
Stream: Internet Engineering Task Force (IETF)

RFC: [9857](#)

Category: Standards Track

Published: October 2025

ISSN: 2070-1721

Authors:

S. Previdi

Individual

K. Talaulikar, Ed.

Cisco Systems

J. Dong

Huawei Technologies

H. Gredler

RtBrick Inc.

J. Tantsura

Nvidia

RFC 9857

Advertisement of Segment Routing Policies Using BGP - Link State

Abstract

This document describes a mechanism used to collect Segment Routing (SR) Policy information that is locally available in a node and advertise it into BGP - Link State (BGP-LS) updates. Such information can be used by external components for path computation, reoptimization, service placement, network visualization, etc.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc9857>.

Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	5
2. Carrying SR Policy Information in BGP	5
3. SR Policy Candidate Path NLRI Type	6
3.1. SR Policy Headend as the BGP-LS Producer	7
3.2. PCE as the BGP-LS Producer	7
4. SR Policy Candidate Path Descriptor	8
5. SR Policy State TLVs	9
5.1. SR Binding SID TLV	9
5.2. SRv6 Binding SID TLV	11
5.3. SR Candidate Path State TLV	13
5.4. SR Policy Name TLV	15
5.5. SR Candidate Path Name TLV	15
5.6. SR Candidate Path Constraints TLV	16
5.6.1. SR Affinity Constraint Sub-TLV	18
5.6.2. SR SRLG Constraint Sub-TLV	20
5.6.3. SR Bandwidth Constraint Sub-TLV	20
5.6.4. SR Disjoint Group Constraint Sub-TLV	21
5.6.5. SR Bidirectional Group Constraint Sub-TLV	23
5.6.6. SR Metric Constraint Sub-TLV	25
5.7. SR Segment List TLV	27
5.7.1. SR Segment Sub-TLV	29
5.7.2. SR Segment List Metric Sub-TLV	39
5.7.3. SR Segment List Bandwidth Sub-TLV	41
5.7.4. SR Segment List Identifier Sub-TLV	42
6. Procedures	42
7. Manageability Considerations	43

8. IANA Considerations	43
8.1. BGP-LS NLRI Types	43
8.2. BGP-LS Protocol-IDs	43
8.3. BGP-LS TLVs	44
8.4. SR Policy Protocol-Origin	45
8.5. BGP-LS SR Segment Descriptors	45
8.6. BGP-LS SR Policy Metric Type	46
9. Security Considerations	47
10. References	48
10.1. Normative References	48
10.2. Informative References	49
Acknowledgements	51
Contributors	51
Authors' Addresses	51

1. Introduction

SR Policy architecture details are specified in [RFC9256]. An SR Policy comprises one or more candidate paths of which at a given time one and only one may be active (i.e., installed in forwarding and usable for the steering of traffic). Each candidate path in turn may have one or more SID lists of which one or more SID lists may be active. When multiple SID lists are active, traffic is load balanced over them. This document covers the advertisement of state information at the individual SR Policy candidate path level.

SR Policies are generally instantiated at the headend and are based on either local configuration or controller-based programming of the node using various APIs and protocols (e.g., the Path Computation Element Communication Protocol (PCEP) or BGP).

In many network environments, the configuration and state of each SR Policy that is available in the network is required by controllers. Such controllers, which are aware of both topology and SR Policy state information, allow the network operator to optimize several functions and operations in their networks.

One example of a controller is the stateful Path Computation Element (PCE) [RFC8231], which can provide benefits in path optimization. While some extensions are proposed in the PCEP for Path Computation Clients (PCCs) to report Label Switched Path (LSP) states to the PCE, this mechanism may not be applicable in a management-based PCE architecture as specified in

Section 5.5 of [RFC4655]. As illustrated in the figure below, the PCC is not a Label Switching Router (LSR) in the routing domain, thus the headend nodes of the SR Policies may not implement the PCEP. In this case, a general mechanism to collect the SR Policy states from the ingress Label Edge Routers (LERs) is needed. This document proposes an SR Policy state collection mechanism complementary to the mechanism defined in [RFC8231].

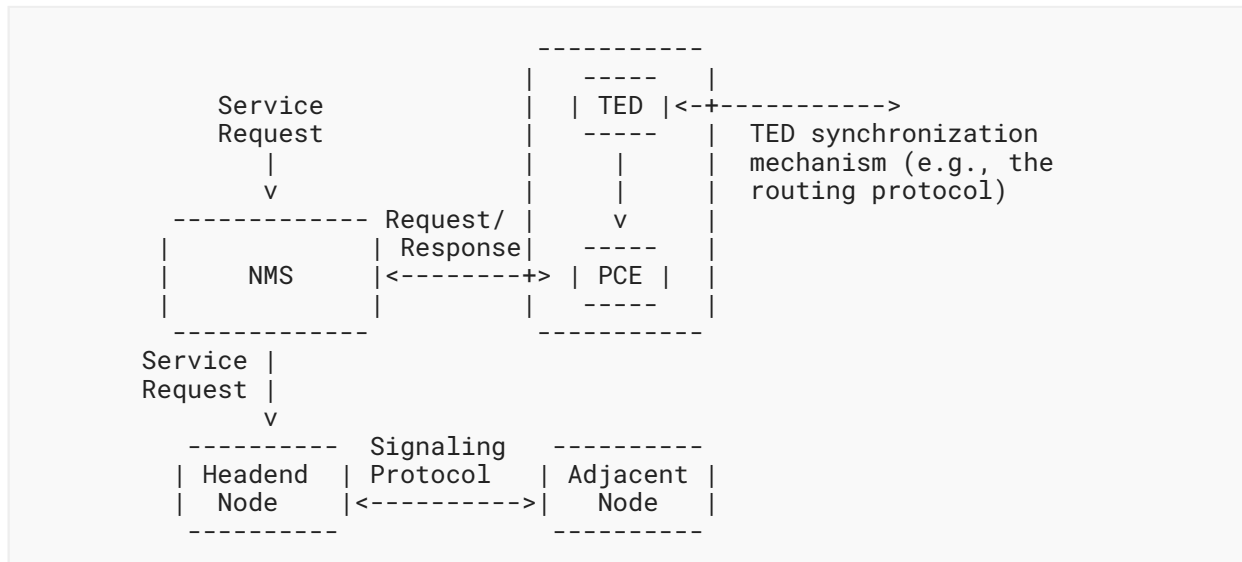


Figure 1: Management-Based PCE Usage

In networks with composite PCE nodes as specified in Section 5.1 of [RFC4655], PCE is implemented on several routers in the network, and the PCCs in the network can use the mechanism described in [RFC8231] to report the SR Policy information to the PCE nodes. An external component may also need to collect the SR Policy information from all the PCEs in the network to obtain a global view of the state of all SR Policy paths in the network.

In multi-area or multi-AS scenarios, each area or AS can have a child PCE to collect the SR Policies in its domain. In addition, a parent PCE needs to collect SR Policy information from multiple child PCEs to obtain a global view of SR Policy paths inside and across the domains involved.

In another network scenario, a centralized controller is used for service placement. Obtaining the SR Policy state information is quite important for making appropriate service placement decisions with the purpose of both meeting the application's requirements and utilizing network resources efficiently.

The Network Management System (NMS) may need to provide global visibility of the SR Policies in the network as part of the network visualization function.

BGP has been extended to distribute link-state and Traffic Engineering (TE) information to external components [RFC9552]. Using the same protocol to collect SR Policy and state information is desirable for these external components since this avoids introducing multiple

protocols for network topology information collection. This document describes a mechanism to distribute SR Policy information (both SR-MPLS and SRv6 [RFC8402]) to external components using BGP-LS and covers both explicit and dynamic candidate paths. The advertisements of a composite candidate path are outside the scope of this document.

The BGP-LS Producer [RFC9552] that is originating the advertisement of SR Policy information can be either:

- an SR Policy headend node or
- a PCE that is receiving the SR Policy information from its PCCs (i.e., SR Policy headend nodes) via PCEP

The extensions specified in this document complement the BGP SR Policy SAFI [RFC9830] [RFC9831] and are used to advertise SR Policies from controllers to the headend routers using BGP by enabling the reporting of the operational state of those SR Policies back from the headend to the controllers.

While this document focuses on SR Policies, [BGP-LS-TE-PATH] introduces further extensions to support other TE paths such as MPLS-TE LSPs.

The encodings specified in this document (specifically in Sections 4 and 5) make use of flags that convey various types of information of the SR Policy. The document uses the term "set" to indicate that the value of a flag bit is 1 and the term "clear" when the value is 0.

1.1. Requirements Language

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

2. Carrying SR Policy Information in BGP

The "Link-State Network Layer Reachability Information (NLRI)" defined in [RFC9552] is extended to carry the SR Policy information. New TLVs carried in the BGP-LS Attribute defined in [RFC9552] are also defined to carry the attributes of an SR Policy in the subsequent sections.

The format of the Link-State NLRI is defined in [RFC9552] as follows:

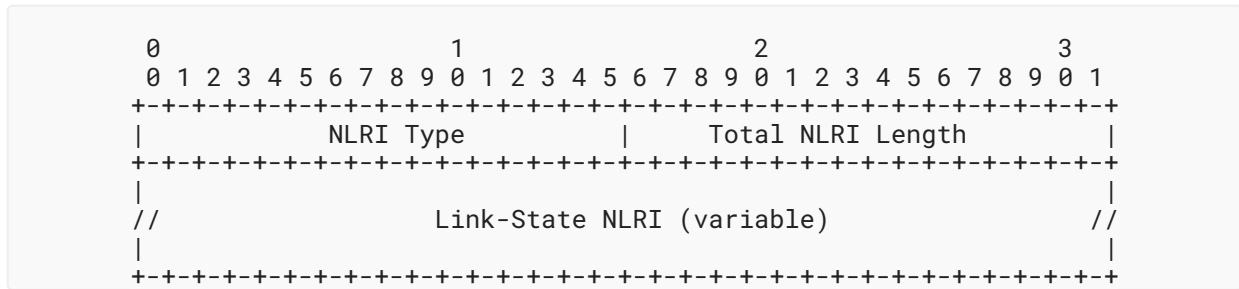


Figure 2: BGP-LS NLRI Format

An additional NLRI Type known as "SR Policy Candidate Path NLRI" (value 5) is defined for the advertisement of SR Policy Information.

This SR Policy Candidate Path NLRI is used to report the state details of individual SR Policy candidate paths along with their underlying segment lists.

3. SR Policy Candidate Path NLRI Type

This document defines the SR Policy Candidate Path NLRI type with its format as shown in the following figure:

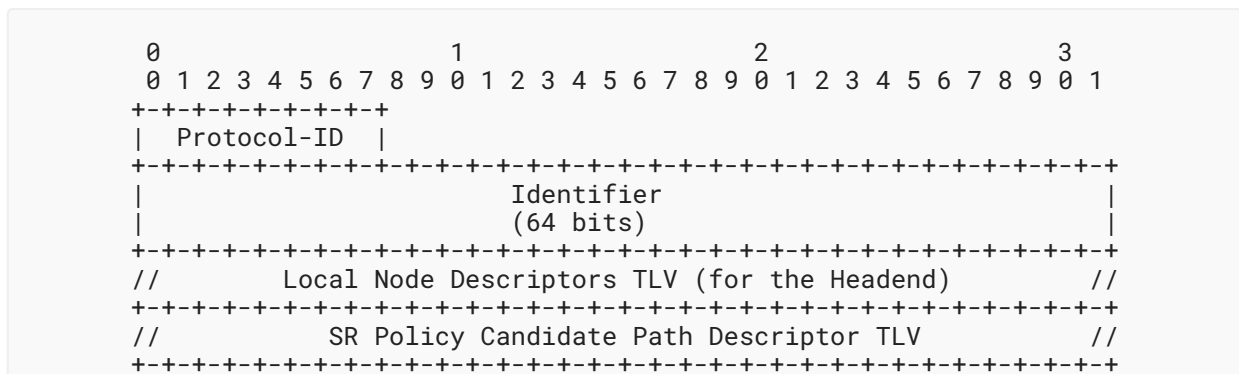


Figure 3: SR Policy Candidate Path NLRI Format

Where:

Protocol-ID field: Specifies the component that owns the SR Policy state in the advertising node. An additional Protocol-ID "Segment Routing" (value 9) is introduced by this document that **MUST** be used for the advertisement of SR Policies.

Identifier: An 8-octet value as defined in [Section 5.2](#) of [\[RFC9552\]](#).

Local Node Descriptors (TLV 256): Defined in [\[RFC9552\]](#) and used as specified further in this section.

SR Policy Candidate Path Descriptor TLV: Specified in [Section 4](#).

The Local Node Descriptors TLV carries information that only identifies the headend node of the SR Policy irrespective of whether the BGP-LS Producer is a headend or a PCE node.

The Local Node Descriptors TLV **MUST** include at least one of the following Node Descriptor TLVs:

- IPv4 Router-ID of Local Node (TLV 1028) [RFC9552], which identifies the headend node of the SR Policy as specified in Section 2.1 of [RFC9256].
- IPv6 Router-ID of Local Node (TLV 1029) [RFC9552], which identifies the headend node of the SR Policy as specified in Section 2.1 of [RFC9256].

The following subsections describe the encoding of sub-TLVs within the Local Node Descriptors TLV depending on which node is the BGP-LS Producer.

3.1. SR Policy Headend as the BGP-LS Producer

The Local Node Descriptors TLV **MUST** include the following Node Descriptor TLVs when the headend node is the BGP-LS Producer:

- BGP Router-ID (TLV 516) [RFC9086], which contains a valid BGP Identifier of the headend node of the SR Policy.
- Autonomous System (TLV 512) [RFC9552], which contains the Autonomous System Number (ASN) (or AS Confederation Identifier [RFC5065], if confederations are used) of the headend node of the SR Policy.

The Local Node Descriptors TLV **MAY** include the following Node Descriptor TLVs when the headend node is the BGP-LS Producer:

- BGP Confederation Member (TLV 517) [RFC9086], which contains the ASN of the confederation member (i.e., Member-AS Number); if BGP confederations are used, it contains the headend node of the SR Policy.
- Other Node Descriptors as defined in [RFC9552] to identify the headend node of the SR Policy. The determination of whether the IGP Router-ID (TLV 515) contains a 4-octet OSPF Router-ID or a 6-octet ISO System-ID is to be done based on the length of that sub-TLV as the Protocol-ID in the NLRI is always going to be "Segment Routing".

3.2. PCE as the BGP-LS Producer

The PCE node **MUST NOT** include its identifiers in the Node Descriptor TLV in the NLRI as the Node Descriptor TLV **MUST** only carry the identifiers of the SR Policy headend.

The Local Node Descriptors TLV **MAY** include the following Node Descriptor TLVs when the PCE node is the BGP-LS Producer and it has this information about the headend (e.g., as part of its topology database):

- BGP Router-ID (TLV 516) [RFC9086], which contains a valid BGP Identifier of the headend node of the SR Policy.
- Autonomous System (TLV 512) [RFC9552], which contains the ASN (or AS Confederation Identifier [RFC5065], if confederations are used) of the headend node of the SR Policy.

- BGP Confederation Member (TLV 517) [RFC9086], which contains the ASN of the confederation member (i.e., Member-AS Number); if BGP confederations are used, it contains the headend node of the SR Policy.
- Other Node Descriptors as defined in [RFC9552] to identify the headend node of the SR Policy. The determination of whether the IGP Router-ID (TLV 515) contains a 4-octet OSPF Router-ID or a 6-octet ISO System-ID is to be done based on the length of that sub-TLV since the Protocol-ID in the NLRI is always going to be "Segment Routing".

When a PCE node is functioning as the BGP-LS Producer on behalf of one or more headends, it **MAY** include its own BGP Router-ID (TLV 516), Autonomous System (TLV 512), or BGP Confederation Member (TLV 517) in the BGP-LS Attribute.

4. SR Policy Candidate Path Descriptor

The SR Policy Candidate Path Descriptor TLV identifies an SR Policy candidate path as defined in [RFC9256]. It is a mandatory TLV for the SR Policy Candidate Path NLRI type. The TLV has the following format:

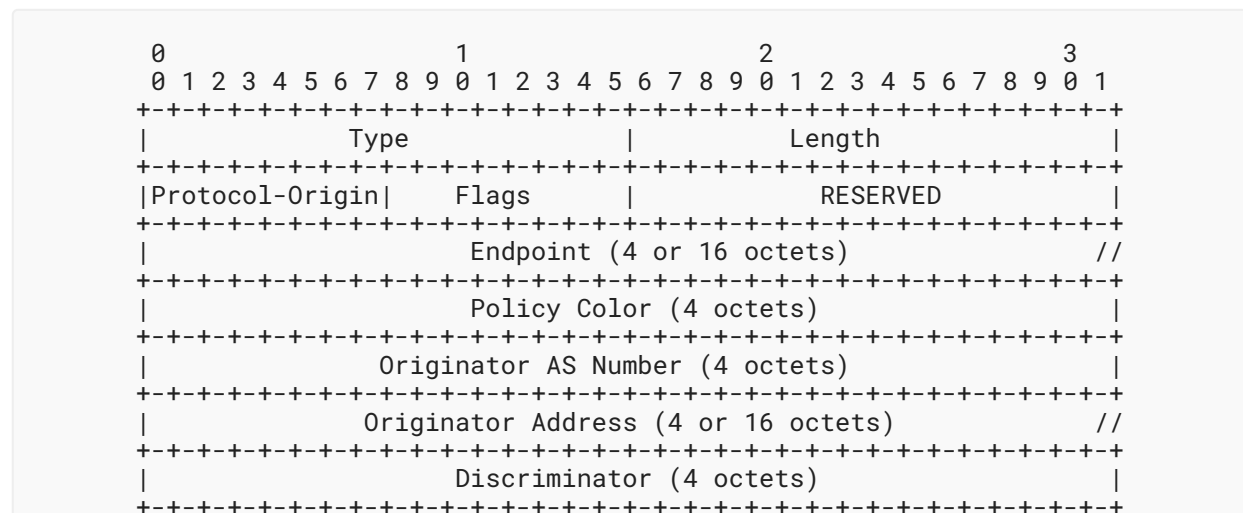


Figure 4: SR Policy Candidate Path Descriptor Format

Where:

Type: 554

Length: Variable (valid values are 24, 36, or 48 octets)

Protocol-Origin: 1-octet field that identifies the protocol or component that is responsible for the instantiation of this path as specified in Section 2.3 of [RFC9256]. The protocol-origin code points to be used are listed in Section 8.4.

Flags: 1-octet field with the following bit positions defined. Other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.


```

  0 1 2 3 4 5 6 7
  +--+--+--+--+--+--+--+
  |E|O|          |
  +--+--+--+--+--+--+--+

```

Where:

E-Flag: Indicates the encoding of an endpoint as an IPv6 address when set and an IPv4 address when clear.

O-Flag: Indicates the encoding of the originator address as an IPv6 address when set and an IPv4 address when clear.

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

Endpoint: 4 or 16 octets (as indicated by the flags) containing the address of the endpoint of the SR Policy as specified in [Section 2.1](#) of [\[RFC9256\]](#).

Policy Color: 4 octets that indicate the color of the SR Policy as specified in [Section 2.1](#) of [\[RFC9256\]](#).

Originator ASN: 4 octets to carry the 4-byte encoding of the ASN of the originator. Refer to [Section 2.4](#) of [\[RFC9256\]](#) for details.

Originator Address: 4 or 16 octets (as indicated by the flags) to carry the address of the originator. Refer to [Section 2.4](#) of [\[RFC9256\]](#) for details.

Discriminator: 4 octets to carry the discriminator of the path. Refer to [Section 2.5](#) of [\[RFC9256\]](#) for details.

5. SR Policy State TLVs

This section defines the various TLVs that enable the headend to report the state at the SR Policy candidate path level. These TLVs (and their sub-TLVs) are carried in the optional non-transitive BGP-LS Attribute defined in [\[RFC9552\]](#) and are associated with the SR Policy Candidate Path NLRI type.

The detailed procedures for the advertisement are described in [Section 6](#).

5.1. SR Binding SID TLV

The SR Binding Segment Identifier (BSID) is an optional TLV that is used to report the BSID and its attributes for the SR Policy candidate path. The TLV **MAY** also optionally contain the Specified BSID value for reporting as described in [Section 6.2.3](#) of [\[RFC9256\]](#). Only a single instance of this TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per [Section 8.2.2](#) of [\[RFC9552\]](#)) is used and the rest are ignored.

The TLV has the following format:

RESERVED: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Binding SID: Indicates the operational or allocated BSID value based on the status flags.

Specified BSID: Used to report the explicitly specified BSID value regardless of whether it is successfully allocated or not. The field is set to value 0 when the BSID has not been specified.

The BSID fields above depend on the data plane (SRv6 or MPLS) indicated by the D-flag. If the D-flag is set (SRv6 data plane), then the length of the BSID fields is 16 octets. If the D-flag is clear (MPLS data plane), then the length of the BSID fields is 4 octets. When carrying the MPLS Label, as shown in the figure below, the TC, S, and TTL (total of 12 bits) are RESERVED and **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

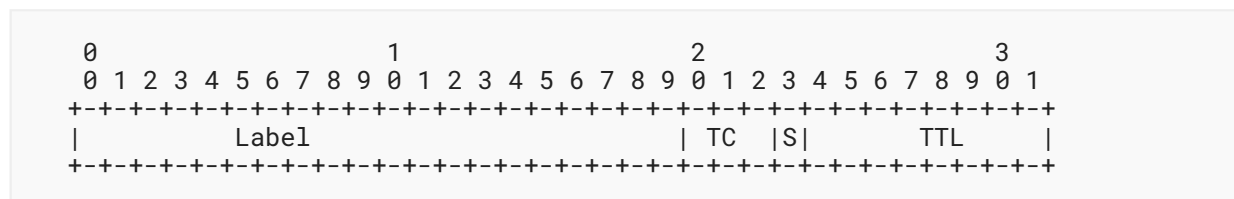


Figure 6: SR Binding SID Label Format

In the case of an SRv6, the SR Binding SID sub-TLV does not have the ability to signal the SRv6 Endpoint behavior [RFC8986] or the structure of the SID. Therefore, the SR Binding SID sub-TLV **SHOULD NOT** be used for the advertisement of an SRv6 Binding SID. Instead, the SRv6 Binding SID TLV defined in Section 5.2 **SHOULD** be used for the signaling of an SRv6 Binding SID. The use of the SR Binding SID sub-TLV for advertisement of the SRv6 Binding SID has been deprecated, and it is documented here only for backward compatibility with implementations that followed early draft versions of this specification.

5.2. SRv6 Binding SID TLV

The SRv6 Binding SID (BSID) is an optional TLV that is used to report the SRv6 BSID and its attributes for the SR Policy candidate path. The TLV **MAY** also optionally contain the Specified SRv6 BSID value for reporting as described in Section 6.2.3 of [RFC9256]. Multiple instances of this TLV may be used to report each of the SRv6 BSIDs associated with the candidate path.

The TLV has the following format:

Specified BSID: Used to report the explicitly specified BSID value regardless of whether it is successfully allocated or not. The field is set to value 0 when the BSID has not been specified.

Sub-TLVs: Variable and contain any other optional attributes associated with the SRv6 BSID.

The SRv6 Endpoint Behavior TLV (1250) and the SRv6 SID Structure TLV (1252) **MAY** optionally be used as sub-TLVs of the SRv6 Binding SID TLV to indicate the SRv6 Endpoint behavior and SID structure for the Binding SID value in the TLV. [RFC9514] defines the SRv6 Endpoint Behavior TLV and the SRv6 SID Structure TLV.

5.3. SR Candidate Path State TLV

The SR Candidate Path State TLV provides the operational status and attributes of the SR Policy at the candidate path level. Only a single instance of this TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The TLV has the following format:

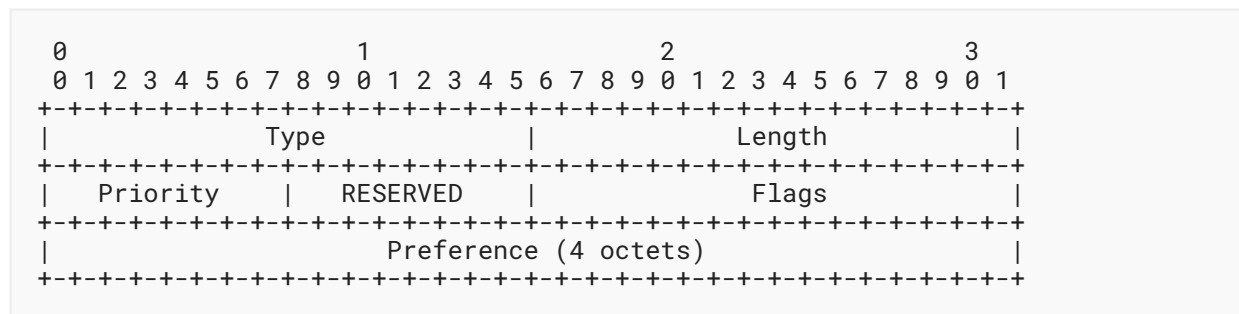


Figure 8: SR Candidate Path State TLV Format

Where:

Type: 1202

Length: 8 octets

Priority: 1-octet value that indicates the priority of the candidate path. Refer to Section 2.12 of [RFC9256].

RESERVED: 1 octet. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Flags: 2-octet field that indicates the attribute and status of the candidate path. The following bit positions are defined, and the semantics are described in Section 5 of [RFC9256] unless stated otherwise for individual flags. Other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.

```

      0                               1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+---+---+
|S|A|B|E|V|O|D|C|I|T|U|           |
+---+---+---+---+---+---+---+---+---+

```

Where:

S-Flag: Indicates that the candidate path is in an administrative shut state when set and not in an administrative shut state when clear.

A-Flag: Indicates that the candidate path is the active path (i.e., one provisioned in the forwarding plane as specified in [Section 2.9](#) of [RFC9256]) for the SR Policy when set and not the active path when clear.

B-Flag: Indicates that the candidate path is the backup path (i.e., one identified for path protection of the active path as specified in [Section 9.3](#) of [RFC9256]) for the SR Policy when set and not the backup path when clear.

E-Flag: Indicates that the candidate path has been evaluated for validity (e.g., headend may evaluate candidate paths based on their preferences) when set and has not been evaluated for validity when clear.

V-Flag: Indicates that the candidate path has at least one valid SID list when set and that no valid SID list is available or evaluated when clear. When the E-flag is clear (i.e., the candidate path has not been evaluated), then this flag **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

O-Flag: Indicates that the candidate path was instantiated by the headend due to an on-demand next hop trigger based on a local template when set and that the candidate path has not been instantiated due to an on-demand next hop trigger when clear. Refer to [Section 8.5](#) of [RFC9256] for details.

D-Flag: Indicates that the candidate path was delegated for computation to a PCE/controller when set and that the candidate path has not been delegated for computation when clear.

C-Flag: Indicates that the candidate path was provisioned by a PCE/controller when set and that the candidate path was not provisioned by a PCE/controller when clear.

I-Flag: Indicates that the candidate path is to perform the "Drop-Upon-Invalid" behavior when no other valid candidate path is available for this SR Policy when the flag is set. Refer to [Section 8.2](#) of [RFC9256] for details. When clear, it indicates that the candidate path is not enabled for the "Drop-Upon-Invalid" behavior.

T-Flag: Indicates that the candidate path has been marked as eligible for use as a transit policy on the headend when set and not eligible for use as a transit policy when clear. Transit policy is a policy whose BSID can be used in the segment list of another SR Policy. Refer to [Section 8.3](#) of [RFC9256] for steering into a transit policy using its BSID.

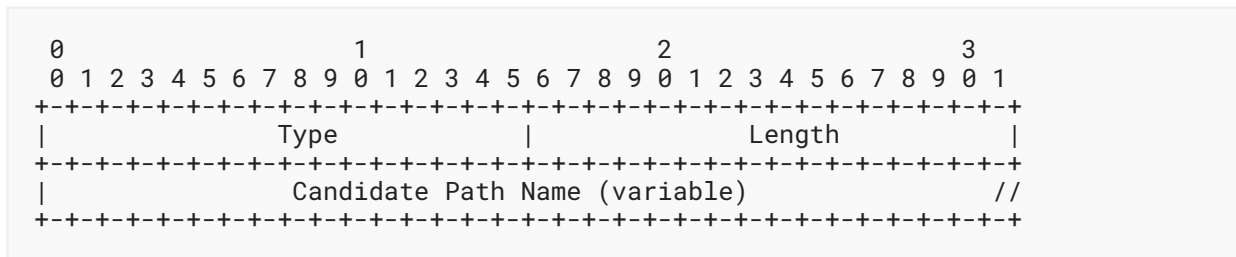


Figure 10: SR Candidate Path Name TLV Format

Where:

Type: 1203

Length: Variable

Candidate Path Name: Symbolic name for the SR Policy candidate path without a NUL terminator as specified in Section 2.6 of [RFC9256]. It is **RECOMMENDED** that the size of the symbolic name be limited to 255 bytes. Implementations **MAY** choose to truncate long names to 255 bytes when signaling via BGP-LS.

5.6. SR Candidate Path Constraints TLV

The SR Candidate Path Constraints TLV is an optional TLV that is used to report the constraints associated with the candidate path. The constraints are generally applied to a dynamic candidate path that is either computed by the headend or delegated to a controller. The constraints may also be applied to an explicit path where the computation entity is expected to validate that the path satisfies the specified constraints; if not, the path is to be invalidated (e.g., due to topology changes). Only a single instance of this TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The TLV has the following format:

T-Flag: Indicates that the candidate path uses only the SIDs belonging to the specified topology when set and that the candidate path does not use only the SIDs belonging to the specified topology when clear.

S-Flag: Indicates that the use of protected (P-flag) or unprotected (U-flag) SIDs becomes a strict constraint instead of a preference when set and that there is no strict constraint (and only a preference) when clear.

F-Flag: Indicates that the candidate path is fixed once computed and not modified except on operator intervention and that the candidate path may be modified as part of recomputation when clear.

H-Flag: Indicates that the candidate path uses only adjacency SIDs and traverses hop-by-hop over the links corresponding to those adjacency SIDs when set and that the candidate path is not restricted to using only hop-by-hop adjacency SIDs when clear.

RESERVED1: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

MTID: Indicates the multi-topology identifier of the IGP topology that is preferred to be used when the path is set up. When the T-flag is set, then the path is strictly using the specified topology SIDs only.

Algorithm: Indicates the algorithm that is preferred to be used when the path is set up. When the A-flag is set, then the path is strictly using the specified algorithm SIDs only. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

RESERVED2: 1 octet. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

sub-TLVs: One or more optional sub-TLVs **MAY** be included in this TLV to describe other constraints. These sub-TLVs are: SR Affinity Constraint, SR Shared Risk Link Group (SRLG) Constraint, SR Bandwidth Constraint, SR Disjoint Group Constraint, SR Bidirectional Group Constraint, and SR Metric Constraint.

These constraint sub-TLVs are defined below.

5.6.1. SR Affinity Constraint Sub-TLV

The SR Affinity Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to carry the affinity constraints [RFC2702] associated with the candidate path. The affinity is expressed in terms of an Extended Administrative Group (EAG) as defined in [RFC7308]. Only a single instance of this sub-TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The sub-TLV has the following format:

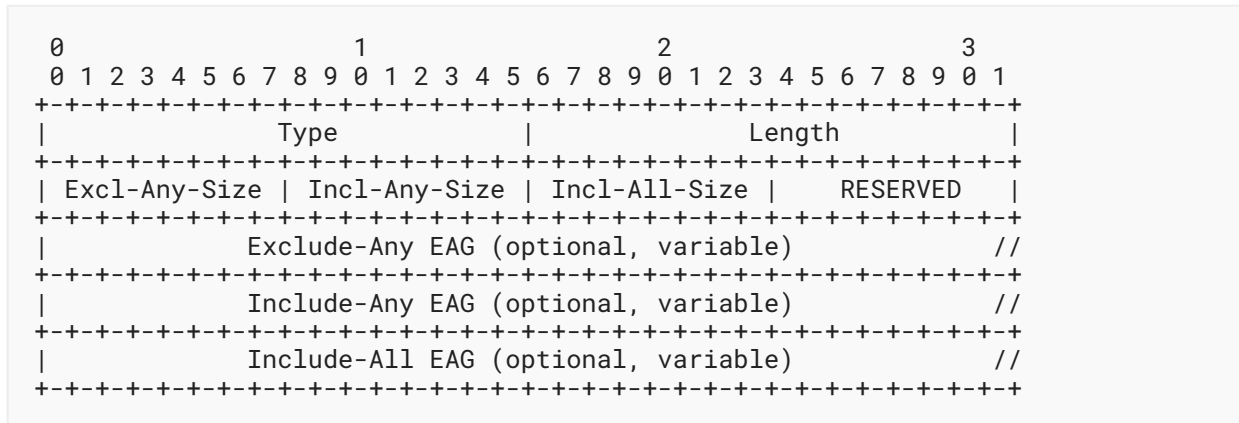


Figure 12: SR Affinity Constraint Sub-TLV Format

Where:

Type: 1208

Length: Variable, dependent on the size of the EAG. **MUST** be a non-zero multiple of 4 octets.

Exclude-Any-Size: 1 octet to indicate the size of Exclude-Any EAG bitmask size in multiples of 4 octets (e.g., value 0 indicates the Exclude-Any EAG field is skipped, and value 1 indicates that 4 octets of Exclude-Any EAG are included).

Include-Any-Size: 1 octet to indicate the size of Include-Any EAG bitmask size in multiples of 4 octets (e.g., value 0 indicates the Include-Any EAG field is skipped, and value 1 indicates that 4 octets of Include-Any EAG are included).

Include-All-Size: 1 octet to indicate the size of Include-All EAG bitmask size in multiples of 4 octets (e.g., value 0 indicates the Include-All EAG field is skipped, and value 1 indicates that 4 octets of Include-All EAG are included).

RESERVED: 1 octet. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Exclude-Any EAG: The bitmask used to represent the affinities that have been excluded from the path.

Include-Any EAG: The bitmask used to represent the affinities that have been included in the path.

Include-All EAG: The bitmask used to represent all the affinities that have been included in the path.

5.6.2. SR SRLG Constraint Sub-TLV

The SR SRLG Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to carry the SRLG values [RFC4202] that have been excluded from the candidate path. Only a single instance of this sub-TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The sub-TLV has the following format:

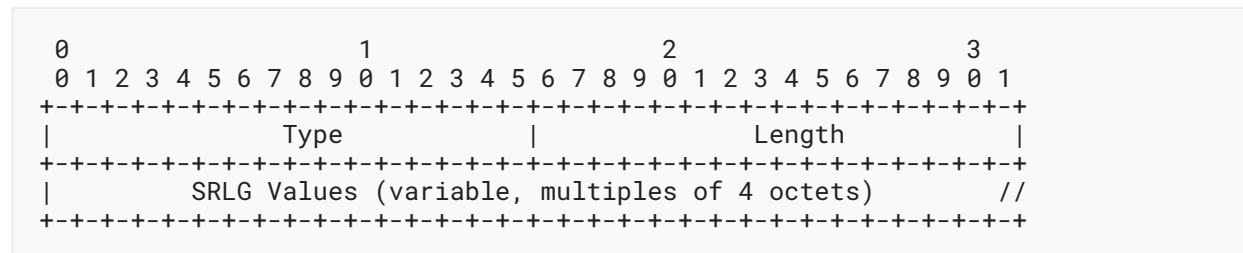


Figure 13: SR SRLG Constraint Sub-TLV Format

Where:

Type: 1209

Length: Variable, dependent on the number of SRLGs encoded. **MUST** be a non-zero multiple of 4 octets.

SRLG Values: One or more SRLG values. Each SRLG value is of 4 octets.

5.6.3. SR Bandwidth Constraint Sub-TLV

The SR Bandwidth Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to indicate the bandwidth that has been requested for the candidate path. Only a single instance of this sub-TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The sub-TLV has the following format:

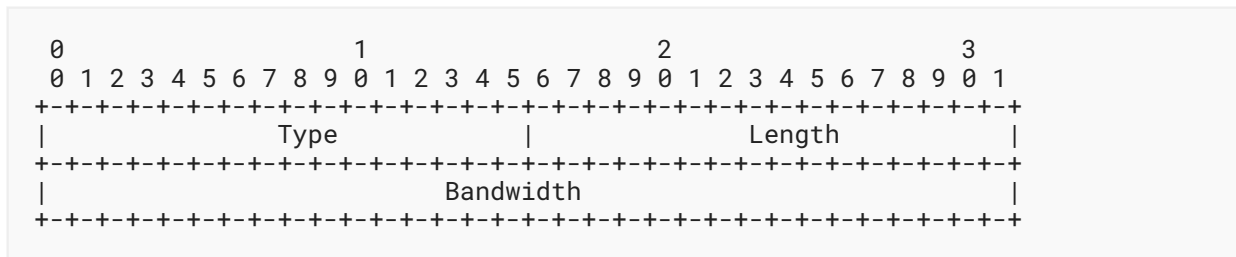


Figure 14: SR Bandwidth Constraint Sub-TLV Format

Where:

Type: 1210

Length: 4 octets

Bandwidth: 4 octets that specify the desired bandwidth in unit of bytes per second in IEEE floating point format [IEEE754].

5.6.4. SR Disjoint Group Constraint Sub-TLV

The SR Disjoint Group Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to carry the disjointness constraint associated with the candidate path. The disjointness between two SR Policy candidate paths is expressed by associating them with the same disjoint group identifier and then specifying the type of disjointness required between their paths. The types of disjointness are described in Section 3 of [RFC8800] where the level of disjointness increases in the order: link, node, SRLG, Node + SRLG. The computation is expected to achieve the highest level of disjointness requested; when that is not possible, then fall back to a lesser level progressively based on the levels indicated. Only a single instance of this sub-TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

The sub-TLV has the following format:

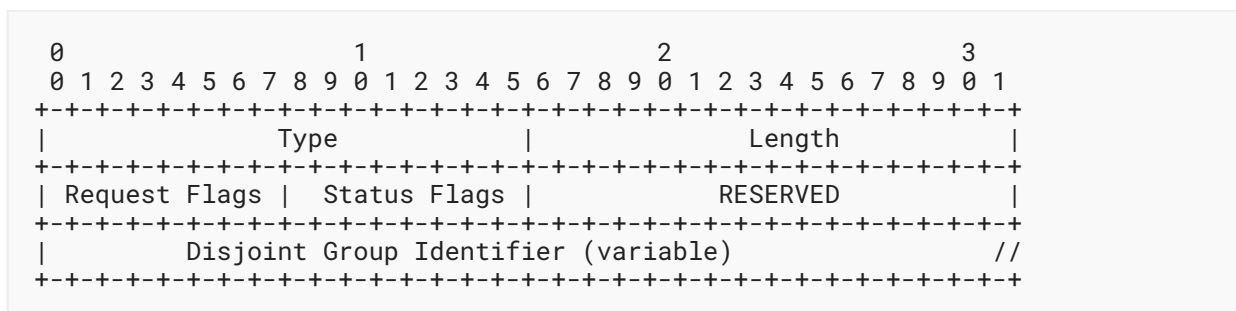


Figure 15: SR Disjoint Group Constraint Sub-TLV Format

Where:

Type: 1211

Length: Variable. Minimum of 8 octets.

Request Flags: 1 octet to indicate the level of disjointness requested as specified in the form of flags. The following flags are defined, and the other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.

```

  0 1 2 3 4 5 6 7
+---+---+---+---+
|S|N|L|F|I|   |
+---+---+---+---+

```

Where:

S-Flag: Indicates that SRLG disjointness is requested when set and that SRLG disjointness is not requested when clear.

N-Flag: Indicates that node disjointness is requested when set and that node disjointness is not requested when clear.

L-Flag: Indicates that link disjointness is requested when set and that the link disjointness is not requested when clear.

F-Flag: Indicates that the computation may fall back to a lower level of disjointness amongst the ones requested when all cannot be achieved when set and that fallback to a lower level of disjointness is not allowed when clear.

I-Flag: Indicates that the computation may fall back to the default best path (e.g., an IGP path) in case none of the desired disjointness can be achieved when set and that fallback to the default best path is not allowed when clear.

Status Flags: 1 octet to indicate the level of disjointness that has been achieved by the computation as specified in the form of flags. The following flags are defined, and the other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.

```

  0 1 2 3 4 5 6 7
+---+---+---+---+
|S|N|L|F|I|X|   |
+---+---+---+---+

```

Where:

S-Flag: Indicates that SRLG disjointness is achieved when set and that SRLG disjointness is not achieved when clear.

N-Flag: Indicates that node disjointness is achieved when set and that node disjointness was not achieved when clear.

L-Flag: Indicates that link disjointness is achieved when set and that link disjointness was not achieved when clear.

F-Flag: Indicates that the computation has fallen back to a lower level of disjointness than requested when set and that there has been no fallback to a lower level of disjointness when clear.

I-Flag: Indicates that the computation has fallen back to the best path (e.g., an IGP path) and disjointness has not been achieved when set and that there has been no fallback to the best path when clear.

X-Flag: Indicates that the disjointness constraint could not be achieved and hence the path has been invalidated when set and that the path has not been invalidated due to unmet disjointness constraints when clear.

RESERVED: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Disjoint Group Identifier: 4-octet value that is the group identifier for a set of disjoint paths. Alternatively, this field **MAY** contain the entire PCEP ASSOCIATION Object as specified in [Section 6.1](#) of [\[RFC8697\]](#) (including its optional TLVs) when PCEP is used for the signaling of the SR Policy candidate path and where the BGP-LS Producer is unable to determine the group identifier that can be accommodated in a 4-octet value (since PCEP supports multiple methods of encoding an association identifier). Note that the parsing of the PCEP object is expected to be performed only by the BGP-LS Consumer (hence, outside the scope of this document) and not by any BGP Speaker as specified in [\[RFC9552\]](#). If the PCEP object size is such that the update for a single SR Policy Candidate Path NLRI would exceed the supported BGP message size by the implementation, then the PCEP ASSOCIATION Object **MUST NOT** be encoded and this sub-TLV skipped along with an error log. Refer to [Section 5.3](#) of [\[RFC9552\]](#) for discussion on implications of encoding large sets of information into BGP-LS.

5.6.5. SR Bidirectional Group Constraint Sub-TLV

The SR Bidirectional Group Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to carry the bidirectional constraint associated with the candidate path. The bidirectional relationship between two SR Policy candidate paths is expressed by associating them with the same bidirectional group identifier and then specifying the type of bidirectional routing required between their paths. Only a single instance of this sub-TLV is advertised for a given candidate path. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per [Section 8.2.2](#) of [\[RFC9552\]](#)) is used and the rest are ignored.

The sub-TLV has the following format:

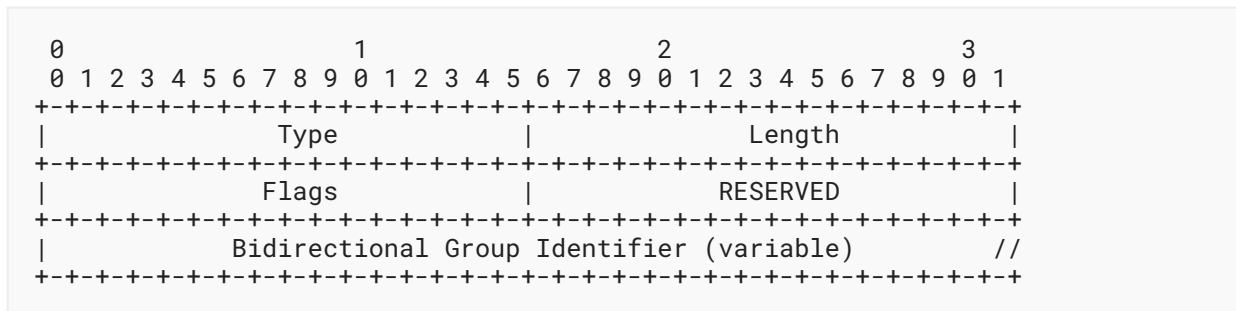


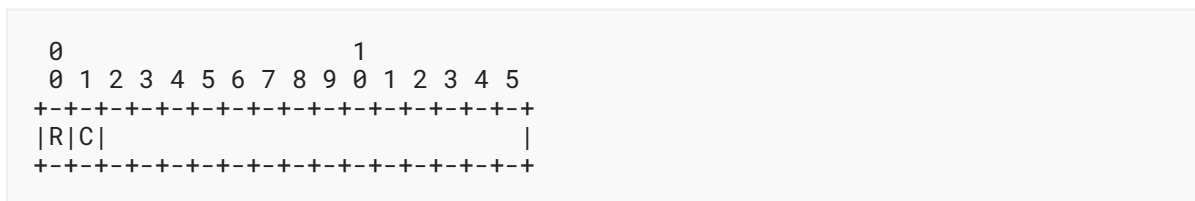
Figure 16: SR Bidirectional Group Constraint Sub-TLV Format

Where:

Type: 1214

Length: Variable. Minimum of 8 octets.

Flags: 2 octets to indicate the bidirectional path setup information as specified in the form of flags. The following flags are defined, and the other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.



Where:

R-Flag: Indicates that the candidate path of the SR Policy forms the reverse path when the R-flag is set. If the R-flag is clear, the candidate path forms the forward path.

C-Flag: Indicates that the bidirectional path is co-routed when set and that the bidirectional path is not co-routed when clear.

RESERVED: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Bidirectional Group Identifier: 4-octet value that is the group identifier for a set of bidirectional paths. Alternatively, this field **MAY** contain the entire PCEP ASSOCIATION Object as specified in Section 6.1 of [RFC8697] (including its optional TLVs) when PCEP is used for the signaling of the SR Policy candidate path and where the BGP-LS Producer is unable to determine the group identifier that can be accommodated in a 4-octet value (since PCEP supports multiple methods of encoding an association identifier). Note that the parsing of the PCEP object is expected to be performed only by the BGP-LS Consumer (hence, outside the scope of this document) and not by any BGP Speaker as specified in [RFC9552]. If the PCEP object size is such that the update for a single SR Policy Candidate Path NLRI would exceed the supported

BGP message size by the implementation, then the PCEP ASSOCIATION Object **MUST NOT** be encoded and this sub-TLV skipped along with an error log. Refer to [Section 5.3](#) of [RFC9552] for discussion on implications of encoding large sets of information into BGP-LS.

5.6.6. SR Metric Constraint Sub-TLV

The SR Metric Constraint sub-TLV is an optional sub-TLV of the SR Candidate Path Constraints TLV that is used to report the optimization metric of the candidate path. For a dynamic path computation, it is used to report the optimization metric used along with its parameters. For an explicit path, this sub-TLV **MAY** be used to report the metric margin or is bound to be used for validation (i.e., the path is invalidated if the metric is beyond specified values). Multiple instances of this sub-TLV may be used to report different metric type uses.

The sub-TLV has the following format:

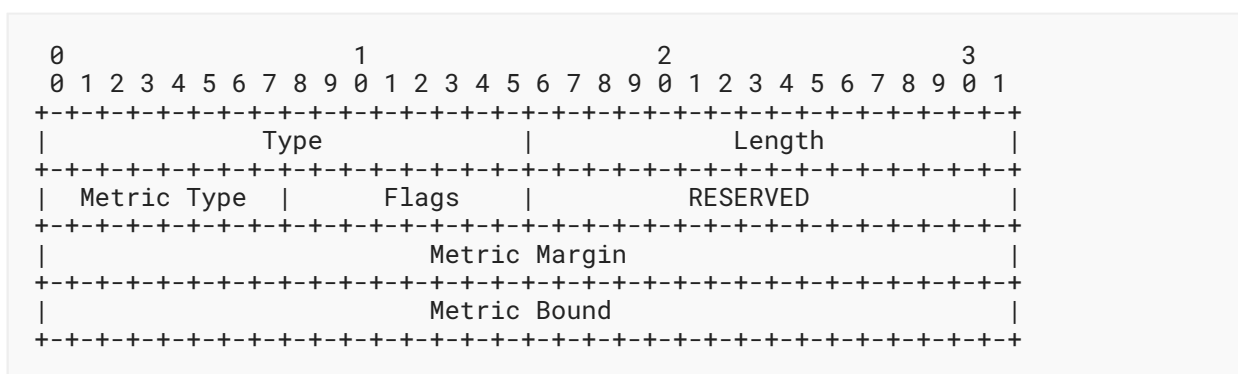


Figure 17: SR Metric Constraint Sub-TLV Format

Where:

Type: 1215

Length: 12 octets

Metric Type: 1-octet field that identifies the type of metric being used. Below is a list of metric types introduced by this document along with references for each. Where the references are for IS-IS and OSPF specifications, those metric types are defined for a link while in the SR Policy context those relate to the candidate path or the segment list. The metric type code points that may be used in this sub-TLV are also listed in [Section 8.6](#) of this document. Note that the metric types in this field are taken from the "BGP-LS SR Policy Metric Types" IANA registry, which includes IGP Metric Types as well as metric types specific to SR Policy path computation (i.e., the metric types are not from the "IGP Metric-Type" registry). Additional metric types may be introduced by future documents. This document does not make any assumptions about a smaller metric value being better than a higher metric value; that is something that is dependent on the semantics of the specific metric type. This document uses the words "best" and "worst" to abstract this aspect when referring to metric margins and bounds.

Type 0: IGP:

This is specified in [Section 3](#) of [\[RFC5305\]](#) for IS-IS and is known as the default metric. This is specified in [\[RFC2328\]](#) for OSPFv2 and in [\[RFC5340\]](#) for OSPFv3 and is known as the metric in both.

Type 1: Min Unidirectional Delay:

This is specified in [Section 4.2](#) of [\[RFC8570\]](#) for IS-IS and in [Section 4.2](#) of [\[RFC7471\]](#) for OSPFv2/OSPFv3.

Type 2: TE:

This is specified in [Section 3.7](#) of [\[RFC5305\]](#) for IS-IS as the TE default metric, in [Section 2.5.5](#) of [\[RFC3630\]](#) for OSPFv2, and in [Section 4](#) of [\[RFC5329\]](#) for OSPFv3.

Type 3: Hop Count:

This is specified in [Section 7](#) of [\[RFC5440\]](#).

Type 4: SID List Length:

This is specified in [Section 4.5](#) of [\[RFC8664\]](#).

Type 5: Bandwidth:

This is specified in [Section 4](#) of [\[RFC9843\]](#).

Type 6: Avg Unidirectional Delay:

This is specified in [Section 4.1](#) of [\[RFC8570\]](#) for IS-IS and in [Section 4.1](#) of [\[RFC7471\]](#) for OSPFv2/OSPFv3.

Type 7: Unidirectional Delay Variation:

This is specified in [Section 4.3](#) of [\[RFC8570\]](#) for IS-IS and in [Section 4.3](#) of [\[RFC7471\]](#) for OSPFv2/OSPFv3.

Type 8: Loss:

This is specified in [Section 4.4](#) of [\[RFC8570\]](#) for IS-IS and in [Section 4.4](#) of [\[RFC7471\]](#) for OSPFv2/OSPFv3.

Types 128 to 255 (both inclusive): User Defined:

This is specified in [Section 2](#) of [\[RFC9843\]](#) for IS-IS and OSPF.

Flags: 1-octet field that indicates the validity of the metric fields and their semantics. The following bit positions are defined, and the other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.

```

  0 1 2 3 4 5 6 7
+--+--+--+--+--+--+
|O|M|A|B|      |
+--+--+--+--+--+--+

```

Where:

O-Flag:

Indicates that this is the optimization metric being reported for a dynamic candidate path when set and that the metric is not the optimization metric when clear. This bit **MUST NOT** be set in more than one instance of this TLV for a given candidate path advertisement.

M-Flag: Indicates that the metric margin allowed is specified when set and that the metric margin allowed is not specified when clear.

A-Flag: Indicates that the metric margin is specified as an absolute value when set and that the metric margin is expressed as a percentage of the minimum metric when clear.

B-Flag: Indicates that the metric bound allowed for the path is specified when set and that the metric bound is not specified when clear.

RESERVED: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Metric Margin: 4-octet value that indicates the metric margin when the M-flag is set. The metric margin is specified, depending on the A-flag, as either an absolute value or a percentage of the best computed path metric based on the specified constraints for path calculation. The metric margin allows for the metric value of the computed path to vary (depending on the semantics of the specific metric type) from the best metric value possible to optimizing for other factors (that are not specified as constraints) such as bandwidth availability, minimal SID stack depth, and the maximizing of ECMP for the computed SR path.

Metric Bound: 4-octet value that indicates the worst metric value (depending on the semantics of the specific metric type) allowed when the B-flag is set. If the computed path metric crosses the specified bound value, then the path is considered invalid.

The absolute metric margin and the metric bound values are encoded as specified for each metric type. For metric types that are smaller than 4 octets in size, the most significant bits are filled with zeros. The percentage metric margin is encoded as an unsigned integer percentage value.

5.7. SR Segment List TLV

The SR Segment List TLV is used to report a single SID list of a candidate path. Multiple instances of this TLV may be used to report multiple SID lists of a candidate path.

The TLV has the following format:

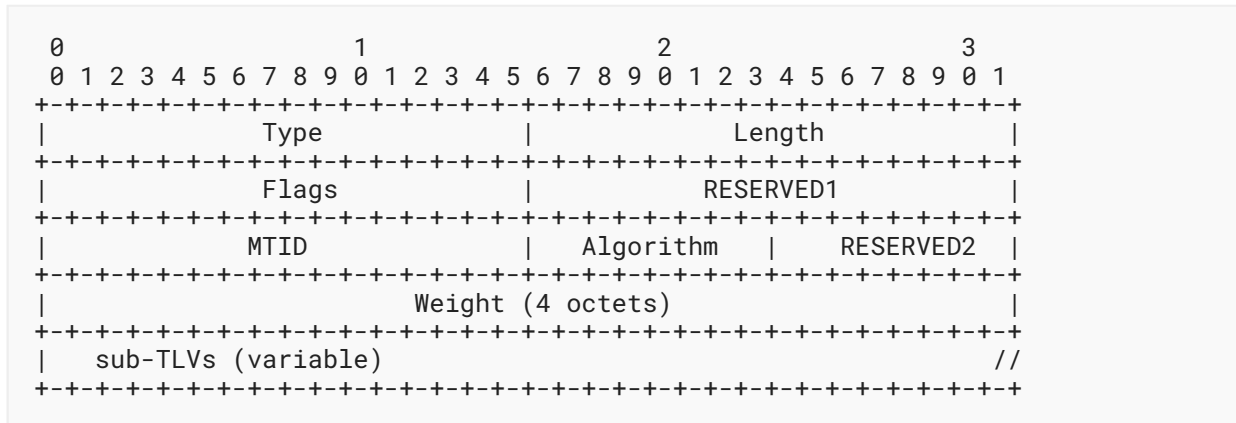


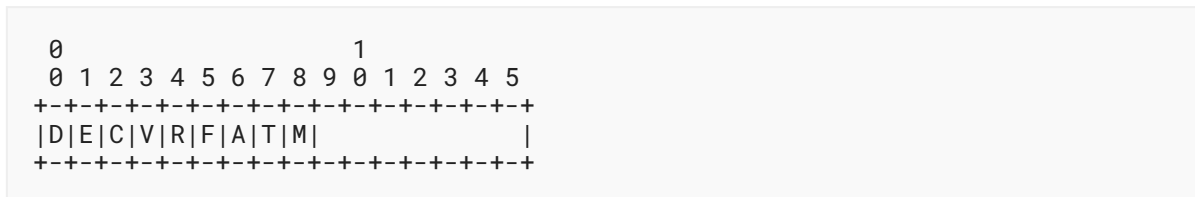
Figure 18: SR Segment List TLV Format

Where:

Type: 1205

Length: Variable

Flags: 2-octet field that indicates the attribute and status of the SID list. The following bit positions are defined, and the semantics are described in detail in [RFC9256]. Other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.



Where:

D-Flag: Indicates that the SID list consists of SRv6 SIDs when set and SR/MPLS labels when clear.

E-Flag: Indicates that the SID list is associated with an explicit candidate path when set and a dynamic candidate path when clear. All segment lists of a given candidate path **MUST** be either explicit or dynamic. In case of inconsistency, the receiver **MAY** consider them all to be dynamic.

C-Flag: Indicates that the SID list has been computed for a dynamic path when set. It is always reported as set for explicit paths. When clear, it indicates that the SID list has not been computed for a dynamic path.

V-Flag: Indicates that the SID list has passed verification or its verification was not required when set and that it failed verification when clear.

R-Flag: Indicates that the first Segment has been resolved when set and that it failed resolution when clear.

F-Flag: Indicates that the computation for the dynamic path failed when set and that it succeeded (or was not required in case of an explicit path) when clear.

A-Flag: Indicates that all the SIDs in the SID list belong to the specified algorithm when set and that not all the SIDs belong to the specified algorithm when clear.

T-Flag: Indicates that all the SIDs in the SID list belong to the specified topology (identified by the multi-topology ID) when set and that not all the SIDs belong to the specified topology when clear.

M-Flag: Indicates that the SID list has been removed from the forwarding plane due to fault detection by a monitoring mechanism (e.g., Bidirectional Forwarding Detection (BFD)) when set and that no fault is detected or no monitoring is being done when clear.

RESERVED1: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

MTID: 2 octets that indicate the multi-topology identifier of the IGP topology that is to be used when the T-flag is set.

Algorithm: 1 octet that indicates the algorithm of the SIDs used in the SID list when the A-flag is set. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

RESERVED2: 1 octet. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Weight: 4-octet field that indicates the weight associated with the SID list for weighted load balancing. Refer to Sections 2.2 and 2.11 of [RFC9256].

Sub-TLVs: Variable and contain the ordered set of Segments and any other optional attributes associated with the specific SID list.

The SR Segment sub-TLV (defined in [Section 5.7.1](#)) **MUST** be included as an ordered set of sub-TLVs within the SR Segment List TLV when the SID list is not empty. A SID list may be empty in certain situations (e.g., for a dynamic path) where the headend has not yet performed the computation and hence not derived the segments required for the path. In such cases where the SID list is empty, the SR Segment List TLV **MUST NOT** include any SR Segment sub-TLVs.

5.7.1. SR Segment Sub-TLV

The SR Segment sub-TLV describes a single segment in a SID list. One or more instances of this sub-TLV in an ordered manner constitute a SID list for an SR Policy candidate path. It is a sub-TLV of the SR Segment List TLV and it has the following format:

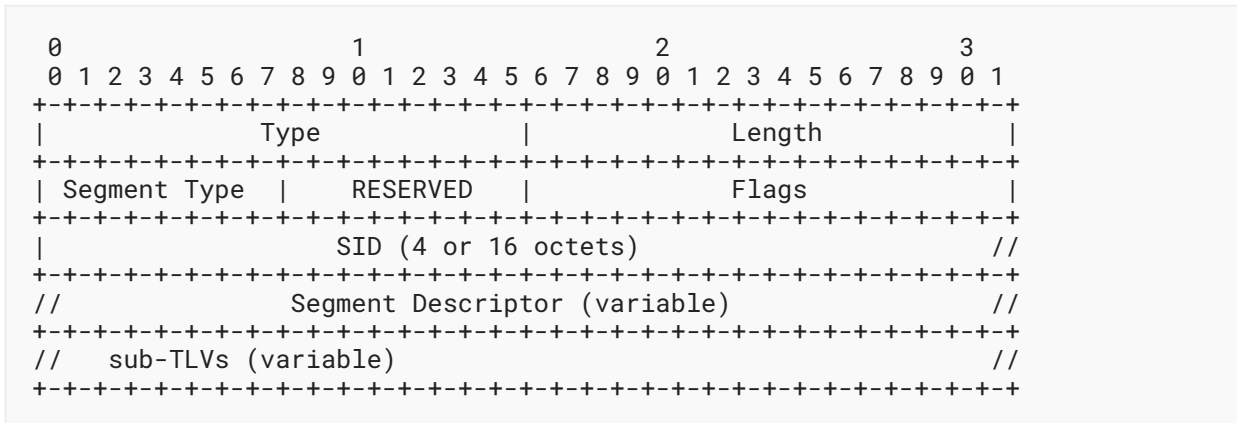


Figure 19: SR Segment Sub-TLV Format

Where:

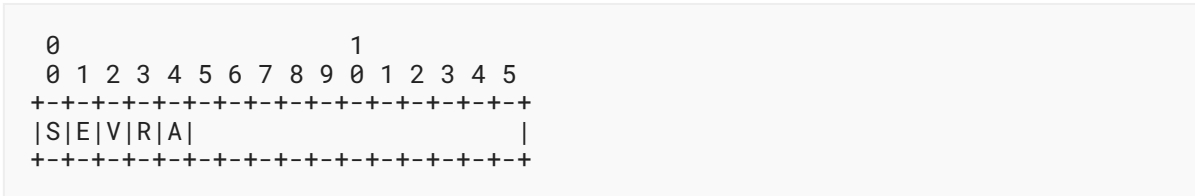
Type: 1206

Length: Variable

Segment Type: 1 octet that indicates the type of segment. Initial values are specified by this document (see Section 5.7.1.1 for details). Additional segment types are possible but are out of scope for this document.

RESERVED: 1 octet. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Flags: 2-octet field that indicates the attribute and status of the Segment and its SID. The following bit positions are defined, and the semantics are described in Section 5 of [RFC9256]. Other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.



Where:

S-Flag: Indicates the presence of the SID value in the SID field when set and no value when clear.

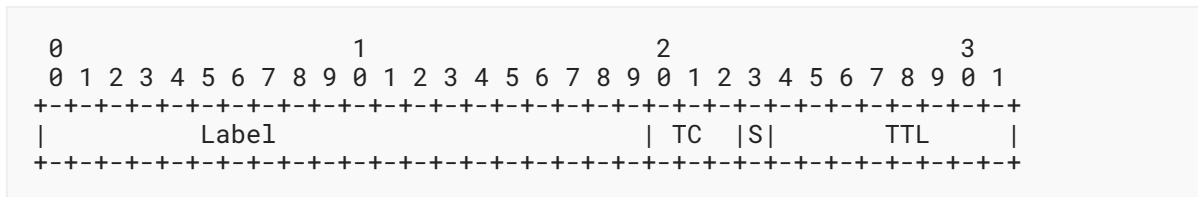
E-Flag: Indicates that the SID value is an explicitly provisioned value (locally on headend or via controller/PCE) when set and is a dynamically resolved value by headend when clear.

V-Flag: Indicates that the SID has passed verification or did not require verification when set. When the V-flag is clear, it indicates the SID has failed verification.

R-Flag: Indicates that the SID has been resolved or did not require resolution (e.g., because it is not the first SID) when set. When the R-flag is clear, it indicates the SID has failed resolution.

A-Flag: Indicates that the Algorithm indicated in the segment descriptor is valid when set. When clear, it indicates that the headend is unable to determine the algorithm of the SID.

SID: 4 octets carrying the MPLS Label or 16 octets carrying the SRv6 SID based on the Segment Type. When carrying the MPLS Label, as shown in the figure below, the TC, S, and TTL (total of 12 bits) are RESERVED and **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.



Segment Descriptor: Variable size segment descriptor based on the type of segment (refer to [Section 5.7.1.1](#) for details).

Sub-Sub-TLVs: Variable and contain any other optional attributes associated with the specific segment.

The SRv6 Endpoint Behavior TLV (1250) and the SRv6 SID Structure TLV (1252) defined in [\[RFC9514\]](#) are used as sub-sub-TLVs of the SR Segment sub-TLV. These two sub-sub-TLVs are used to optionally indicate the SRv6 Endpoint behavior and SID structure when advertising the SRv6-specific segment types.

5.7.1.1. Segment Descriptors

[Section 4](#) of [\[RFC9256\]](#) defines multiple types of segments and their descriptions. This section defines the encoding of the segment descriptors for each of those segment types to be used in the Segment sub-TLV described previously in [Section 5.7.1](#).

The following types are currently defined, and their mappings to the respective segment types are defined in [\[RFC9256\]](#):

Type	Segment Description
1	(Type A) SR-MPLS Label
2	(Type B) SRv6 SID
3	(Type C) IPv4 Prefix with optional SR Algorithm
4	(Type D) IPv6 Global Prefix with optional SR Algorithm for SR-MPLS
5	(Type E) IPv4 Prefix with Local Interface ID

Type	Segment Description
6	(Type F) IPv4 Addresses for link endpoints as Local, Remote pair
7	(Type G) IPv6 Prefix and Interface ID for link endpoints as Local, Remote pair for SR-MPLS
8	(Type H) IPv6 Addresses for link endpoints as Local, Remote pair for SR-MPLS
9	(Type I) IPv6 Global Prefix with optional SR Algorithm for SRv6
10	(Type J) IPv6 Prefix and Interface ID for link endpoints as Local, Remote pair for SRv6
11	(Type K) IPv6 Addresses for link endpoints as Local, Remote pair for SRv6

Table 1: SR Segment Types

5.7.1.1.1. Type 1: Segment Type A

The Segment is an SR-MPLS type and is specified simply as the label. The format of its segment descriptor is as follows:



Figure 20: Type 1 Segment Descriptor

Where:

Algorithm: 1-octet value that indicates the algorithm used for picking the SID. This is valid only when the A-flag has been set in the Segment TLV. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

5.7.1.1.2. Type 2: Segment Type B

The Segment is an SRv6 type and is specified simply as SRv6 SID. The format of its segment descriptor is as follows:


```

      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
+---+---+---+---+---+---+
|   Algorithm   |
+---+---+---+---+---+---+

```

Figure 21: Type 2 Segment Descriptor

Where:

Algorithm: 1-octet value that indicates the algorithm used for picking the SID. This is valid only when the A-flag has been set in the Segment TLV. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

5.7.1.1.3. Type 3: Segment Type C

The Segment is an SR-MPLS Prefix SID type and is specified as an IPv4 node address. The format of its segment descriptor is as follows:

```

      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
+---+---+---+---+---+---+
|   Algorithm   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     IPv4 Node Address (4 octets)                                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 22: Type 3 Segment Descriptor

Where:

Algorithm: 1-octet value that indicates the algorithm used for picking the SID. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

IPv4 Node Address: 4-octet value that carries the IPv4 address associated with the node.

5.7.1.1.4. Type 4: Segment Type D

The Segment is an SR-MPLS Prefix SID type and is specified as an IPv6 node global address. The format of its segment descriptor is as follows:

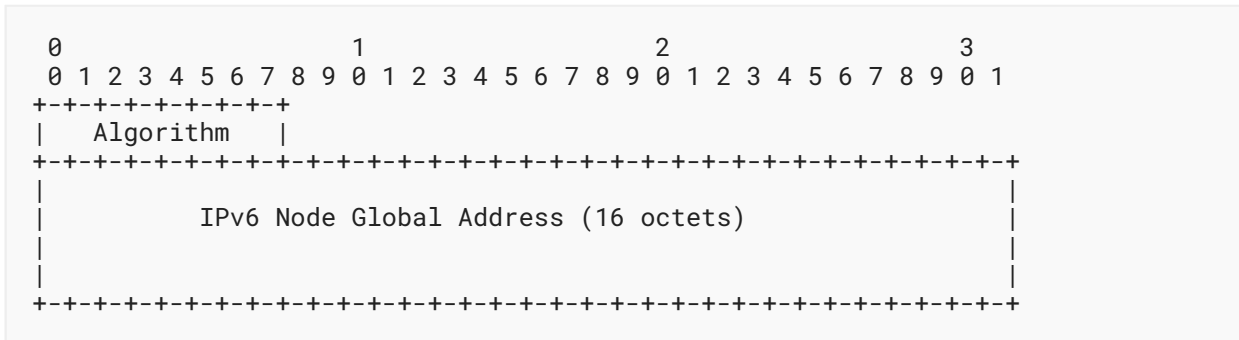


Figure 23: Type 4 Segment Descriptor

Where:

Algorithm: 1-octet value that indicates the algorithm used for picking the SID. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

IPv6 Node Global Address: 16-octet value that carries the IPv6 global address associated with the node.

5.7.1.1.5. Type 5: Segment Type E

The Segment is an SR-MPLS Adjacency SID type and is specified as an IPv4 node address along with the local interface ID on that node. The format of its segment descriptor is as follows:

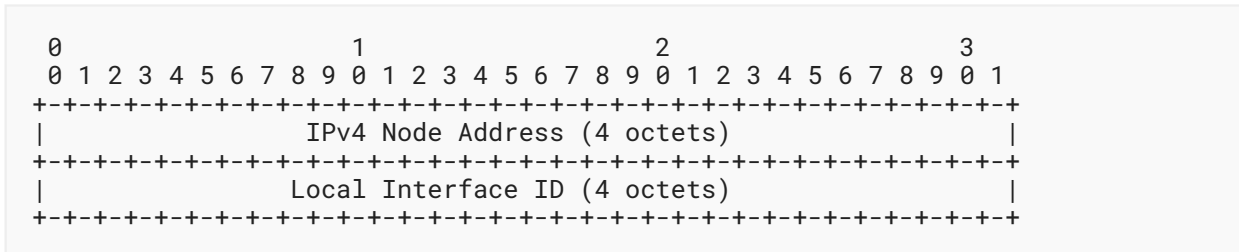


Figure 24: Type 5 Segment Descriptor

Where:

IPv4 Node Address: 4-octet value that carries the IPv4 address associated with the node.

Local Interface ID: 4-octet value that carries the local interface ID of the node identified by the Node Address.

5.7.1.1.6. Type 6: Segment Type F

The Segment is an SR-MPLS Adjacency SID type and is specified as a pair of IPv4 local and remote interface addresses. The format of its segment descriptor is as follows:

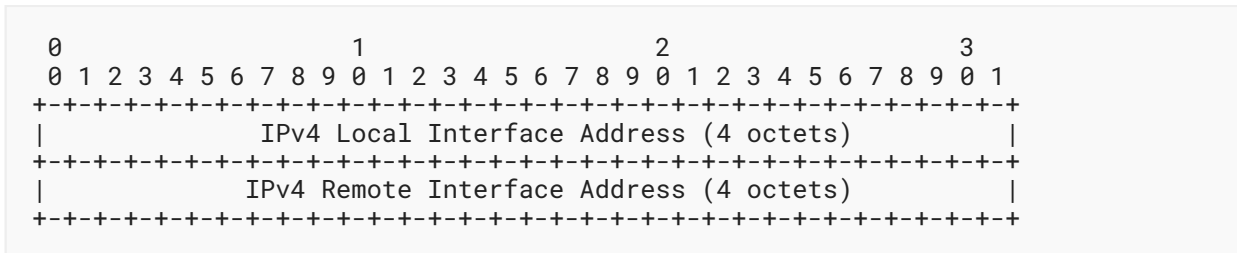


Figure 25: Type 6 Segment Descriptor

Where:

IPv4 Local Interface Address: 4-octet value that carries the local IPv4 address associated with the node's interface.

IPv4 Remote Interface Address: 4-octet value that carries the remote IPv4 address associated with the interface on the node's neighbor. This is optional and **MAY** be set to 0 when not used (e.g., when identifying point-to-point links).

5.7.1.1.7. Type 7: Segment Type G

The Segment is an SR-MPLS Adjacency SID type and is specified as a pair of IPv6 global address and interface ID for local and remote nodes. The format of its segment descriptor is as follows:

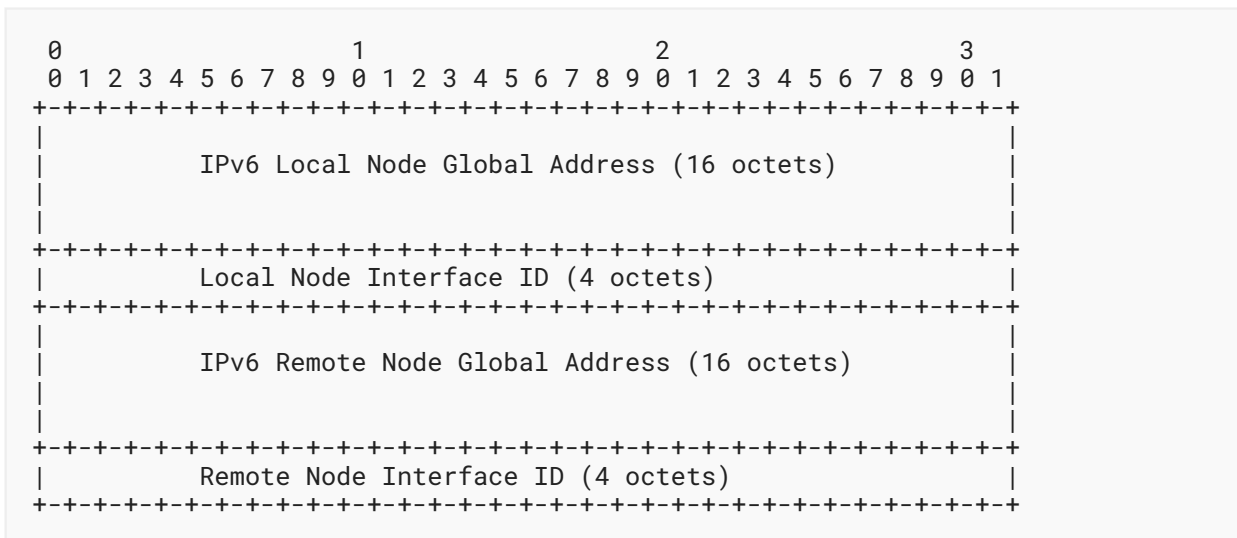


Figure 26: Type 7 Segment Descriptor

Where:

IPv6 Local Node Global Address: 16-octet value that carries the IPv6 global address associated with the local node.

Local Node Interface ID: 4-octet value that carries the interface ID of the local node identified by the Local Node Address.

IPv6 Remote Node Global Address: 16-octet value that carries the IPv6 global address associated with the remote node. This is optional and **MAY** be set to 0 when not used (e.g., when identifying point-to-point links).

Remote Node Interface ID: 4-octet value that carries the interface ID of the remote node identified by the Remote Node Address. This is optional and **MAY** be set to 0 when not used (e.g., when identifying point-to-point links).

5.7.1.1.8. Type 8: Segment Type H

The Segment is an SR-MPLS Adjacency SID type and is specified as a pair of IPv6 global addresses for local and remote interface addresses. The format of its segment descriptor is as follows:

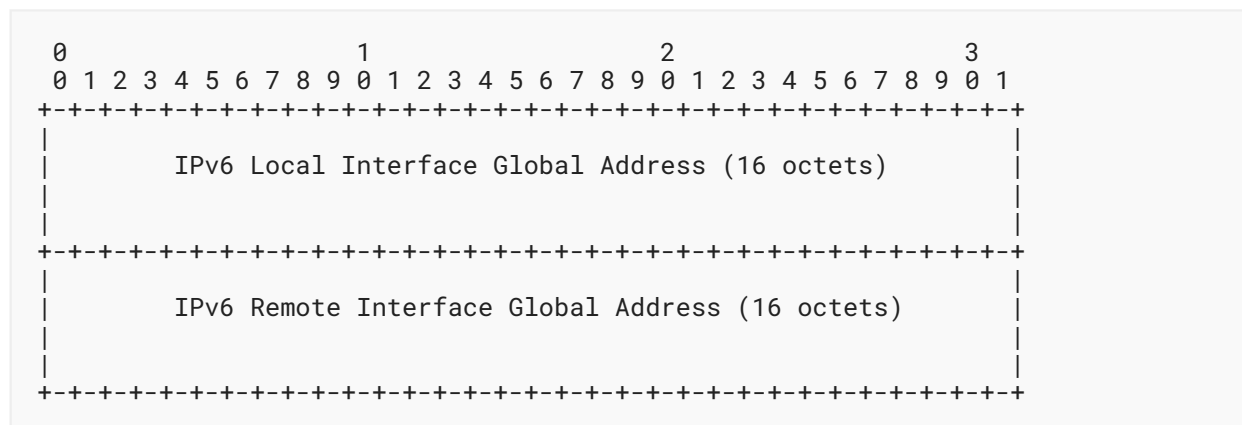


Figure 27: Type 8 Segment Descriptor

Where:

IPv6 Local Global Address: 16-octet value that carries the local IPv6 address associated with the node's interface.

IPv6 Remote Global Address: 16-octet value that carries the remote IPv6 address associated with the interface on the node's neighbor.

5.7.1.1.9. Type 9: Segment Type I

The Segment is an SRv6 END SID type and is specified as an IPv6 node global address. The format of its segment descriptor is as follows:



Figure 28: Type 9 Segment Descriptor

Where:

Algorithm: 1-octet value that indicates the algorithm used for picking the SID. The algorithm values are from the "IGP Algorithm Types" IANA registry under the "Interior Gateway Protocol (IGP) Parameters" registry group.

IPv6 Node Global Address: 16-octet value that carries the IPv6 global address associated with the node.

5.7.1.1.10. Type 10: Segment Type J

The Segment is an SRv6 END.X SID type and is specified as a pair of IPv6 global address and interface ID for local and remote nodes. The format of its segment descriptor is as follows:

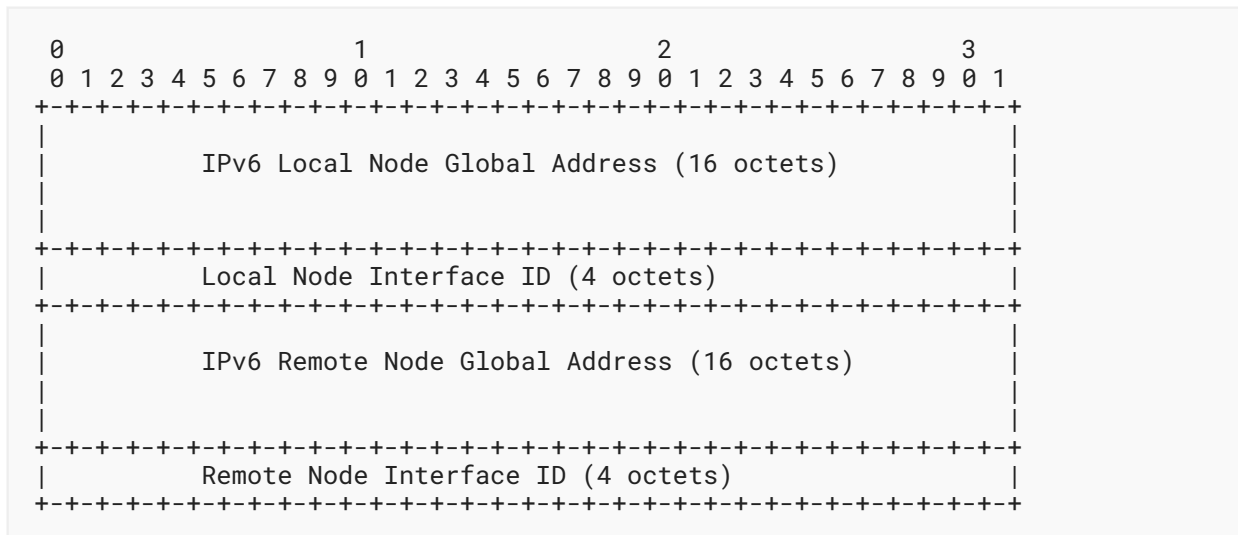


Figure 29: Type 10 Segment Descriptor

Where:

IPv6 Local Node Global Address: 16-octet value that carries the IPv6 global address associated with the local node.

Local Node Interface ID: 4-octet value that carries the interface ID of the local node identified by the Local Node Address.

IPv6 Remote Node Global Address: 16-octet value that carries the IPv6 global address associated with the remote node. This is optional and **MAY** be set to 0 when not used (e.g., when identifying point-to-point links).

Remote Node Interface ID: 4-octet value that carries the interface ID of the remote node identified by the Remote Node Address. This is optional and **MAY** be set to 0 when not used (e.g., when identifying point-to-point links).

5.7.1.1.11. Type 11: Segment Type K

The Segment is an SRv6 END.X SID type and is specified as a pair of IPv6 global addresses for local and remote interface addresses. The format of its segment descriptor is as follows:

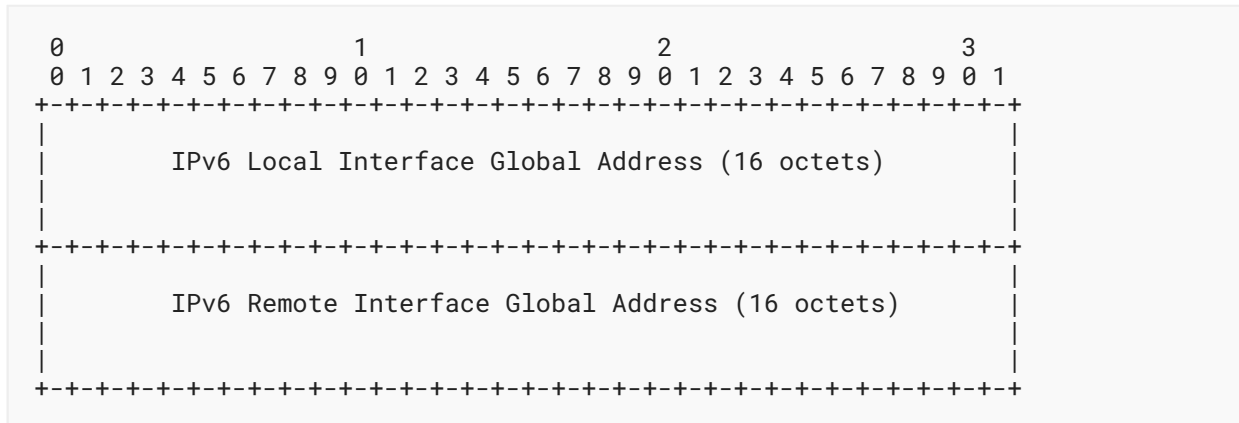


Figure 30: Type 11 Segment Descriptor

Where:

IPv6 Local Global Address: 16-octet value that carries the local IPv6 address associated with the node's interface.

IPv6 Remote Global Address: 16-octet value that carries the remote IPv6 address associated with the interface on the node's neighbor.

5.7.2. SR Segment List Metric Sub-TLV

The SR Segment List Metric sub-TLV reports the computed metric of the specific SID list. It is used to report the type of metric and its computed value by the computation entity (i.e., either the headend or the controller when the path is delegated) when available. More than one instance of this sub-TLV may be present in the SR Segment List to report metric values of different metric types. The metric margin and bound may be optionally reported using this sub-TLV when this information is not being reported using the SR Metric Constraint sub-TLV (refer to [Section 5.6.6](#)) at the SR Policy candidate path level.

It is a sub-TLV of the SR Segment List TLV and has the following format:

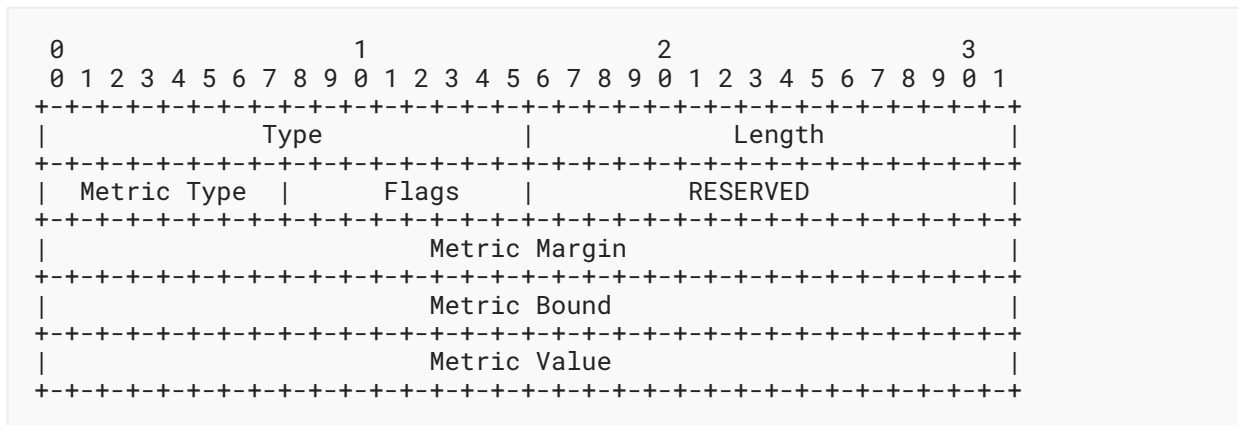


Figure 31: SR Segment List Metric Sub-TLV Format

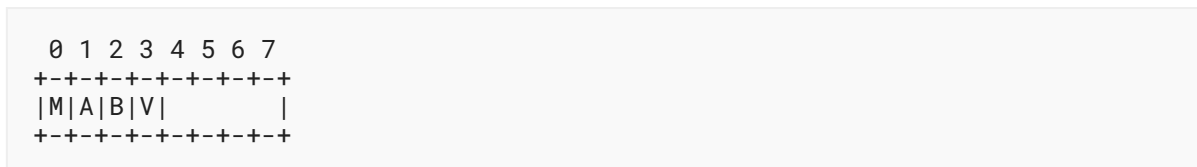
Where:

Type: 1207

Length: 16 octets

Metric Type: 1-octet field that identifies the type of metric. The semantics are the same as the metric type field in the SR Metric Constraint sub-TLV in [Section 5.6.6](#) of this document.

Flags: 1-octet field that indicates the validity of the metric fields and their semantics. The following bit positions are defined, and the other bits **MUST** be cleared by the originator and **MUST** be ignored by a receiver.



Where:

M-Flag: Indicates that the metric margin allowed for this path computation is specified when set and that the metric margin allowed is not specified when clear.

A-Flag: Indicates that the metric margin is specified as an absolute value when set and that the metric margin is expressed as a percentage of the minimum metric when clear.

B-Flag: Indicates that the metric bound allowed for the path is specified when set and that the metric bound is not specified when clear.

V-Flag: Indicates that the computed metric value is being reported when set and that the computed metric value is not being reported when clear.

RESERVED: 2 octets. **MUST** be set to 0 by the originator and **MUST** be ignored by a receiver.

Metric Margin: 4-octet value that indicates the metric margin value when the M-flag is set. The metric margin is specified, depending on the A-flag, as either an absolute value or a percentage of the best computed path metric based on the specified constraints for path calculation. The metric margin allows for the metric value of the computed path to vary (depending on the semantics of the specific metric type) from the best metric value possible to optimizing for other factors (that are not specified as constraints) such as bandwidth availability, minimal SID stack depth, and the maximizing of ECMP for the computed SR path.

Metric Bound: 4-octet value that indicates the worst metric value (depending on the semantics of the specific metric type) that is allowed when the B-flag is set. If the computed path metric crosses the specified bound value, then the path is considered invalid.

Metric Value: 4-octet value that indicates the metric of the computed path when the V-flag is set. This value is available and reported when the computation is successful and a valid path is available.

The absolute metric margin, metric bound, and metric values are encoded as specified for each metric type. For metric types that are smaller than 4 octets in size, the most significant bits are filled with zeros. The percentage metric margin is encoded as an unsigned integer percentage value.

5.7.3. SR Segment List Bandwidth Sub-TLV

The SR Segment List Bandwidth sub-TLV is an optional sub-TLV used to report the bandwidth allocated to the specific SID list by the path computation entity. Only a single instance of this sub-TLV is advertised for a given segment list. If multiple instances are present, then the first valid one (i.e., not determined to be malformed as per Section 8.2.2 of [RFC9552]) is used and the rest are ignored.

It is a sub-TLV of the SR Segment List TLV and has the following format:

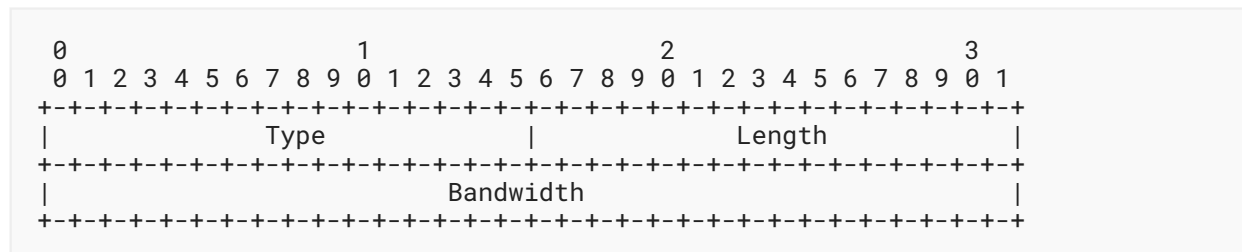


Figure 32: SR Segment List Bandwidth Sub-TLV Format

Where:

Type: 1216

Length: 4 octets

Sections 5.1 and 5.2 is included to report the BSID of the candidate path when one is either specified or allocated by the headend. The constraints and the optimization metric for the SR Policy candidate path are reported using the SR Candidate Path Constraints TLV and its sub-TLVs as described in Section 5.6. The SR Segment List TLV is included for each SID list(s) associated with the candidate path. Each SR Segment List TLV in turn includes an SR Segment sub-TLV(s) to report the segment(s) and its status. The SR Segment List Metric sub-TLV is used to report the metric values at an individual SID list level.

7. Manageability Considerations

The existing BGP operational and management procedures apply to this document. No new procedures are defined in this document. The considerations as specified in [RFC9552] apply to this document.

In general, the SR Policy headend nodes are responsible for the advertisement of SR Policy state information.

8. IANA Considerations

This section describes the code point allocations by IANA for this document.

8.1. BGP-LS NLRI Types

IANA maintains a registry called "BGP-LS NLRI Types" under the "Border Gateway Protocol - Link State (BGP-LS) Parameters" registry group.

The following NLRI Type code point has been allocated by IANA:

Type	NLRI Type	Reference
5	SR Policy Candidate Path NLRI	RFC 9857

Table 2: NLRI Type Code Point

8.2. BGP-LS Protocol-IDs

IANA maintains a registry called "BGP-LS Protocol-IDs" under the "Border Gateway Protocol - Link State (BGP-LS) Parameters" registry group.

The following Protocol-ID code point has been allocated by IANA:

Protocol-ID	NLRI information source protocol	Reference
9	Segment Routing	RFC 9857

Table 3: Protocol-ID Code Point

8.3. BGP-LS TLVs

IANA maintains a registry called "BGP-LS NLRI and Attribute TLVs" under the "Border Gateway Protocol - Link State (BGP-LS) Parameters" registry group.

The following table lists the TLV code points that have been allocated by IANA:

TLV Code Point	Description	Reference
554	SR Policy Candidate Path Descriptor	RFC 9857
1201	SR Binding SID	RFC 9857
1202	SR Candidate Path State	RFC 9857
1203	SR Candidate Path Name	RFC 9857
1204	SR Candidate Path Constraints	RFC 9857
1205	SR Segment List	RFC 9857
1206	SR Segment	RFC 9857
1207	SR Segment List Metric	RFC 9857
1208	SR Affinity Constraint	RFC 9857
1209	SR SRLG Constraint	RFC 9857
1210	SR Bandwidth Constraint	RFC 9857
1211	SR Disjoint Group Constraint	RFC 9857
1212	SRv6 Binding SID	RFC 9857
1213	SR Policy Name	RFC 9857
1214	SR Bidirectional Group Constraint	RFC 9857
1215	SR Metric Constraint	RFC 9857
1216	SR Segment List Bandwidth	RFC 9857
1217	SR Segment List Identifier	RFC 9857

Table 4: NLRI and Attribute TLV Code Points

8.4. SR Policy Protocol-Origin

Per this document, IANA has created and maintains a new registry called "SR Policy Protocol-Origin" under the "Segment Routing" registry group with the allocation policy of Expert Review [RFC8126] using the guidelines for designated experts as specified in [RFC9256]. This registry contains the code points allocated to the "Protocol-Origin" field defined in Section 4.

IANA has assigned the initial values as follows:

Code Point	Protocol-Origin	Reference
0	Reserved	RFC 9857
1	PCEP	RFC 9857
2	BGP SR Policy	RFC 9857
3	Configuration (CLI, YANG model via NETCONF, etc.)	RFC 9857
4-9	Unassigned	RFC 9857
10	PCEP (in PCEP or when BGP-LS Producer is PCE)	RFC 9857
11-19	Unassigned	RFC 9857
20	BGP SR Policy (in PCEP or when BGP-LS Producer is PCE)	RFC 9857
21-29	Unassigned	RFC 9857
30	Configuration (CLI, YANG model via NETCONF, etc. In PCEP or when BGP-LS Producer is PCE)	RFC 9857
31-250	Unassigned	RFC 9857
251-255	Reserved for Private Use	RFC 9857

Table 5: SR Policy Protocol-Origin Code Points

8.5. BGP-LS SR Segment Descriptors

Per this document, IANA has created a registry called "BGP-LS SR Segment Descriptor Types" under the "Border Gateway Protocol - Link State (BGP-LS) Parameters" registry group with the allocation policy of Expert Review [RFC8126] using the guidelines for designated experts as specified in [RFC9552]. There is also an additional guideline for the designated experts to maintain the alignment between the allocations in this registry with those in the "Segment Types" registry under the "Segment Routing" registry group. This requires that an allocation in the Segment Routing "Segment Types" registry is required before allocation can be done in the

"BGP-LS SR Segment Descriptor Types" registry for a new segment type. However, this does not mandate that the specification of a new Segment Routing Segment Type also requires the specification of its equivalent SR Segment Descriptor Type in BGP-LS; that can be done as and when required while maintaining alignment.

This registry contains the code points allocated to the "Segment Type" field defined in [Section 5.7.1](#) and described in [Section 5.7.1.1](#). IANA has assigned the initial values as follows:

Code Point	Segment Descriptor	Reference
0	Reserved	RFC 9857
1	Type A Segment	RFC 9857
2	Type B Segment	RFC 9857
3	Type C Segment	RFC 9857
4	Type D Segment	RFC 9857
5	Type E Segment	RFC 9857
6	Type F Segment	RFC 9857
7	Type G Segment	RFC 9857
8	Type H Segment	RFC 9857
9	Type I Segment	RFC 9857
10	Type J Segment	RFC 9857
11	Type K Segment	RFC 9857
12-255	Unassigned	RFC 9857

Table 6: BGP-LS SR Segment Descriptor Type Code Points

8.6. BGP-LS SR Policy Metric Type

Per this document, IANA has created a registry called "BGP-LS SR Policy Metric Types" under the "Border Gateway Protocol - Link State (BGP-LS) Parameters" registry group with the allocation policy of Expert Review [[RFC8126](#)] using the guidelines for designated experts as specified in [[RFC9552](#)]. This registry contains the code points allocated to the metric type field defined in [Section 5.7.2](#). IANA has assigned the initial values as follows:

Code Point	Metric Type	Reference
0	IGP	RFC 9857
1	Min Unidirectional Delay	RFC 9857
2	TE	RFC 9857
3	Hop Count	RFC 9857
4	SID List Length	RFC 9857
5	Bandwidth	RFC 9857
6	Avg Unidirectional Delay	RFC 9857
7	Unidirectional Delay Variation	RFC 9857
8	Loss	RFC 9857
9-127	Unassigned	RFC 9857
128-255	User Defined	RFC 9857

Table 7: SR Policy Metric Type Code Point

9. Security Considerations

Procedures and protocol extensions defined in this document do not affect the base BGP security model. See [RFC6952] for details. The security considerations of the base BGP-LS specification as described in [RFC9552] also apply.

The BGP-LS SR Policy extensions specified in this document enable TE and service programming use cases within an SR domain as described in [RFC9256]. SR operates within a trusted SR domain [RFC8402], and its security considerations also apply to BGP sessions when carrying SR Policy information. The SR Policies advertised to controllers and other applications via BGP-LS are expected to be used entirely within this trusted SR domain, i.e., within a single AS or between multiple ASes/domains within a single provider network. Therefore, precaution is necessary to ensure that the SR Policy information advertised via BGP sessions is limited to nodes and/or controllers/applications in a secure manner within this trusted SR domain. The general guidance for BGP-LS with respect to isolation of BGP-LS sessions from BGP sessions for other address-families (refer to the security considerations of [RFC9552]) may be used to ensure that the SR Policy information is not advertised to an External BGP (EBGP) peering session outside the SR domain by accident or error.

Additionally, it may be considered that the export of SR Policy information, as described in this document, constitutes a risk to the confidentiality of mission-critical or commercially sensitive information about the network (more specifically, endpoint/node addresses, SR SIDs, and the SR

Policies deployed). BGP peerings are not automatic and require configuration. Thus, it is the responsibility of the network operator to ensure that only trusted nodes (that include both routers and controller applications) within the SR domain are configured to receive such information.

10. References

10.1. Normative References

- [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>>.
- [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>>.
- [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>>.
- [RFC5329] Ishiguro, K., Manral, V., Davey, A., and A. Lindem, Ed., "Traffic Engineering Extensions to OSPF Version 3", RFC 5329, DOI 10.17487/RFC5329, September 2008, <<https://www.rfc-editor.org/info/rfc5329>>.
- [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>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [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>>.
- [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>>.
- [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>>.
- [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>>.

- [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>>.
- [RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <<https://www.rfc-editor.org/info/rfc8664>>.
- [RFC8697] Minei, I., Crabbe, E., Sivabalan, S., Ananthkrishnan, H., Dhody, D., and Y. Tanaka, "Path Computation Element Communication Protocol (PCEP) Extensions for Establishing Relationships between Sets of Label Switched Paths (LSPs)", RFC 8697, DOI 10.17487/RFC8697, January 2020, <<https://www.rfc-editor.org/info/rfc8697>>.
- [RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.
- [RFC9086] Previdi, S., Talaulikar, K., Ed., Filsfils, C., Patel, K., Ray, S., and J. Dong, "Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing BGP Egress Peer Engineering", RFC 9086, DOI 10.17487/RFC9086, August 2021, <<https://www.rfc-editor.org/info/rfc9086>>.
- [RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <<https://www.rfc-editor.org/info/rfc9256>>.
- [RFC9514] Dawra, G., Filsfils, C., Talaulikar, K., Ed., Chen, M., Bernier, D., and B. Decraene, "Border Gateway Protocol - Link State (BGP-LS) Extensions for Segment Routing over IPv6 (SRv6)", RFC 9514, DOI 10.17487/RFC9514, December 2023, <<https://www.rfc-editor.org/info/rfc9514>>.
- [RFC9552] Talaulikar, K., Ed., "Distribution of Link-State and Traffic Engineering Information Using BGP", RFC 9552, DOI 10.17487/RFC9552, December 2023, <<https://www.rfc-editor.org/info/rfc9552>>.
- [RFC9843] Hegde, S., Britto, W., Shetty, R., Decraene, B., Psenak, P., and T. Li, "IGP Flexible Algorithms: Bandwidth, Delay, Metrics, and Constraints", RFC 9843, DOI 10.17487/RFC9843, September 2025, <<https://www.rfc-editor.org/info/rfc9843>>.

10.2. Informative References

- [BGP-LS-TE-PATH] Previdi, S., Talaulikar, K., Ed., Dong, J., Gredler, H., and J. Tantsura, "Advertisement of Traffic Engineering Paths using BGP Link-State", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-ls-te-path-02, 11 November 2024, <<https://datatracker.ietf.org/doc/html/draft-ietf-idr-bgp-ls-te-path-02>>.

-
- [IEEE754] IEEE, "IEEE Standard for Floating-Point Arithmetic", IEEE Std 754-2019, DOI 10.1109/ieeestd.2019.8766229, 22 July 2019, <<https://ieeexplore.ieee.org/document/8766229>>.
- [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>>.
- [RFC4202] Kompella, K., Ed. and Y. Rekhter, Ed., "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4202, DOI 10.17487/RFC4202, October 2005, <<https://www.rfc-editor.org/info/rfc4202>>.
- [RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <<https://www.rfc-editor.org/info/rfc5065>>.
- [RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <<https://www.rfc-editor.org/info/rfc6952>>.
- [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>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.
- [RFC9830] Previdi, S., Filsfils, C., Talaulikar, K., Ed., Mattes, P., and D. Jain, "Advertising Segment Routing Policies in BGP", RFC 9830, DOI 10.17487/RFC9830, September 2025, <<https://www.rfc-editor.org/info/rfc9830>>.
- [RFC9831] Talaulikar, K., Ed., Filsfils, C., Previdi, S., Mattes, P., and D. Jain, "Segment Type Extensions for BGP Segment Routing (SR) Policy", RFC 9831, DOI 10.17487/RFC9831, September 2025, <<https://www.rfc-editor.org/info/rfc9831>>.

Acknowledgements

The authors would like to thank Dhruv Dhody, Mohammed Abdul Aziz Khalid, Lou Berger, Acee Lindem, Siva Sivabalan, Arjun Sreekantiah, Dhanendra Jain, Francois Clad, Zafar Ali, Stephane Litkowski, Aravind Babu Mahendra Babu, Geetanjali Bhalla, Ahmed Bashandy, Mike Koldychev, Samuel Sidor, Alex Tokar, Rajesh Melarcode Venkateswaran, Lin Changwang, Liu Yao, Joel Halpern, and Ned Smith for their reviews and valuable comments. The authors would also like to thank Susan Hares for her shepherd review and helpful comments to improve this document. The authors would like to thank John Scudder for his AD review and helpful suggestions to improve this document.

Contributors

The following people have contributed substantially to the content of this document and should be considered coauthors:

Clarence Filsfil

Cisco Systems

Email: cfilsfil@cisco.com

Mach(Guoyi) Chen

Huawei Technologies

Email: mach.chen@huawei.com

Authors' Addresses

Stefano Previdi

Individual

Email: stefano@previdi.net

Ketan Talaulikar (EDITOR)

Cisco Systems

India

Email: ketant.ietf@gmail.com

Jie Dong

Huawei Technologies

Huawei Campus, No. 156 Beiqing Rd.

Beijing

100095

China

Email: jie.dong@huawei.com

Hannes Gredler

RtBrick Inc.

Email: hannes@rtbrick.com**Jeff Tantsura**

Nvidia

Email: jefftant.ietf@gmail.com