Abstract

This document proposes a set of constraints for PCEP to configure PCE to use specific virtual network topology or application attributes during path computation. A simple COLOR parameter is also introduced to simplify network operations.

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 May 6, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of
1. Introduction

[RFC5440] describes the Path Computation Element Protocol (PCEP) which is used between a Path Computation Element (PCE) and a Path Computation Client (PCC) (or other PCE) to enable computation of Multi-protocol Label Switching (MPLS) for Traffic Engineering Label Switched Path (TE LSP). PCEP Extensions for the Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to enable active control of MPLS-TE and Generalized MPLS (GMPLS) tunnels. [RFC8281] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC, thus allowing for dynamic centralized control of a network.

As depicted in [RFC4655], a PCE MUST be able to compute the path of a TE LSP by operating on the TED and considering bandwidth and other constraints applicable to the TE LSP service request. The constraint parameters are provided such as metric, bandwidth, delay, affinity, etc. However these parameters can’t meet the virtual network service requirements. A PCE always perform path computation based on the network topology information collected through BGP-LS [RFC7752]. BGP-LS can get multiple link-state data from multiple IGP instance, or multiple virtual topologies from a single IGP instance. It is necessary to restrict the PCE to a small topology scope during path computation for some special purpose. BGP-LS can also get application specific TE attributes for a link, it is also necessary...
to restrict PCE to use TE attributes of specific application during path computation.

This document will extend PCEP to support some new constraint parameters during path computation, e.g., IGP instance, virtual network, specific application, as well as a simple COLOR parameter.

2. Conventions used in this document

2.1. Terminology

The terminology is defined as [RFC5440] and [RFC7752].

2.2. Requirements Language

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

3. PCEP Extensions for Constraints

3.1. Source Protocol Object

The Source Protocol object is optional and can be used for several purposes.

In a PCReq message, a PCC MAY insert one Source Protocol object to indicate the source protocol that MUST be considered by the PCE. The PCE will perform path computation based on the sub-topology identified by the specific source protocol. The absence of the Source Protocol object MUST be interpreted by the PCE as a path computation request for which no constraints need be applied to any of the source protocols.

In a PCRep/PCInit/PCUpd message, the Source Protocol object MAY be inserted so as to provide the source protocol information for the computed path.

Only one Source Protocol Object could be inserted in the above messages, otherwise the first one MUST be considered and others MUST be ignored.

Source Protocol Object-Class is TBA.

Source Protocol Object-Type is 1.
The format of the Source Protocol object is shown as Figure 1:

```
<table>
<thead>
<tr>
<th>Protocol-ID</th>
<th>Reserved</th>
<th>Identifier (64 bits)</th>
</tr>
</thead>
</table>
+-------------+----------+---------------------|
```

Figure 1: Source Protocol Object

The Source Protocol object body has a fixed length of 12 bytes.


Reserved (24 bits): This field MUST be set to zero on transmission and MUST be ignored on receipt.

Identifier (64 bits): defined in [RFC7752] section 3.2.

3.2. Multi-topology Object

The Multi-topology object is optional and can be used for several purposes.

In a PCReq message, a PCC MAY insert one Multi-topology object to indicate the sub-topology of an IGP instance that MUST be considered by the PCE. The PCE will perform path computation based on the sub-topology identified by the specific Multi-Topology ID within a source protocol. The absence of the Multi-topology object MUST be interpreted by the PCE as a path computation request for which no constraints need be applied to any of the multi-topologies.

In a PCRep/PCInit/PCUpd message, the Multi-topology object MAY be inserted so as to provide the Multi-topology information for the computed path.

Only one Multi-topology Object could be inserted in the above messages, otherwise the first one MUST be considered and others MUST be ignored. It MUST be inserted with a Source Protocol Object, if not it MUST be ignored.

Multi-topology Object-Class is TBA.
Multi-topology Object-Type is 1.

The format of the Multi-topology object is shown as Figure 2:

```
  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
+-----------------------------+-----------------------------+
| R R R R | Multi-Topology ID | Reserved |
+-----------------------------+-----------------------------+
```

Figure 2: Multi-topology Object

The Multi-topology object body has a fixed length of 4 bytes.

Multi-Topology ID (16 bits): Semantics of the IS-IS MT-ID are defined in Section 7.2 of [RFC5120]. Semantics of the OSPF MT-ID are defined in Section 3.7 of [RFC4915]. If the value is derived from OSPF, then the upper 9 bits MUST be set to 0. Bits R are reserved and SHOULD be set to 0 when originated and ignored on receipt.

Reserved (16 bits): This field MUST be set to zero on transmission and MUST be ignored on receipt.

### 3.3. The AII Object

The AII object is optional and can be used for several purposes.

In a PCReq message, a PCC MAY insert one AII object to indicate the global virtual network that MUST be considered by the PCE. The PCE will perform path computation based on the intra or inter-domain sub-topology identified by the specific AII, which is independent of routing protocols such as IGP/BGP. The absence of the AII object MUST be interpreted by the PCE as a path computation request for which no constraints need be applied to any of the virtual network, i.e., a default AII (0) will be applied.

In a PCRep/PCInit/PCUpd message, the AII object MAY be inserted so as to provide the network slicing information for the computed path.

Only one AII Object could be inserted in the above messages, otherwise the first one MUST be considered and others MUST be ignored.

AII Object-Class is TBA.

AII Object-Type is 1.
The format of the AII object is shown as Figure 3:

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

Figure 3: AII Object

The AII object body has a fixed length of 4 bytes.

AII (32 bits): Administrative Instance Identifier defined in [I-D.peng-lsr-network-slicing].

3.4. Application Specific Object

The Application Specific object is optional and can be used for several purposes.

In a PCReq message, a PCC MAY insert one Application Specific object to indicate the application that MUST be considered by the PCE. The PCE will perform path computation using the specific application attributes. The absence of the Application Specific object MUST be interpreted by the PCE as a path computation request for which no constraints need be applied to any of the Application Specific attributes.

In a PCRep/PCInit/PCUpd message, the Application Specific object MAY be inserted so as to provide the Application Specific information for the computed path.

Only one Application Specific Object could be inserted in the above messages, otherwise the first one MUST be considered and others MUST be ignored.

Application Specific Object-Class is TBA.

Application Specific Object-Type is 1.

The format of the Application Specific object is shown as Figure 4:
The Application Specific object body has a fixed length of 8 bytes.

Standard Application ID : Represents a bit-position value for a single STANDARD application that is defined in the IANA "IGP Parameters" registries under the "Link Attribute Applications" registry [I-D.ietf-isis-te-app].

User Defined Application ID : Represents a single user defined application that is implementation specific.

3.5. The Color Object

The Color object is optional and can be used for several purposes.

In a PCReq message, a PCC MAY insert one Color object to indicate the traffic engineering purpose that is recognized by the both PCE and PCC with no conflict meaning. The PCE will perform path computation based on the color template defined in local and extract the detailed constraints from the color template. Note the same color template is also defined in PCC side. At this time, any other traditional constraints (i.e, metric, bandwidth, delay, etc) that is directly contained in the message MUST be ignored. The absence of the Color object MUST be interpreted by the PCE as a path computation request for which traditional constraints that are contained in message need be applied.

In a PCRep/PCInit/PCUpd message, the Color object MAY be inserted so as to provide the TE purpose information for the computed path, the PCC recognize the color value that match a local color-template.

Only one Color Object could be inserted in the above messages, otherwise the first one MUST be considered and others MUST be ignored.

Color Object-Class is TBA.

Color Object-Type is 1.
The format of the Color object is shown as Figure 5:

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

Figure 5: Color Object

The Color object body has a fixed length of 4 bytes.

Color (32 bits): Represent a TE purpose, 0 is invalid value. It is consistent with the meaning of Color Extended Community that is defined in [I-D.ietf-idr-tunnel-encaps], and color of SR policy that is also defined in [I-D.ietf-spring-segment-routing-policy].

Note that Color Object defined in this document is used to represent a TE purpose, it can be suitable for any TE instance such as RSVP-TE, SR-TE, SR-policy. [I-D.barth-pce-segment-routing-policy-cp] has already been using SR policy KEY (that also includes a color information) as an association group KEY to associate many candidate paths, however it is only for association purpose but not constraint purpose for path computation.

A color template can be defined to use any constraints such as traditional metric, bandwidth, delay, affinity parameters, but also any sub-topology parameters above defined in this document. Both PCE and PCC MUST have the same understanding for a same color value.

4. Security Considerations

TBA

5. Acknowledgements

TBA

6. IANA Considerations

IANA is requested to make allocations from the registry, as follows:
### Table 1

<table>
<thead>
<tr>
<th>Value</th>
<th>Object</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBA1</td>
<td>Source Protocol Object</td>
<td>[this document]</td>
</tr>
<tr>
<td>TBA2</td>
<td>Multi-topology Object</td>
<td>[this document]</td>
</tr>
<tr>
<td>TBA3</td>
<td>AII Object</td>
<td>[this document]</td>
</tr>
<tr>
<td>TBA4</td>
<td>Application Specific Object</td>
<td>[this document]</td>
</tr>
<tr>
<td>TBA5</td>
<td>Color Object</td>
<td>[this document]</td>
</tr>
</tbody>
</table>

#### 7. Normative References

[I-D.barth-pce-segment-routing-policy-cp]

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

[I-D.ietf-isis-te-app]

[I-D.ietf-spring-segment-routing-policy]

[I-D.peng-lsr-network-slicing]


Authors’ Addresses
Shaofu Peng
ZTE Corporation
No.50 Software Avenue
Nanjing, Jiangsu 210012
China

Email: peng.shaofu@zte.com.cn

Quan Xiong
ZTE Corporation
No.6 Huashi Park Rd
Wuhan, Hubei 430223
China

Email: xiong.quan@zte.com.cn

Fengwei Qin
China Mobile
Beijing
China

Email: qinfengwei@chinamobile.com