BGP Request for Advertising Candidate Path of Segment Routing TE Policies
draft-li-ldr-bgp-request-cp-sr-te-policy-00

Abstract

An SR Policy is a set of candidate paths. The headend of an SR Policy may learn multiple candidate paths for an SR Policy via a number of different mechanisms, e.g., CLI, NetConf, PCEP, or BGP. BGP distribute candidate paths has been defined in [I-D.ietf-idr-segment-routing-te-policy]. This document defines an extension of BGP to request BGP speaker(controller) advertise the candidate paths. The goal is to unify the protocol when the candidate path of SR Policy provision is via BGP to reduce the network complexity and potential bugs cause by different protocol interactions.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 9, 2020.
1. Introduction

An SR Policy defined in [I-D.ietf-spring-segment-routing-policy] is a set of candidate paths. The headend of an SR Policy may be informed by various means including: Configuration, PCEP[RFC8281] or BGP[I-D.ietf-idr-segment-routing-te-policy]. All these mechanisms are PCE/Controller initiated, but in some situations headend maybe want pull one or a set of candidate paths from PCE/Controller rather than get all information passively. Actually PCEP can use request and reply messages defined in [RFC5440]to match this requirement but the mechanism is not clear when controller advertise candidate path via BGP protocol.
This document wants to define a way to request controller (BGP speaker) advertise candidate path via BGP update message to let BGP protocol can also get the similar mechanism with request and reply like PCEP.

2. Terminology

RP: Request Parameters
LSPA: LSP Attributes
SVEC: Synchronization VECtor
IRO: Include Route Object
ERO: Explicit Route Object
MSD: Base MPLS Imposition Maximum SID Depth, as defined in [RFC8491]
NAI: Node or Adjacency Identifier
PCC: Path Computation Client
PCE: Path Computation Element
PCEP: Path Computation Element Communication Protocol
SID: Segment Identifier
SR: Segment Routing
SR-TE: Segment Routing Traffic Engineering

3. Overview of BGP UPDATE Message for Request SR TE policy candidate path advertising

[I-D.ietf-idr-segment-routing-te-policy] defined the headend get candidate paths by BGP from controller (BGP speaker). In some situations headend just want get these candidate paths when some special event occur (e.g. receive a customer route (VPN route) with special color or special BGP attribute). This document define the mechanism when this special situation occurred how the headend request controller to advertise the matched SR policy with the candidate paths.

Step1. The headend decide to get a new candidate path from controller based on some trigger event. This trigger mechanism is out of scope of this document.
Step 2. The headend creates a new BGP request UPDATE message (defined in this document) with constraints of TE, such as affinity, metric, SRLG, and so on. This special request UPDATE message SHOULD NOT be used for BGP best path selection.

Step 3. The controller will calculate one or a set of segment list based on the payload of BGP request message from headend. How to calculate the path is out of scope of this document.

Step 4. The controller advertise SR Policy with candidate path to headend. How to advertise the policy is out of scope of this document and defined in [I-D.ietf-idr-segment-routing-te-policy]

4. BGP request UPDATE Message Encoding

4.1. Extention of SR Policy Message Encoding

Defined the SR Policy NLRI as follows:

```
+------------------+
|   NLRI Length    | 1 octet
+------------------+
|  Distinguisher    | 4 octets
+------------------+
|  Policy Color    | 4 octets
+------------------+
|    Endpoint      | 4 or 16 octets
+------------------+
```

where:

- NLRI Length: 1 octet of length expressed in bits as defined in [RFC4760].

- Distinguisher: 4-octet value uniquely identifying the policy in the context of <color, endpoint> tuple. The distinguisher has no semantic value and is solely used by the SR Policy originator to make unique (from an NLRI perspective) multiple occurrences of the same SR Policy.

- Policy Color: 4-octet value identifying (with the endpoint) the policy. The color is used to match the color of the destination prefixes to steer traffic into the SR Policy [I-D.ietf-spring-segment-routing-policy]

- Endpoint: identifies the endpoint of a policy. The Endpoint may represent a single node or a set of nodes (e.g., an anycast)
The Endpoint is an IPv4 (4-octet) address or an IPv6 (16-octet) address according to the AFI of the NLRI.

NLRI Length, Policy Color, Endpoint field remains unchanged, while the Distinguisher field will be set to FF:FF:FF:FF to signal the request to controller.

4.2. New SR Policy and Tunnel Encapsulation Attribute

The content of the SR Policy is encoded in the Tunnel Encapsulation Attribute originally defined in [I-D.ietf-idr-tunnel-encaps] using a new Tunnel-Type TLV (codepoint is 15, assigned by IANA. The SR Policy Encoding structure is as follows:

SR Policy SAFI NLRI: <Distinguisher, Policy-Color, Endpoint>
Attributes:
  Tunnel Encaps Attribute (23)
    Tunnel Type: SR Policy
    <Sub-TLVs>

Preference, Binding SID, Priority, Policy Name, ENLP, Segment-List, Weight and Segment sub-TLVs are all defined in [I-D.ietf-idr-segment-routing-te-policy] for SR Policy advertise to headend.

Additional 6 new Sub-TLVs are defined in this document section 3.3 for request mechanism.

1. LSPA Sub-TLV
2. SVEC Sub-TLV
3. Metric Sub-TLV
4. Include Route Sub-TLV
5. Load-Balancing

4.3. New SR Policy Sub-TLVs

4.3.1. LSPA Sub-TLV

The LSPA(LSP Attributes) Sub-TLV carries the same content as LSPA Object defined in [RFC5440] and [RFC3209].

The LSPA Sub-TLV is optional and specifies various TE LSP attributes to be taken for path computation. Most of the fields of the LSPA Sub-TLV are identical to the fields of the SESSION-ATTRIBUTE object.
(C-Type = 7) defined in [RFC3209]. See Section 4.7.4 of [RFC3209] for a detailed description of the use of resource affinities.

The LSPA sub-TLV has following format:

```
+--------------------------------------------------+
|    Type    |    Length    |     Flags    |L|   Reserved   |
+--------------------------------------------------+
```

where:

- **Type**: TBD1
- **Length**: specifies the length of the value field not including Type and Length fields.
- **Flags (8 bits)**
  - **L flag**: Corresponds to the "Local Protection Desired" bit of the SESSION-ATTRIBUTE Object. When set, this means that the computed path must include links protected with Fast Reroute as defined in [RFC4090].
  - **Reserved (8 bits)**: This field MUST be set to zero on transmission and MUST be ignored on receipt.
  - **Unassigned flags**: MUST be set to zero on transmission and MUST be ignored on receipt.
- **Optional TLVs**: may be defined in the future to carry additional TE LSP attributes such as those defined in [RFC5420]
- **Exclude-any, Include-any, Include-all sub-TLV’s format is the same as subobjects defined in [RFC3209]
4.3.2. SVEC Sub-TLV

The SVEC (Synchronization VECtor) Sub-TLV carries the same content as SVEC Object defined in [RFC5440].

The SVEC Sub-TLV allows headend to request the synchronization of a set of dependent or independent segment list computation requests. It’s optional be carried.

The SVEC sub-TLV has following format:

```
+---------------+---------------+---------------+---------------+
|     Type      |     Length    |        Flags            |S|N|L|
+---------------+---------------+---------------+---------------+
```

where:

- **Type**: TBD2
- **Length**: the total length (not including the Type and Length fields) of the sub-TLVs encoded
- **Flags** (24 bits): Defines the potential dependency between a set of segment lists in one candidate path.

- L (Link diverse) bit: when set, this indicates that the computed segment lists of the same candidate path MUST NOT have any link in common.
- N (Node diverse) bit: when set, this indicates that the computed segment lists of the same candidate path MUST NOT have any node in common.
- S (SRLG diverse) bit: when set, this indicates that the computed segment lists of the same candidate path MUST NOT share any SRLG (Shared Risk Link Group).

In case of a set of segment lists synchronized independent path computation, the bits L, N, and S are cleared.

Unassigned flags MUST be set to zero on transmission and MUST be ignored on receipt.
4.3.3. Metric Sub-TLV

The Metric Sub-TLV carries the same content as Metric Object defined in [RFC5440] and [I-D.ietf-pce-segment-routing]. The Metric sub-TLV has following format:

<table>
<thead>
<tr>
<th>Type</th>
<th>Length</th>
<th>Flags</th>
<th>B</th>
<th>C</th>
<th>T</th>
</tr>
</thead>
<tbody>
<tr>
<td>metric-value (4 octets)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Type**: TBD3
- **Length**: specifies the length of the value field not including Type and Length fields.
- **Flags (8 bits)**: Two flags are currently defined:
  - **B** (Bound – 1 bit): When set in a BGP UPDATE message, the metric-value indicates a bound (a maximum) for the path metric that must not be exceeded for the headend to consider the computed path as acceptable. The path metric must be less than or equal to the value specified in the metric-value field. When the B flag is cleared, the metric-value field is not used to reflect a bound constraint.
  - **C** (Computed Metric – 1 bit): When set in a BGP UPDATE message, this indicates that the controller MUST provide the computed path metric value (should a path satisfying the constraints be found) in the advertise message for the corresponding metric.
  - Unassigned flags MUST be set to zero on transmission and MUST be ignored on receipt.
- **T** (Type – 8 bits): Specifies the metric type
  - Four values are currently defined:
    - **T=1**: IGP metric
    - **T=2**: TE metric
    - **T=3**: Hop Counts
    - **T=11**: Maximum SID Depth of the requested path
o Metric-value (32 bits): metric value encoded in 32 bits in IEEE floating point format (see [IEEE.754.1985]).

4.3.4. Include Route Sub-TLV

The Include Route Sub-TLV is optional and can be used to specify that the computed candidate path MUST traverse a set of specified network elements. The Include Route Sub-TLV carries the same content as IRO Object defined in [RFC5440], [RFC3209] and SR-ERO defined in [I-D.ietf-pce-segment-routing]

The Include Route Sub-TLV has following format:

```
+-------+-------+-------+-------+
| Type  | Length | NT    | Flags |
+-------+-------+-------+-------+
```

Where:

- **Type**: TBD4
- **Length**: specifies the length of the value field not including Type and Length fields.
- **NAI Type (NT)**: Indicates the type and format of the NAI contained in the Sub-TLV body, if any is present then the NT field has no meaning and MUST be ignored by the receiver. This document describes the following NT values:
  * NT=0 The NAI is absent.
  * NT=1 The NAI is an IPv4 node ID.
  * NT=2 The NAI is an IPv6 node ID.
  * NT=3 The NAI is an IPv4 adjacency.
  * NT=4 The NAI is an IPv6 adjacency with global IPv6 addresses.
  * NT=5 The NAI is an unnumbered adjacency with IPv4 node IDs.
* NT=6 The NAI is an IPv6 adjacency with link-local IPv6 addresses.

- SID and NAI are the same as SR-ERO defined in
  [I-D.ietf-pce-segment-routing]

4.3.5. Load-Balancing Sub-TLV

The Load-Balancing Sub-TLV defined how many segment list should be included in one candidate path. The Load-Balancing sub-TLV has 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Length    |     Flag      |   Max-Slist   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Option TLV                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where:

- Type: TBD5
- Length: specifies the length of the value field not including Type and Length fields.
- Flags (8 bits): No flag is currently defined. The Flags field MUST be set to zero on transmission and MUST be ignored on receipt.
- Max-Slist (8 bits): maximum number of S-Lists in the candidate path.
- Option TLV: No Option TLV currently defined. If bandwidth can be reserved in SR-Policy candidate path or different load-balancing principle between segment lists for different weight here can define additional TLVs.

5. IANA

This document defines new Sub-TLVs in following existing registries:
<table>
<thead>
<tr>
<th>Type</th>
<th>Sub-TLV</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD1</td>
<td>LSA-Sub-TLV</td>
<td>This document</td>
</tr>
<tr>
<td>TBD2</td>
<td>SVEC Sub-TLV</td>
<td>This document</td>
</tr>
<tr>
<td>TBD3</td>
<td>Metric Sub-TLV</td>
<td>This document</td>
</tr>
<tr>
<td>TBD4</td>
<td>Include Route Sub-TLV</td>
<td>This document</td>
</tr>
<tr>
<td>TBD5</td>
<td>Load-Balancing</td>
<td>This document</td>
</tr>
</tbody>
</table>

6. Contributors

TBD

7. Acknowledgments

TBD

8. References

[I-D.ietf-idr-segment-routing-te-policy]

[I-D.ietf-idr-tunnel-encaps]

[I-D.ietf-pce-segment-routing]

[I-D.ietf-spring-segment-routing-policy]
Authors’ Addresses

Zhenbin Li
Huawei
156 Beiqing Road
Beijing, 100095
P.R. China

Email: lizhenbin@huawei.com