Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 10, 2021

P. Psenak, Ed.
Cisco Systems
S. Hegde
Juniper Networks, Inc.
C. Filsfils
K. Talaulikar
Cisco Systems, Inc.
A. Gulko
Individual

October 7, 2020

IGP Flexible Algorithm
draft-ietf-lsr-flex-algo-12

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 steer traffic over a path that is computed using different metrics or constraints than the shortest IGP path. This document proposes a solution that allows IGPs themselves to compute constraint-based paths over the network. This document also specifies a way of using Segment Routing (SR) Prefix-SIDs and SRv6 locators to steer packets along the constraint-based paths.

Copyright Notice

Table of Contents

1. Introduction

An IGP-computed path based on the shortest IGP metric must often be replaced by a traffic-engineered path due to the traffic requirements which are not reflected by 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 an application's perspective.

To overcome this limitation, various sorts of traffic engineering have been deployed, including RSVP-TE and SR-TE, in which case the TE component is responsible for computing paths based on additional metrics and/or constraints. Such paths need to be installed in the forwarding tables in addition to, or as a replacement for, 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.

This document specifies a set of extensions to ISIS, OSPFv2, and OSPFv3 that enable a router to advertise TLVs that identify (a) calculation-type, (b) specify a metric-type, and (c) describe a set of constraints on the topology, that are to be used to compute the best paths along the constrained topology. A given combination of calculation-type, metric-type, and constraints is known as a "Flexible Algorithm Definition". A router that sends such a set of TLVs also assigns a Flex-Algorithm value to the specified combination of calculation-type, metric-type, and constraints.

This document also specifies a way for a router to use IGPs to associate one or more SR Prefix-SIDs or SRv6 locators with a particular Flex-Algorithm. Each such Prefix-SID or SRv6 locator then represents a path that is computed according to the identified Flex-Algorithm.

2. 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 [BCP14] [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

This section defines terms that are often used in this document.

Flexible Algorithm Definition (FAD) - the set consisting of (a) calculation-type, (b) metric-type, and (c) a set of constraints.

Flexible Algorithm - a numeric identifier in the range 128-255 that is associated via configuration with the Flexible-Algorithm Definition.

Local Flexible Algorithm Definition - Flexible Algorithm Definition defined locally on the node.

Remote Flexible Algorithm Definition - Flexible Algorithm Definition received from other nodes via IGP flooding.

Flexible Algorithm Participation - per application configuration state that expresses whether the node is participating in a particular Flexible Algorithm.

IGP Algorithm - value from the the "IGP Algorithm Types" registry defined under "Interior Gateway Protocol (IGP) Parameters" IANA registries. IGP Algorithms represents the triplet (Calculation Type, Metric, Constraints), where the second and third elements of the triple MAY be unspecified.

ABR - Area Border Router. In ISIS terminology it is also known as L1/L2 router.

ASBR - Autonomous System Border Router.

4. 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 [RFC8570]. 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 want to provide a mechanism that allows a router to (a) identify a particular calculation-type, (b) metric-type, (c) describe a particular set of constraints, and (d) assign a numeric identifier, referred to as Flex-Algorithm, to the combination of that calculation-type, metric-type, and those constraints. We want the mapping between the Flex-Algorithm and its meaning to be flexible and defined by the user. As long as all routers in the domain have a common understanding as to what a particular Flex-Algorithm represents, the resulting routing computation is consistent and traffic is not subject to any looping.

The set consisting of (a) calculation-type, (b) metric-type, and (c) a set of constraints is referred to as a Flexible-Algorithm Definition.

Flexible-Algorithm is a numeric identifier in the range 128-255 that is associated via configuratin with the Flexible-Algorithm Definition.

IANA "IGP Algorithm Types" registry defines the set of values for IGP Algorithms. We propose to allocate the following values for Flex-Algorithms from this registry:

128-255 - Flex-Algorithms

5. Flexible Algorithm Definition Advertisement

To guarantee the loop-free forwarding for paths computed for a particular Flex-Algorithm, all routers that (a) are configured to participate in a particular Flex-Algorithm, and (b) are in the same Flex-Algorithm definition advertisement scope MUST agree on the definition of the Flex-Algorithm.

5.1. ISIS Flexible Algorithm Definition Sub-TLV

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

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

ISIS FAD Sub-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    |Flex-Algorithm |  Metric-Type  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Calc-Type   |    Priority   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Sub-TLVs                             |
+                                                               +
|                            ...                                |

|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 26

Length: variable, dependent on the included Sub-TLVs

Flex-Algorithm: Single octet 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 [RFC8570], section 4.2, encoded as application specific link attribute as

specified in [I-D.ietf-isis-te-app] and Section 11 of this document.

2: Traffic Engineering Default Metric as defined in [RFC5305], section 3.7, encoded as application specific link attribute as specified in [I-D.ietf-isis-te-app] and Section 11 of this document.

Calc-Type: value from 0 to 127 inclusive from the "IGP Algorithm Types" registry defined under "Interior Gateway Protocol (IGP) Parameters" IANA registries. IGP algorithms in the range of 0-127 have a defined triplet (Calculation Type, Metric, Constraints). When used to specify the Calc-Type in the FAD Sub-TLV, only the Calculation Type defined for the specified IGP Algorithm is used. The Metric/Constraints MUST NOT be inherited. If the required calculation type is Shortest Path First, the value 0 SHOULD appear in this field.

Priority: Value between 0 and 255 inclusive that specifies the priority of the advertisement.

Sub-TLVs - optional sub-TLVs.

The ISIS FAD Sub-TLV MAY be advertised in an LSP of any number, but a router MUST NOT advertise more than one ISIS FAD Sub-TLV for a given Flexible-Algorithm. A router receiving multiple ISIS FAD Sub-TLVs for a given Flexible-Algorithm from the same originator SHOULD select the first advertisement in the lowest numbered LSP.

The ISIS FAD Sub-TLV has an area scope. The Router Capability TLV in which the FAD Sub-TLV is present MUST have the S-bit clear.

ISIS L1/L2 router MAY be configured to re-generate the winning FAD from level 2, without any modification to it, to level 1 area. The re-generation of the FAD Sub-TLV from level 2 to level 1 is determined by the L1/L2 router, not by the originator of the FAD advertisement in the level 2. In such case, the re-generated FAD Sub-TLV will be advertised in the level 1 Router Capability TLV originated by the L1/L2 router.

L1/L2 router MUST NOT re-generate any FAD Sub-TLV from level 1 to level 2.

5.2. OSPF Flexible Algorithm Definition TLV

OSPF FAD TLV is advertised as a top-level TLV of the RI LSA that is defined in [RFC7770].

OSPF FAD 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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Flex-Algorithm |   Metric-Type |   Calc-Type   |    Priority   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Sub-TLVs                            |
+                                                               +
|                              ...                              |

|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 16

Length: variable, dependent on the included Sub-TLVs

Flex-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 [RFC7471], section 4.2, encoded as application specific link attribute as specified in [I-D.ietf-ospf-te-link-attr-reuse] and Section 11 of this document.

2: Traffic Engineering metric as defined in [RFC3630], section 2.5.5, encoded as application specific link attribute as specified in [I-D.ietf-ospf-te-link-attr-reuse] and Section 11 of this document.

Calc-Type: as described in Section 5.1

Priority: as described in Section 5.1

Sub-TLVs - optional sub-TLVs.

When multiple OSPF FAD TLVs, for the same Flexible-Algorithm, are received from a given router, the receiver MUST use the first occurrence of the TLV in the Router Information LSA. If the OSPF FAD TLV, for the same Flex-Algorithm, appears in multiple Router Information LSAs that have different flooding scopes, the OSPF FAD TLV in the Router Information LSA with the area-scoped flooding scope MUST be used. If the OSPF FAD TLV, for the same algorithm, appears in multiple Router Information LSAs that have the same flooding scope, the OSPF FAD TLV in the Router Information (RI) LSA with the numerically smallest Instance ID MUST be used and subsequent instances of the OSPF FAD TLV MUST be ignored.

The RI LSA can be advertised at any of the defined opaque flooding scopes (link, area, or Autonomous System (AS)). For the purpose of OSPF FAD TLV advertisement, area-scoped flooding is REQUIRED. The Autonomous System flooding scope SHOULD NOT be used by default unless local configuration policy on the originating router indicates domain wide flooding.

5.3. Common Handling of Flexible Algorithm Definition TLV

This section describes the protocol-independent handling of the FAD TLV (OSPF) or FAD Sub-TLV (ISIS). We will refer to it as FAD TLV in this section, even though in case of ISIS it is a Sub-TLV.

The value of the Flex-Algorithm MUST be between 128 and 255 inclusive. If it is not, the FAD TLV MUST be ignored.

Only a subset of the routers participating in the particular Flex-Algorithm need to advertise the definition of the Flex-Algorithm.

Every router, that is configured to participate in a particular Flex-Algorithm, MUST select the Flex-Algorithm definition based on the following ordered rules. This allows for the consistent Flex-Algorithm definition selection in cases where different routers advertise different definitions for a given Flex-Algorithm:

1. From the advertisements of the FAD in the area (including both locally generated advertisements and received advertisements) select the one(s) with the highest priority value.

2. 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 System-ID, in the case of ISIS, or Router ID, in the case of OSPFv2 and OSPFv3. For ISIS, the System-ID is described in [ISO10589]. For OSPFv2 and OSPFv3, standard Router ID is described in [RFC2328] and [RFC5340] respectively.

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

A router that is not participating in a particular Flex-Algorithm is allowed to advertise FAD for such Flex-Algorithm. Receiving routers MUST consider FAD advertisement regardless of the Flex-Algorithm participation of the FAD originator.

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.

If a node is configured to participate in a particular Flexible-Algorithm, but the selected Flex-Algorithm definition includes calculation-type, metric-type, constraint, flag, or Sub-TLV that is not supported by the node, it MUST stop participating in such Flexible-Algorithm. That implies that it MUST NOT announce participation for such Flexible-Algorithm as specified in Section 10 and it MUST remove any forwarding state associated with it.

Flex-Algorithm definition is topology independent. It applies to all topologies that a router participates in.

6. Sub-TLVs of ISIS FAD Sub-TLV

6.1. ISIS 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.

The ISIS Flexible Algorithm Exclude Admin Group Sub-TLV is used to advertise the exclude rule that is used during the Flex-Algorithm path calculation as specified in Section 12.

The ISIS Flexible Algorithm Exclude Admin Group Sub-TLV (FAEAG Sub-TLV) is a Sub-TLV of the ISIS FAD Sub-TLV. It 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     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                      Extended Admin Group                     |
    +-                                                             -+
    |                            ...                                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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].

The ISIS FAEAG Sub-TLV MUST NOT appear more then once in an ISIS FAD Sub-TLV. If it appears more then once, the ISIS FAD Sub-TLV MUST be ignored by the receiver.

6.2. ISIS Flexible Algorithm Include-Any Admin Group Sub-TLV

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

The ISIS Flexible Algorithm Include-Any Admin Group Sub-TLV is used to advertise include-any rule that is used during the Flex-Algorithm path calculation as specified in Section 12.

The format of the ISIS Flexible Algorithm Include-Any Admin Group Sub-TLV is identical to the format of the FAEAG Sub-TLV in Section 6.1.

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

The ISIS Flexible Algorithm Include-Any Admin Group Sub-TLV MUST NOT appear more then once in an ISIS FAD Sub-TLV. If it appears more then once, the ISIS FAD Sub-TLV MUST be ignored by the receiver.

6.3. ISIS Flexible Algorithm Include-All Admin Group Sub-TLV

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

The ISIS Flexible Algorithm Include-All Admin Group Sub-TLV is used to advertise include-all rule that is used during the Flex-Algorithm path calculation as specified in Section 12.

The format of the ISIS Flexible Algorithm Include-All Admin Group Sub-TLV is identical to the format of the FAEAG Sub-TLV in Section 6.1.

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

The ISIS Flexible Algorithm Include-All Admin Group Sub-TLV MUST NOT appear more then once in an ISIS FAD Sub-TLV. If it appears more then once, the ISIS FAD Sub-TLV MUST be ignored by the receiver.

6.4. ISIS Flexible Algorithm Definition Flags Sub-TLV

The ISIS Flexible Algorithm Definition Flags Sub-TLV (FADF Sub-TLV) is a Sub-TLV of the ISIS FAD Sub-TLV. It 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     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                             |
+-                                                             -+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 4

Length: variable, non-zero number of octets of the Flags field

Flags:

```
 0 1 2 3 4 5 6 7...
+-+-+-+-+-+-+-+-+...
|M| | |          ...
+-+-+-+-+-+-+-+-+...
```

M-flag: when set, the Flex-Algorithm specific prefix metric MUST be used, if advertised with the prefix. This flag is not applicable to prefixes advertised as SRv6 locators.

Bits are defined/sent starting with Bit 0 defined above. Additional bit definitions that may be defined in the future SHOULD be assigned in ascending bit order so as to minimize the number of bits that will need to be transmitted.

Undefined bits MUST be transmitted as 0.

Bits that are NOT transmitted MUST be treated as if they are set to 0 on receipt.

The ISIS FADF Sub-TLV MUST NOT appear more then once in an ISIS FAD Sub-TLV. If it appears more then once, the ISIS FAD Sub-TLV MUST be ignored by the receiver.

If the ISIS FADF Sub-TLV is not present inside the ISIS FAD Sub-TLV, all the bits are assumed to be set to 0.

6.5. ISIS Flexible Algorithm Exclude SRLG Sub-TLV

The Flexible Algorithm definition can specify Shared Risk Link Groups (SRLGs) that the operator wants to exclude during the Flex-Algorithm path computation.

The ISIS Flexible Algorithm Exclude SRLG Sub-TLV (FAESRLG) is used to advertise the exclude rule that is used during the Flex-Algorithm path calculation as specified in Section 12.

The ISIS FAESRLG Sub-TLV is a Sub-TLV of the ISIS FAD Sub-TLV. It 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     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Shared Risk Link Group Value             |
+-                                                             -+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 5

Length: variable, dependent on number of SRLG values. MUST be a multiple of 4 octets.

Shared Risk Link Group Value: SRLG value as defined in [RFC5307].

The ISIS FAESRLG Sub-TLV MUST NOT appear more then once in an ISIS FAD Sub-TLV. If it appears more then once, the ISIS FAD Sub-TLV MUST be ignored by the receiver.

7. Sub-TLVs of OSPF FAD TLV

7.1. OSPF Flexible Algorithm Exclude Admin Group Sub-TLV

The Flexible Algorithm Exclude Admin Group Sub-TLV (FAEAG Sub-TLV) is a Sub-TLV of the OSPF FAD TLV. It's usage is described in Section 6.1. It 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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Extended Admin Group                     |
+-                                                             -+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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].

The OSPF FAEAG Sub-TLV MUST NOT appear more then once in an OSPF FAD TLV. If it appears more then once, the OSPF FAD TLV MUST be ignored by the receiver.

7.2. OSPF Flexible Algorithm Include-Any Admin Group Sub-TLV

The usage of this Sub-TLVs is described in Section 6.2.

The format of the OSPF Flexible Algorithm Include-Any Admin Group Sub-TLV is identical to the format of the OSPF FAEAG Sub-TLV in Section 7.1.

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

The OSPF Flexible Algorithm Include-Any Admin Group Sub-TLV MUST NOT appear more then once in an OSPF FAD TLV. If it appears more then once, the OSPF FAD TLV MUST be ignored by the receiver.

7.3. OSPF Flexible Algorithm Include-All Admin Group Sub-TLV

The usage of this Sub-TLVs is described in Section 6.3.

The format of the OSPF Flexible Algorithm Include-Any Admin Group Sub-TLV is identical to the format of the OSPF FAEAG Sub-TLV in Section 7.1.

The OSPF Flexible Algorithm Include-Any Admin Group Sub-TLV Type is 3.

The OSPF Flexible Algorithm Include-All Admin Group Sub-TLV MUST NOT appear more then once in an OSPF FAD TLV. If it appears more then once, the OSPF FAD TLV MUST be ignored by the receiver.

7.4. OSPF Flexible Algorithm Definition Flags Sub-TLV

The OSPF Flexible Algorithm Definition Flags Sub-TLV (FADF Sub-TLV) is a Sub-TLV of the OSPF FAD TLV. It 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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                             |
+-                                                             -+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 4

Length: variable, dependent on the size of the Flags field. MUST be a multiple of 4 octets.

Flags:

```
 0 1 2 3 4 5 6 7...
+-+-+-+-+-+-+-+-+...
|M| | |          ...
+-+-+-+-+-+-+-+-+...
```

M-flag: when set, the Flex-Algorithm specific prefix metric MUST be used, if advertised with the prefix. This flag is not applicable to prefixes advertised as SRv6 locators.

Bits are defined/sent starting with Bit 0 defined above. Additional bit definitions that may be defined in the future SHOULD be assigned in ascending bit order so as to minimize the number of bits that will need to be transmitted.

Undefined bits MUST be transmitted as 0.

Bits that are NOT transmitted MUST be treated as if they are set to 0 on receipt.

The OSPF FADF Sub-TLV MUST NOT appear more then once in an OSPF FAD TLV. If it appears more then once, the OSPF FAD TLV MUST be ignored by the receiver.

If the OSPF FADF Sub-TLV is not present inside the OSPF FAD TLV, all the bits are assumed to be set to 0.

7.5. OSPF Flexible Algorithm Exclude SRLG Sub-TLV

The OSPF Flexible Algorithm Exclude SRLG Sub-TLV (FAESRLG Sub-TLV) is a Sub-TLV of the OSPF FAD TLV. Its usage is described in Section 6.5. It 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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Shared Risk Link Group Value                |
+-                                                             -+
|                            ...                                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 5

Length: variable, dependent on the number of SRLGs. MUST be a multiple of 4 octets.

Shared Risk Link Group Value: SRLG value as defined in [RFC4203].

The OSPF FAESRLG Sub-TLV MUST NOT appear more then once in an OSPF FAD TLV. If it appears more then once, the OSPF FAD TLV MUST be ignored by the receiver.

8. ISIS Flexible Algorithm Prefix Metric Sub-TLV

The ISIS Flexible Algorithm Prefix Metric (FAPM) Sub-TLV supports the advertisement of a Flex-Algorithm specific prefix metric associated with a given prefix advertisement.

The ISIS FAPM Sub-TLV is a sub-TLV of TLVs 135, 235, 236, and 237 and 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     |Flex-Algorithm |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Metric                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 6

Length: 5 octets

Flex-Algorithm: Single octet value between 128 and 255 inclusive.

Metric: 4 octets of metric information

The ISIS FAPM Sub-TLV MAY appear multiple times in its parent TLV. If it appears more then once with the same Flex-Algorithm value, the first instance MUST be used and any subsequent instances MUST be ignored.

If a prefix is advertised with a Flex-Algorithm prefix metric larger then MAX_PATH_METRIC as defined in [RFC5305] this prefix MUST NOT be considered during the Flexible-Algorithm computation.

The usage of the Flex-Algorithm prefix metric is described in Section 12.

The ISIS FAPM Sub-TLV MUST NOT be advertised as a sub-TLV of the ISIS SRv6 Locator TLV [I-D.ietf-lsr-isis-srv6-extensions]. The ISIS SRv6 Locator TLV includes the Algorithm and Metric fields which MUST be used instead. If the FAPM Sub-TLV is present as a sub-TLV of the

ISIS SRv6 Locator TLV in the received LSP, such FAPM Sub-TLV MUST be ignored.

9. OSPF Flexible Algorithm Prefix Metric Sub-TLV

The OSPF Flexible Algorithm Prefix Metric (FAPM) Sub-TLV supports the advertisement of a Flex-Algorithm specific prefix metric associated with a given prefix advertisement.

The OSPF Flex-Algorithm Prefix Metric (FAPM) Sub-TLVis a Sub-TLV of the:

- OSPFv2 Extended Prefix TLV [RFC7684]

- Following OSPFv3 TLVs as defined in [RFC8362]:

  Intra-Area Prefix TLV

  Inter-Area Prefix TLV

  External Prefix TLV

OSPF FAPM Sub-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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Flex-Algorithm |                  Reserved                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Metric                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where:

Type: 3 for OSPFv2, 26 for OSPFv3

Length: 8 octets

Flex-Algorithm: Single octet value between 128 and 255 inclusive.

Reserved: Must be set to 0, ignored at reception.

Metric: 4 octets of metric information

The OSPF FAPM Sub-TLV MAY appear multiple times in its parent TLV. If it appears more then once with the same Flex-Algorithm value, the

first instance MUST be used and any subsequent instances MUST be ignored.

The usage of the Flex-Algorithm prefix metric is described in Section 12.

10. Advertisement of Node Participation in a Flex-Algorithm

When a router is configured to support a particular Flex-Algorithm, we say it is participating in that Flex-Algorithm.

Paths computed for a specific Flex-Algorithm MAY be used by various applications, each potentially using its own specific data plane for forwarding traffic over such paths. To guarantee the presence of the application specific forwarding state associated with a particular Flex-Algorithm, a router MUST advertise its participation for a particular Flex-Algorithm for each application specifically.

10.1. Advertisement of Node Participation for Segment Routing

[RFC8667], [RFC8665], and [RFC8666] (IGP Segment Routing extensions) describe how the SR-Algorithm is used to compute the IGP best path.

Routers advertise the support for the SR-Algorithm as a node capability as described in the above mentioned IGP Segment Routing extensions. To advertise participation for a particular Flex-Algorithm for Segment Routing, including both SR MPLS and SRv6, the Flex-Algorithm value MUST be advertised in the SR-Algorithm TLV (OSPF) or sub-TLV (ISIS).

Segment Routing Flex-Algorithm participation advertisement is topology independent. When a router advertises participation in an SR-Algorithm, the participation applies to all topologies in which the advertising node participates.

10.2. Advertisement of Node Participation for Other Applications

This section describes considerations related to how other applications can advertise their participation in a specific Flex-Algorithm.

Application-specific Flex-Algorithm participation advertisements MAY be topology specific or MAY be topology independent, depending on the application itself.

Application-specific advertisement for Flex-Algorithm participation MUST be defined for each application and is outside of the scope of this document.

11. Advertisement of Link Attributes for Flex-Algorithm

Various link attributes may be used during the Flex-Algorithm path calculation. For example, include or exclude rules based on link affinities can be part of the Flex-Algorithm definition as defined in Section 6 and Section 7.

Link attribute advertisements that are to be used during Flex-Algorithm calculation MUST use the Application-Specific Link Attribute (ASLA) advertisements defined in [I-D.ietf-isis-te-app] or [I-D.ietf-ospf-te-link-attr-reuse]. In the case of IS-IS, this includes use of the L-flag as defined in [I-D.ietf-isis-te-app] Section 4.2 subject to the constraints discussed in Section 6 of the [I-D.ietf-isis-te-app]. The mandatory use of ASLA advertisements applies to link attributes specifically mentioned in this document (Min Unidirectional Link Delay, TE Default Metric, Administrative Group, Extended Administrative Group and Shared Risk Link Group) and any other link attributes that may be used in support of Flex-Algorithm in the future.

A new Application Identifier Bit is defined to indicate that the ASLA advertisement is associated with the Flex-Algorithm application. This bit is set in the Standard Application Bit Mask (SABM) defined in [I-D.ietf-isis-te-app] or [I-D.ietf-ospf-te-link-attr-reuse]:

Bit-3: Flexible Algorithm (X-bit)

ASLA Admin Group Advertisements to be used by the Flexible Algorithm Application MAY use either the Administrative Group or Extended Administrative Group encodings. If the Administrative Group encoding is used, then the first 32 bits of the corresponding FAD sub-TLVs are mapped to the link attribute advertisements as specified in RFC 7308.

12. Calculation of Flexible Algorithm Paths

A router MUST be configured to participate in a given Flex-Algorithm K and MUST select the FAD based on the rules defined in Section 5.3 before it can compute any path for that Flex-Algorithm.

As described in Section 10, participation for any particular Flex-Algorithm MUST be advertised on a per-application basis. Calculation of the paths for any particular Flex-Algorithm MUST be application specific.

The way applications handle nodes that do not participate in Flexible-Algorithm is application specific. If the application only wants to consider participating nodes during the Flex-Algorithm calculation, then when computing paths for a given Flex-Algorithm,

all nodes that do not advertise participation for that Flex-Algorithm in their application-specific advertisements MUST be pruned from the topology. Segment Routing, including both SR MPLS and SRv6, are applications that MUST use such pruning when computing Flex-Algorithm paths.

When computing the path for a given Flex-Algorithm, the metric-type that is part of the Flex-Algorithm definition (Section 5) MUST be used.

When computing the path for a given Flex-Algorithm, the calculation-type that is part of the Flex-Algorithm definition (Section 5) MUST be used.

Various link include or exclude rules can be part of the Flex-Algorithm definition. To refer to a particular bit within an AG or EAG we uses term 'color'.

Rules, in the order as specified below, MUST be used to prune links 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 is set, the link MUST be pruned from the computation.

2. Check if any exclude SRLG rule is part of the Flex-Algorithm definition. If such exclude rule exists, check if the link is part of any SRLG that is also part of the SRLG exclude rule. If the link is part of such SRLG, the link MUST be pruned from the computation.

3. 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 no such color is set, the link MUST be pruned from the computation.

4. 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 all such colors are not set on the link, the link MUST be pruned from the computation.

5. If the Flex-Algorithm definition uses other than IGP metric (Section 5), 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.

12.1. Multi-area and Multi-domain Considerations

Any IGP Shortest Path Tree calculation is limited to a single area. This applies to Flex-Algorithm calculations as well. Given that the computing router does not have visibility of the topology of the next areas or domain, the Flex-Algorithm specific path to an inter-area or inter-domain prefix will be computed for the local area only. The egress L1/L2 router (ABR in OSPF), or ASBR for inter-domain case, will be selected based on the best path for the given Flex-Algorithm in the local area and such egress ABR or ASBR router will be responsible to compute the best Flex-Algorithm specific path over the next area or domain. This may produce an end-to-end path, which is sub-optimal based on Flex-Algorithm constraints. In cases where the ABR or ASBR has no reachability to a prefix for a given Flex-Algorithm in the next area or domain, the traffic may be dropped by the ABR/ASBR.

To allow the optimal end-to-end path for an inter-area or inter-domain prefix for any Flex-Algorithm to be computed, the FAPM has been defined in Section 8 and Section 9.

If the FAD selected based on the rules defined in Section 5.3 includes the M-flag, an ABR or ASBR MUST include the FAPM (Section 8, Section 9) when advertising the prefix between areas or domains. Such metric will be equal to the metric to reach the prefix for a given Flex-Algorithm in a source area or domain. This is similar in nature to how the metric is set when prefixes are advertised between areas or domains for the default algorithm.

If the FAD selected based on the rules defined in Section 5.3 includes the M-flag, the FAPM MUST be used during calculation of prefix reachability for the inter-area and external prefixes. If the FAPM for the Flex-Algorithm is not advertised with the inter-area or external prefix reachability advertisement, the prefix MUST be considered as unreachable for that Flex-Algorithm.

Flex-Algorithm prefix metrics MUST NOT be used during the Flex-Algorithm computation unless the FAD selected based on the rules defined in Section 5.3 includes the M-Flag, as described in (Section 6.4 or Section 7.4).

If the FAD selected based on the rules defined in Section 5.3 does not includes the M-flag, it is NOT RECOMMENDED to use the Flex-Algorithm for inter-area or inter-domain prefix reachability. The

reason is that without the explicit Flex-Algorithm Prefix Metric advertisement, it is not possible to conclude whether the ABR or ASBR has reachability to the inter-area or inter-domain prefix for a given Flex-Algorithm in the next area or domain. Sending the Flex-Algoritm traffic for such prefix towards the ABR or ASBR may result in traffic looping or black-holing.

The FAPM MUST NOT be advertised with ISIS L1 or L2 intra-area, OSPFv2 intra-area, or OSPFv3 intra-area routes. If the FAPM is advertised for these route-types, it MUST be ignored during the prefix reachability calculation.

The M-flag in FAD is not applicable to prefixes advertised as SRv6 locators. The ISIS SRv6 Locator TLV includes the Algorithm and Metric fields [I-D.ietf-lsr-isis-srv6-extensions]. When the ISIS SRv6 Locator is advertised between areas or domains, the metric field in the Locator TLV MUST be used irrespective of the M-flag in the FAD advertisement.

13. Flex-Algorithm and Forwarding Plane

This section describes how Flex-Algorithm paths are used in forwarding.

13.1. Segment Routing MPLS Forwarding for Flex-Algorithm

This section describes how Flex-Algorithm paths are used with SR MPLS forwarding.

Prefix SID advertisements include an SR-Algorithm value and, as such, are associated with the specified SR-Algorithm. Prefix-SIDs are also associated with a specific topology which is inherited from the associated prefix reachability advertisement. When the algorithm value advertised is a Flex-Algorithm value, the Prefix SID is associated with paths calculated using that Flex-Algorithm in the associated topology.

A Flex-Algorithm path MUST be installed in the MPLS forwarding plane using the MPLS label that corresponds to the Prefix-SID that was advertised for that Flex-algorithm. If the Prefix SID for a given Flex-algorithm is not known, the Flex-Algorithm specific path cannot be installed in the MPLS forwarding plane.

Traffic that is supposed to be routed via Flex-Algorithm specific paths, MUST be dropped when there are no such paths available.

Loop Free Alternate (LFA) paths for a given Flex-Algorithm MUST be computed using the same constraints as the calculation of the primary

paths for that Flex-Algorithm. LFA paths MUST only use Prefix-SIDs advertised specifically for the given algorithm. LFA paths MUST NOT use an Adjacency-SID that belongs to a link that has been pruned from the Flex-Algorithm computation.

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

13.2. SRv6 Forwarding for Flex-Algorithm

This section describes how Flex-Algorithm paths are used with SRv6 forwarding.

In SRv6 a node is provisioned with topology/algorithm specific locators for each of the topology/algorithm pairs supported by that node. Each locator is an aggregate prefix for all SIDs provisioned on that node which have the matching topology/algorithm.

The SRv6 locator advertisement in IGPs ([I-D.ietf-lsr-isis-srv6-extensions] [I-D.ietf-lsr-ospfv3-srv6-extensions]) includes the MTID value that associates the locator with a specific topology. SRv6 locator advertisements also includes an Algorithm value that explicitly associates the locator with a specific algorithm. When the algorithm value advertised with a locator represents a Flex-Algorithm, the paths to the locator prefix MUST be calculated using the specified Flex-Algorithm in the associated topology.

Forwarding entries for the locator prefixes advertised in IGPs MUST be installed in the forwarding plane of the receiving SRv6 capable routers when the associated topology/algorithm is participating in them. Forwarding entries for locators associated with Flex-Algorithms in which the node is not participating MUST NOT be installed in the forwarding palne.

When the locator is associated with a Flex-Algorithm, LFA paths to the locator prefix MUST be calculated using such Flex-Algorithm in the associated topology, to guarantee that they follow the same constraints as the calculation of the primary paths. LFA paths MUST only use SRv6 SIDs advertised specifically for the given Flex-Algorithm.

If LFA protection is being used to protect locators associated with a given Flex-Algorithm, all routers in the area participating in the given Flex-Algorithm SHOULD advertise at least one Flex-Algorithm

specific locator and END SID per node and one END.X SID for every link that has not been pruned from such Flex-Algorithm computation. These locators and SIDs are used to steer traffic over the LFA-computed backup path.

13.3. Other Applications' Forwarding for Flex-Algorithm

Any application that wants to use Flex-Algorithm specific forwarding needs to install some form of Flex-Algorithm specific forwarding entries.

Application-specific forwarding for Flex-Algorithm MUST be defined for each application and is outside of the scope of this document.

14. Operational considerations

14.1. Inter-area Considerations

The scope of the FA computation is an area, so is the scope of the FAD. In ISIS, the Router Capability TLV in which the FAD Sub-TLV is advertised MUST have the S-bit clear, which prevents it to be flooded outside of the level in which it was originated. Even though in OSPF the FAD Sub-TLV can be flooded in an RI LSA that has AS flooding scope, the FAD selection is performed for each individual area in which it is being used.

There is no requirement for the FAD for a particular Flex-Algorithm to be identical in all areas in the network. For example, traffic for the same Flex-Algorithm may be optimized for minimal delay (e.g., using delay metric) in one area or level, while being optimized for available bandwidth (e.g., using IGP metric) in another area or level.

As described in Section 5.1, ISIS allows the re-generation of the winning FAD from level 2, without any modification to it, into a level 1 area. This allows the operator to configure the FAD in one or multiple routers in the level 2, without the need to repeat the same task in each level 1 area, if the intent is to have the same FAD for the particular Flex-Algorithm across all levels. This can similarly be achieved in OSPF by using the AS flooding scope of the RI LSA in which the FAD Sub-TLV for the particular Flex-Algoritm is advertised.

Re-generation of FAD from a level 1 area to the level 2 area is not supported in ISIS, so if the intent is to regenerate the FAD between ISIS levels, the FAD MUST be defined on router(s) that are in level 2. In OSPF, the FAD definition can be done in any area and be

propagated to all routers in the OSPF routing domain by using the AS flooding scope of the RI LSA.

14.2. Usage of SRLG Exclude Rule with Flex-Algorithm

There are two different ways in which SRLG information can be used with Flex-Algorithm:

In a context of a single Flex-Algorithm, it can be used for computation of backup paths, as described in [I-D.ietf-rtgwg-segment-routing-ti-lfa]. This usage does not require association of any specific SRLG constraint with the given Flex-Algorithm definition.

In the context of multiple Flex-Algorithms, it can be used for creating disjoint sets of paths by pruning the links belonging to a specific SRLG from the topology on which a specific Flex-Algorithm computes its paths. This usage:

Facilitates the usage of already deployed SRLG configurations for setup of disjoint paths between two or more Flex-Algorithms.

Requires explicit association of a given Flex-Algorithm with a specific set of SRLG constraints as defined in Section 6.5 and Section 7.5.

The two usages mentioned above are orthogonal.

14.3. Max-metric consideration

Both ISIS and OSPF have a mechanism to set the IGP metric on a link to a value that would make the link either non-reachable or to serve as the link of last resort. Similar functionality would be needed for the Min Unidirectional Link Delay and TE metric, as these can be used to compute Flex-Algorithm paths.

The link can be made un-reachable for all Flex-Algorithms that use Min Unidirectional Link Delay as metric, as described in Section 5.1, by removing the Flex-Algorithm ASLA Min Unidirectional Link Delay advertisement for the link. The link can be made the link of last resort by setting the delay value in the Flex-Algorithm ASLA delay advertisement for the link to the value of 16,777,215 (2^24 - 1).

The link can be made un-reachable for all Flex-Algorithms that use TE metric, as described in Section 5.1, by removing the Flex-Algorithm ASLA TE metric advertisement for the link. The link can be made the link of last resort by setting the TE metric value in the Flex-

Algorithm ASLA delay advertisement for the link to the value of (2^24 - 1) in ISIS and (2^32 - 1) in OSPF.

15. Backward Compatibility

This extension brings no new backward compatibility issues.

16. Security Considerations

This draft adds two new ways to disrupt IGP networks:

An attacker can hijack a particular Flex-Algorithm by advertising a FAD with a priority of 255 (or any priority higher than that of the legitimate nodes).

An attacker could make it look like a router supports a particular Flex-Algorithm when it actually doesn't, or vice versa.

Both of these attacks can be addressed by the existing security extensions as described in [RFC5304] and [RFC5310] for ISIS, in [RFC2328] and [RFC7474] for OSPFv2, and in [RFC5340] and [RFC4552] for OSPFv3.

17. IANA Considerations

17.1. IGP IANA Considerations

17.1.1. IGP Algorithm Types Registry

This document makes the following registrations in the "IGP Algorithm Types" registry:

Type: 128-255.

Description: Flexible Algorithms.

Reference: This document (Section 4).

17.1.2. IGP Metric-Type Registry

IANA is requested to set up a registry called "IGP Metric-Type Registry" under a "Interior Gateway Protocol (IGP) Parameters" IANA registries. The registration policy for this registry is "Standards Action" ([RFC8126] and [RFC7120]).

Values in this registry come from the range 0-255.

This document registers following values in the "IGP Metric-Type Registry":

Type: 0

Description: IGP metric

Reference: This document (Section 5.1)

Type: 1

Description: Min Unidirectional Link Delay as defined in [RFC8570], section 4.2, and [RFC7471], section 4.2.

Reference: This document (Section 5.1)

Type: 2

Description: Traffic Engineering Default Metric as defined in [RFC5305], section 3.7, and Traffic engineering metric as defined in [RFC3630], section 2.5.5

Reference: This document (Section 5.1)

17.2. Flexible Algorithm Definition Flags Registry

IANA is requested to set up a registry called "ISIS Flexible Algorithm Definition Flags Registry" under a "Interior Gateway Protocol (IGP) Parameters" IANA registries. The registration policy for this registry is "Standards Action" ([RFC8126] and [RFC7120]).

This document defines the following single bit in Flexible Algorithm Definition Flags registry:

Bit #	Name
0	Prefix Metric Flag (M-flag)

Reference: This document (Section 6.4, Section 7.4).

17.3. ISIS IANA Considerations

17.3.1. Sub TLVs for Type 242

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

Type: 26.

Description: Flexible Algorithm Definition.

Reference: This document (Section 5.1).

17.3.2. Sub TLVs for for TLVs 135, 235, 236, and 237

This document makes the following registrations in the "Sub-TLVs for for TLVs 135, 235, 236, and 237" registry.

Type: 6

Description: Flexible Algorithm Prefix Metric.

Reference: This document (Section 8).

17.3.3. Sub-Sub-TLVs for Flexible Algorithm Definition Sub-TLV

This document creates the following Sub-Sub-TLV Registry:

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

Registration Procedure: Expert review

Reference: This document (Section 5.1)

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

Type: 1

Description: Flexible Algorithm Exclude Admin Group

Reference: This document (Section 6.1).

Type: 2

Description: Flexible Algorithm Include-Any Admin Group

Reference: This document (Section 6.2).

Type: 3

Description: Flexible Algorithm Include-All Admin Group

Reference: This document (Section 6.3).

Type: 4

Description: Flexible Algorithm Definition Flags

Reference: This document (Section 6.4).

Type: 5

Description: Flexible Algorithm Exclude SRLG

Reference: This document (Section 6.5).

17.4. OSPF IANA Considerations

17.4.1. OSPF Router Information (RI) TLVs Registry

This specification updates the OSPF Router Information (RI) TLVs Registry.

Type: 16

Description: Flexible Algorithm Definition TLV.

Reference: This document (Section 5.2).

17.4.2. OSPFv2 Extended Prefix TLV Sub-TLVs

This document makes the following registrations in the "OSPFv2 Extended Prefix TLV Sub-TLVs" registry.

Type: 3

Description: Flexible Algorithm Prefix Metric.

Reference: This document (Section 9).

17.4.3. OSPFv3 Extended-LSA Sub-TLVs

This document makes the following registrations in the "OSPFv3 Extended-LSA Sub-TLVs" registry.

Type: 26

Description: Flexible Algorithm Prefix Metric.

Reference: This document (Section 9).

17.4.4. OSPF Flexible Algorithm Definition TLV Sub-TLV Registry

This document creates the following registry:

Registry: OSPF Flexible Algorithm Definition TLV sub-TLV

Registration Procedure: Expert review

Reference: This document (Section 5.2)

The "OSPF Flexible Algorithm Definition TLV sub-TLV" registry will define sub-TLVs at any level of nesting for the Flexible Algorithm TLV and should be added to the "Open Shortest Path First (OSPF) Parameters" registries group. New values can be allocated via IETF Review or IESG Approval.

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

Type: 1

Description: Flexible Algorithm Exclude Admin Group

Reference: This document (Section 7.1).

Type: 2

Description: Flexible Algorithm Include-Any Admin Group

Reference: This document (Section 7.2).

Type: 3

Description: Flexible Algorithm Include-All Admin Group

Reference: This document (Section 7.3).

Type: 4

Description: Flexible Algorithm Definition Flags

Reference: This document (Section 7.4).

Type: 5

Description: Flexible Algorithm Exclude SRLG

Reference: This document (Section 7.5).

Types in the range 32768-33023 are for experimental use; these will not be registered with IANA, and MUST NOT be mentioned by RFCs.

Types in the range 33024-65535 are not to be assigned at this time. Before any assignments can be made in the 33024-65535 range, there MUST be an IETF specification that specifies IANA Considerations that covers the range being assigned.

17.4.5. Link Attribute Applications Registry

This document registers following bit in the Link Attribute Applications Registry:

Bit-3

Description: Flexible Algorithm (X-bit)

Reference: This document (Section 11).

18. Acknowledgements

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 Eric Rosen, Tony Przygienda for their detailed review and excellent comments.

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

Thanks to Kenji Kumaki for his comments.

Thanks to William Britto A J. for his suggestions.

Thanks to Acee Lindem for editorial comments.

19. References

19.1. Normative References

[BCP14] "Key words for use in RFCs to Indicate Requirement Levels", <https://tools.ietf.org/html/bcp14>.

[I-D.ietf-isis-te-app]
Ginsberg, L., Psenak, P., Previdi, S., Henderickx, W., and J. Drake, "IS-IS Application-Specific Link Attributes", draft-ietf-isis-te-app-19 (work in progress), June 2020.

[I-D.ietf-lsr-isis-srv6-extensions]
Psenak, P., Filsfils, C., Bashandy, A., Decraene, B., and Z. Hu, "IS-IS Extension to Support Segment Routing over IPv6 Dataplane", draft-ietf-lsr-isis-srv6-extensions-10 (work in progress), September 2020.

[I-D.ietf-lsr-ospfv3-srv6-extensions]
Li, Z., Hu, Z., Cheng, D., Talaulikar, K., and P. Psenak, "OSPFv3 Extensions for SRv6", draft-ietf-lsr-ospfv3-srv6-extensions-01 (work in progress), August 2020.

[I-D.ietf-ospf-te-link-attr-reuse]
Psenak, P., Ginsberg, L., Henderickx, W., Tantsura, J., and J. Drake, "OSPF Application-Specific Link Attributes", draft-ietf-ospf-te-link-attr-reuse-16 (work in progress), June 2020.

[ISO10589]
International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Second Edition, Nov 2002.

[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>.

[RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <https://www.rfc-editor.org/info/rfc4203>.

[RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <https://www.rfc-editor.org/info/rfc5307>.

[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>.

[RFC7684] Psenak, P., Gredler, H., Shakir, R., Henderickx, W., Tantsura, J., and A. Lindem, "OSPFv2 Prefix/Link Attribute Advertisement", RFC 7684, DOI 10.17487/RFC7684, November 2015, <https://www.rfc-editor.org/info/rfc7684>.

[RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 7770, DOI 10.17487/RFC7770, February 2016, <https://www.rfc-editor.org/info/rfc7770>.

[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>.

[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>.

[RFC8362] Lindem, A., Roy, A., Goethals, D., Reddy Vallem, V., and F. Baker, "OSPFv3 Link State Advertisement (LSA) Extensibility", RFC 8362, DOI 10.17487/RFC8362, April 2018, <https://www.rfc-editor.org/info/rfc8362>.

[RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <https://www.rfc-editor.org/info/rfc8665>.

[RFC8666] Psenak, P., Ed. and S. Previdi, Ed., "OSPFv3 Extensions for Segment Routing", RFC 8666, DOI 10.17487/RFC8666, December 2019, <https://www.rfc-editor.org/info/rfc8666>.

[RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <https://www.rfc-editor.org/info/rfc8667>.

19.2. 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.

[I-D.ietf-rtgwg-segment-routing-ti-lfa]
Litkowski, S., Bashandy, A., Filsfils, C., Decraene, B., Francois, P., Voyer, D., Clad, F., and P. Camarillo, "Topology Independent Fast Reroute using Segment Routing", draft-ietf-rtgwg-segment-routing-ti-lfa-04 (work in progress), August 2020.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <https://www.rfc-editor.org/info/rfc3630>.

[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>.

[RFC4552] Gupta, M. and N. Melam, "Authentication/Confidentiality for OSPFv3", RFC 4552, DOI 10.17487/RFC4552, June 2006, <https://www.rfc-editor.org/info/rfc4552>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[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>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC7120] Cotton, M., "Early IANA Allocation of Standards Track Code Points", BCP 100, RFC 7120, DOI 10.17487/RFC7120, January 2014, <https://www.rfc-editor.org/info/rfc7120>.

[RFC7471] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <https://www.rfc-editor.org/info/rfc7471>.

[RFC7474] Bhatia, M., Hartman, S., Zhang, D., and A. Lindem, Ed., "Security Extension for OSPFv2 When Using Manual Key Management", RFC 7474, DOI 10.17487/RFC7474, April 2015, <https://www.rfc-editor.org/info/rfc7474>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8570] Ginsberg, L., Ed., Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 8570, DOI 10.17487/RFC8570, March 2019, <https://www.rfc-editor.org/info/rfc8570>.

Authors' Addresses

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

Email: ppsenak@cisco.com

Shraddha Hegde
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

Ketan Talaulikar
Cisco Systems, Inc.
S.No. 154/6, Phase I, Hinjawadi
PUNE, MAHARASHTRA 411 057
India

Email: ketant@cisco.com

Arkadiy Gulko
Individual

Email: arkadiy.gulko@gmail.com