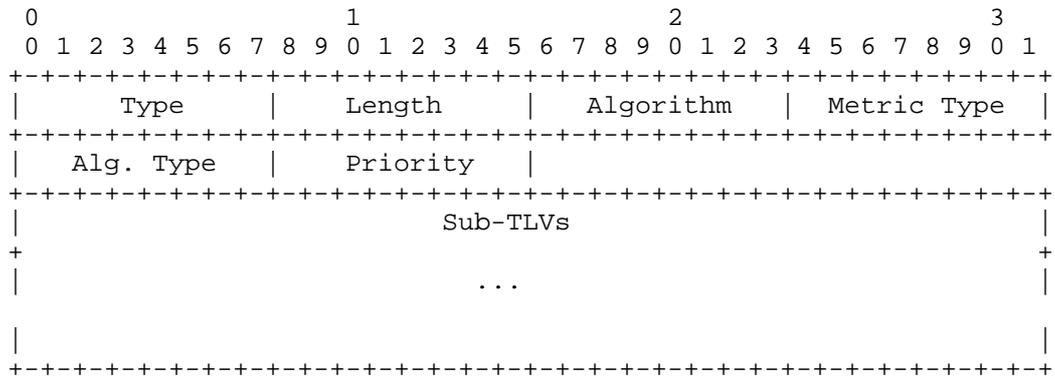
4.1. Flexible Algorithm Definition Sub-TLV

Flexible Algorithm Definition Sub-TLV (FAD Sub-TLV) is used to advertise the definition of the Flex-Algorithm.

FAD Sub-TLV is advertised as Sub-TLV of the IS-IS Router Capability TLV-242 that is defined in [RFC7981].

When the definition of the Flex Algorithm is advertised, it is applicable to all topologies supported on the receiving node.

FAD Sub-TLV has the following format:



where:

- Type: TBD1
- Length: variable, dependent on the included Sub-TLVs
- Algorithm: Flex-Algorithm number. Value between 128 and 255 inclusive.
- Metric Type: Type of metric to be used during the calculation. Following values are defined:
 - 0: IGP Metric
 - 1: Min Unidirectional Link Delay as defined in [RFC7810].
 - 2: TE default metric as defined in [RFC5305].
- Algorithm Type: Single octet identifying the algorithm type used to compute paths for the Flex-Algorithm. Values are defined in "IGP Algorithm Types" registry defined under "Interior Gateway Protocol (IGP) Parameters" IANA registries.
- Priority: Single octet that specifies the priority of the advertisement.
- Sub-TLVs - optional sub-TLVs.

When the router is configured with the local definition of the Flex-Algorithm, the router MUST advertise its local definition in the FAD Sub-TLV. If the local definition of the Flex-Algorithm is not advertised, the inconsistency in the configuration of the Flex-Algorithm on various nodes cannot be detected and traffic routed based on a Flex-Algorithm path may loop permanently.

Every router, that is configured to support a particular Flex-Algorithm, MUST select the Flex-Algorithm definition based on the following rules:

From the received advertisements of the FAD, select the one(s) with the highest priority.

If there are multiple advertisements of the FAD with the same highest priority, select the one that is originated from the router with the highest Router ID. Router ID is required to be advertised in every Router Capability TLV [RFC7981].

If the router has a local definition of the Flex-Algorithm, compare it with the received FAD advertisements using the same rules as have been used to pick the best FAD advertisement, e.g., priority and Router ID.

A router that is not configured to support a particular Flex-Algorithm MUST ignore FAD Sub-TLVs advertisements for such Flex-Algorithm.

Having a deterministic way that always produces a valid Flex-Algorithm definition avoids conflicts and maximizes the availability of the forwarding for the traffic that is using the Flex-Algorithm paths.

Any change in the Flex-Algorithm definition may result in temporary disruption of traffic that is forwarded based on such Flex-Algorithm paths. The impact is similar to any other event that requires network wide convergence

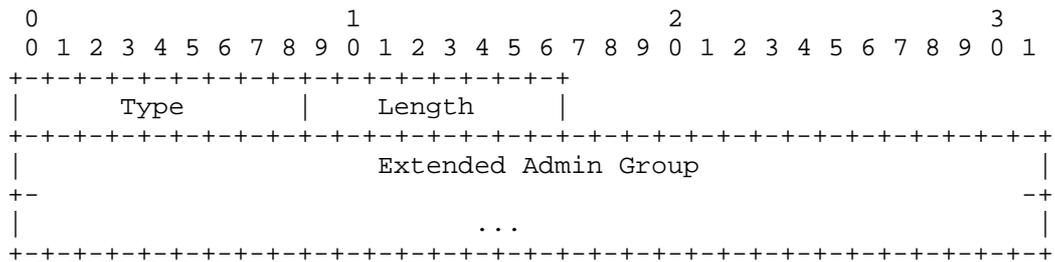
The FAD Sub-TLV of the IS-IS Router Capability TLV-242 MUST be propagated throughout the level. It MAY be advertised across level boundaries, if the S-flag in the Router Capability TLV is set. The S-Flag SHOULD not be set by default unless local configuration policy on the originating router indicates domain wide flooding.

Flex-Algorithm definition is topology independent. A node which advertises support for a given Flex-Algorithm may support that Flex-Algorithm on any subset of the topologies it supports. Enabling of a supported Flex-Algorithm on a given topology is a matter of local configuration. For a given topology, if out of the set of nodes supporting that topology AND advertising support for a given Flex-Algorithm only a subset of the nodes actually compute/install Flex-Algorithm specific paths in the forwarding plane for that topology, some traffic intended for such topology/Flex-Algorithm could be dropped if forwarded to a node on which the Flex-Algorithm is not enabled on that topology.

4.2. Flexible Algorithm Exclude Admin Group Sub-TLV

The Flexible-Algorithm definition can specify 'colors' that are used by the operator to exclude links during the Flex-Algorithm path computation.

Flexible Algorithm Exclude Admin Group Sub-TLV (FAEAG Sub-TLV) is a Sub-TLV of the FAD Sub-TLV. It has the following format:



where:

Type: 1

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

Extended Administrative Group: Extended Administrative Group as defined in [RFC7308].

FAEAG Sub-TLV SHOULD only appear once in FAD Sub-TLV. If it appears more than once, FAD Sub-TLV MUST be ignored by the receiver.

4.3. Flexible Algorithm Include Admin Group Sub-TLVs

The Flexible-Algorithm definition can specify 'colors' that are used by the operator to include links during the Flex-Algorithm path computation.

The format of the include Sub-TLVs is identical to the format of the FAEAG Sub-TLV in Section 4.2.

Two forms of inclusion are available - include-any and include-all.

Flexible Algorithm Include-Any Admin Group Sub-TLV - Type 2.

Flexible Algorithm Include-All Admin Group Sub-TLV - Type 3.

Flexible Algorithm Include Admin Group Sub-TLVs SHOULD only appear once in FAD Sub-TLV. If any of these Sub-TLVs appear more than once, FAD Sub-TLV MUST be ignored by the receiver.

5. Calculation of Flexible Algorithm Paths

A router may compute path for multiple Flex-Algorithms.

A router MUST be configured to support Flex-Algorithm K before it can compute any path for Flex-Algorithm K.

A router MUST either be configured with a local definition of Flex-Algorithm K or receive the definition via the FAD Sub-TLV, as described in Section 4.1, before it can compute any path for Flex-Algorithm K.

When computing the path for Flex-Algorithm K, all nodes that do not advertise support for Flex-Algorithm K in SR-Algorithm Sub-TLV ([I-D.ietf-isis-segment-routing-extensions]), MUST be pruned from the topology.

When computing the path for Flex-Algorithm K, the metric that is part of the Flex-Algorithm definition (Section 4.1) MUST be used.

Various link include or exclude rules can be part of the Flex-Algorithm definition. These rules use Extended Administrative Groups (EAG) as defined in [RFC7308]. [RFC7308] uses term 'colors' as a shorthand to refer to particular bits with an EAG. Link advertisement CAN also include EAG, which describe which color is set on the link.

Link advertisement CAN also include Administrative Group (AG) TLV ([RFC5305]). The coexistence of EAG and AG is described in the section 2.3.1 of [RFC7308].

Rules, in the order as specified below, MUST be used to prune link from the topology during the Flex-Algorithm computation.

For all links in the topology:

1. Check if any exclude rule is part of the Flex-Algorithm definition. If such exclude rule exists, check if any color that is part of the exclude rule is also set on the link. If such a color exist, the link MUST be pruned from the computation.
2. Check if any include-any rule is part of the Flex-Algorithm definition. if such include-any rule exists, check if any color that is part of the include-any rule is also set on the link. If

such color does not exist, the link MUST be pruned from the computation.

3. Check if any include-all rule is part of the Flex-Algorithm definition. If such include-all rule exists, check if all colors that are part of the include-all rule are also set on the link. If not all such colors are set on the link, the link MUST be pruned from the computation.

4. If the Flex-Algorithm definition uses other than IGP metric (Section 4.1), and such metric is not advertised for the particular link in a topology for which the computation is done, such link MUST be pruned from the computation. A metric of value 0 MUST NOT be assumed in such case.

Flex-Algorithm K path MUST be installed in the MPLS forwarding plane using the MPLS label that corresponds to the Prefix-SID that was advertised for algorithm K. If the Prefix SID for algorithm K is not known, the Flex-Algorithm K path to such prefix MUST NOT be installed in the MPLS forwarding plane.

Loop Free Alternate (LFA) paths for Flex-Algorithm K path MUST be computed using the same constraints as the calculation of the primary paths for Flex-Algorithm K. LFA path MUST only use Prefix-SIDs advertised specifically for algorithm K to enforce the traffic over such path. LFA path MUST NOT use Adjacency-SID that belong to the link that has been pruned from the computation.

If LFA protection is being used to protect Flex-Algorithm K paths, all routers in the area SHOULD advertise at least one Flex-Algorithm K specific Prefix-SID. These Prefix-SIDs are used to enforce traffic over the LFA computed backup path.

Flex-Algorithm paths MAY be used by other applications, that do not utilize MPLS forwarding plane. It is outside of the scope of this specification, how these application learn and use the Flex-Algorithm specific paths.

Any Shortest Path Tree calculation is limited to a single area. Same applies to Flex-Algorithm calculations. Given that the computing router may not have the visibility to the topology of remote areas, the Flex-Algorithm K path to an inter-area prefix will only be computed for the local area. The egress L1/L2 router will be selected based on the best path for the Flex-Algorithm K in the local area and such egress L1/L2 router will be responsible to compute the best Flex-Algorithm K path over the next area. This may produce end-to-end path, which is not the best from the Flex-Algorithm K perspective. If the best end-to-end path for Flex-Algorithm K needs

to be used for inter-area destinations, paths for such destinations need to be computed by the entity that has the topological information about all areas.

6. Backward Compatibility

This extension brings no new backward compatibility issues.

7. Security Considerations

This extension adds no new security considerations.

8. IANA Considerations

This documents request allocation for the following ISIS TLVs and subTLVs.

8.1. Sub TLVs for Type 242

This document makes the following registrations in the "sub-TLVs for TLV 242" registry.

Type: TBD1 (suggested value 24).

Description: Flexible Algorithm Definition Sub-TLV.

Reference: This document (Section 4.1).

8.2. New Sub-Sub-TLV registry

This document creates the following Sub-TLV Registry:

Registry: Sub-TLVs for Flexible Algorithm Definition Sub-TLV

Registration Procedure: Expert review

Reference: This document (Section 4.1)

This document registers following Sub-TLVs in the "Sub-TLVs for Flexible Algorithm Definition Sub-TLV" registry:

Type: 1

Description: Flexible Algorithm Exclude Admin Group Sub-TLV

Reference: This document (Section 4.2).

Type: 2

Description: Flexible Algorithm Include-Any Admin Group Sub-TLV

Reference: This document (Section 4.3).

Type: 3

Description: Flexible Algorithm Include-All Admin Group Sub-TLV

Reference: This document (Section 4.3).

8.2.1. Flexible Algorithm Definition TLV Metric Registry

This document creates the following Registry:

Registry: Flexible Algorithm Definition TLV Metric Registry

Registration Procedure: Expert review

Reference: This document (Section 4.1)

This document registers following values in the "Flexible Algorithm Definition TLV Metric Registry":

Type: TBD, suggested value 0

Description: IGP metric

Reference: This document (Section 4.1)

Type: TBD, suggested value 1

Description: Min Unidirectional Link Delay [RFC7810]

Reference: This document (Section 4.1)

Type: TBD, suggested value 2

Description: TE Default Metric [RFC5305]

Reference: This document (Section 4.1)

9. Acknowledgments

This draft, among other things, is also addressing the problem that the [I-D.gulkohegde-routing-planes-using-sr] was trying to solve. All authors of that draft agreed to join this draft.

Thanks to Les Ginsberg and Ketan Talaulikar for review and useful comments.

Thanks to Cengiz Halit for his review and feedback during initial phase of the solution definition.

10. References

10.1. Normative References

- [I-D.ietf-isis-segment-routing-extensions]
Previdi, S., Ginsberg, L., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and J. Tantsura, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-15 (work in progress), December 2017.
- [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>>.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <<https://www.rfc-editor.org/info/rfc5305>>.
- [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>>.
- [RFC7810] Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 7810, DOI 10.17487/RFC7810, May 2016, <<https://www.rfc-editor.org/info/rfc7810>>.
- [RFC7981] Ginsberg, L., Previdi, S., and M. Chen, "IS-IS Extensions for Advertising Router Information", RFC 7981, DOI 10.17487/RFC7981, October 2016, <<https://www.rfc-editor.org/info/rfc7981>>.

10.2. Informative References

- [I-D.gulkohegde-routing-planes-using-sr]
Hegde, S. and a. arkadiy.gulko@thomsonreuters.com, "Separating Routing Planes using Segment Routing", draft-gulkohegde-routing-planes-using-sr-00 (work in progress), March 2017.

[RFC3906] Shen, N. and H. Smit, "Calculating Interior Gateway Protocol (IGP) Routes Over Traffic Engineering Tunnels", RFC 3906, DOI 10.17487/RFC3906, October 2004, <<https://www.rfc-editor.org/info/rfc3906>>.

Authors' Addresses

Peter Psenak (editor)
Cisco Systems
Apollo Business Center
Mlynske nivy 43
Bratislava, 82109
Slovakia

Email: ppsenak@cisco.com

Shraddha Hegde (editor)
Juniper Networks, Inc.
Embassy Business Park
Bangalore, KA, 560093
India

Email: shraddha@juniper.net

Clarence Filsfils
Cisco Systems, Inc.
Brussels
Belgium

Email: cfilsfil@cisco.com

Arkadiy Gulko
Thomson Reuters

Email: arkadiy.gulko@thomsonreuters.com