PCE
Internet-Draft
Intended status: Standards Track
Expires: May 24, 2018

S. Sivabalan
C. Filsfils
Cisco Systems, Inc.
J. Tantsura
Individual
W. Henderickx
Nokia
J. Hardwick
Metaswitch Networks

November 20, 2017

PCEP Extensions for Segment Routing
draft-ietf-pce-segment-routing-11

Abstract

Segment Routing (SR) enables any head-end node to select any path without relying on a hop-by-hop signaling technique (e.g., LDP or RSVP-TE). It depends only on "segments" that are advertised by Link-State Interior Gateway Protocols (IGPs). A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE). This document specifies extensions to the Path Computation Element Protocol (PCEP) that allow a stateful PCE to compute and initiate Traffic Engineering (TE) paths, as well as a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks.

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 [RFC2119].

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 24, 2018.

Copyright Notice

Table of Contents

1. Introduction

SR technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols (IS-IS or OSPF). [I-D.ietf-spring-segment-routing] provides an introduction to SR architecture. The corresponding IS-IS and OSPF extensions are specified in [I-D.ietf-isis-segment-routing-extensions] and [I-D.ietf-ospf-segment-routing-extensions], respectively. SR architecture defines a "segment" as a piece of information advertised by a link-state routing protocols, e.g. an IGP prefix or an IGP adjacency. Several types of segments are defined. A Node segment represents an ECMP-aware shortest-path computed by IGP to a specific node, and is always global within SR/IGP domain. An Adjacency Segment represents a unidirectional adjacency. An Adjacency Segment is local to the node which advertises it. Both Node segments and Adjacency segments can be used for SR Traffic Engineering (SR-TE).

The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). This document is relevant to the MPLS forwarding plane only. In this document, "Node-SID" and "Adjacency-SID" denote Node Segment Identifier and Adjacency Segment Identifier respectively.

A Segment Routed path (SR path) can be derived from an IGP Shortest Path Tree (SPT). SR-TE paths may not follow an IGP SPT. Such paths may be chosen by a suitable network planning tool and provisioned on the ingress node of the SR-TE path.

[RFC5440] describes the Path Computation Element Protocol (PCEP) for communication between a Path Computation Client (PCC) and a Path Computation Element (PCE) or between a pair of PCEs. A PCE, or a PCC operating as a PCE (in hierarchical PCE environment), computes paths for MPLS Traffic Engineering LSPs (MPLS-TE LSPs) based on various constraints and optimization criteria. [RFC8231] specifies extensions to PCEP that allow a stateful PCE to compute and recommend network paths in compliance with [RFC4657] and defines objects and

TLVs for MPLS-TE LSPs. Stateful PCEP extensions provide synchronization of LSP state between a PCC and a PCE or between a pair of PCEs, delegation of LSP control, reporting of LSP state from a PCC to a PCE, controlling the setup and path routing of an LSP from a PCE to a PCC. Stateful PCEP extensions are intended for an operational model in which LSPs are configured on the PCC, and control over them is delegated to the PCE.

A mechanism to dynamically initiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE is specified in [I-D.ietf-pce-pce-initiated-lsp]. This mechanism is useful in Software Defined Networking (SDN) applications, such as on-demand engineering, or bandwidth calendaring.

It is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [I-D.ietf-pce-pce-initiated-lsp] using the SR specific PCEP extensions specified in this document. Additionally, using procedures described in this document, a PCC can request an SR path from either stateful or a stateless PCE. This specification relies on the procedures specified in [I-D.ietf-pce-lsp-setup-type].

2. Terminology

The following terminologies are used in this document:

ERO: Explicit Route Object

IGP: Interior Gateway Protocol

IS-IS: Intermediate System to Intermediate System

LSR: Label Switching Router

MSD: Maximum SID Depth

NAI: Node or Adjacency Identifier

OSPF: Open Shortest Path First

PCC: Path Computation Client

PCE: Path Computation Element

PCEP: Path Computation Element Protocol

RRO: Record Route Object

SID: Segment Identifier

SR: Segment Routing

SR-TE: Segment Routed Traffic Engineering

TED: Traffic Engineering Database

3. Overview of PCEP Operation in SR Networks

In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (or MPLS labels in the context of this document). The header has all necessary information to guide the packets from the ingress node to the egress node of the path, and hence there is no need for any signaling protocol.

In a PCEP session, LSP information is carried in the Explicit Route Object (ERO), which consists of a sequence of subobjects. Various types of ERO subobjects have been specified in [RFC3209], [RFC3473], and [RFC3477]. In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (or MPLS labels in the context of this document). SR-TE LSPs computed by a PCE can be represented in one of the following forms:

- An ordered set of IP address(es) representing network nodes/links: In this case, the PCC needs to convert the IP address(es) into the corresponding MPLS labels by consulting its Traffic Engineering Database (TED).

- An ordered set of SID(s).

- An ordered set of both MPLS label(s) and IP address(es): In this case, the PCC needs to convert the IP address(es) into the corresponding SID(s) by consulting its TED.

This document defines a new ERO subobject denoted by "SR-ERO subobject" capable of carrying a SID as well as the identity of the node/adjacency represented by the SID. SR-capable PCEP speakers should be able to generate and/or process such ERO subobject. An ERO containing SR-ERO subobjects can be included in the PCEP Path Computation Reply (PCRep) message defined in [RFC5440], the PCEP LSP Initiate Request message (PCInitiate) defined in [I-D.ietf-pce-pce-initiated-lsp], as well as in the PCEP LSP Update Request (PCUpd) and PCEP LSP State Report (PCRpt) messages defined in [RFC8231].

When a PCEP session between a PCC and a PCE is established, both PCEP speakers exchange their capabilites to indicate their ability to support SR-specific functionality.

An PCE can update an LSP that is initially established via RSVP-TE signaling to use an SR-TE path, by sending a PCUpd to the PCC that delegated the LSP to it ([RFC8231]). Similarly, an LSP initially created with an SR-TE path can be updated to use RSVP-TE signaling, if necessary. This capability is useful when a network is migrated from RSVP-TE to SR-TE technology.

A PCC MAY include an RRO object containing the recorded LSP in PCReq and PCRpt messages as specified in [RFC5440] and [RFC8231], respectively. This document defines a new RRO subobject for SR networks. The methods used by a PCC to record the SR-TE LSP are outside the scope of this document.

In summary, this document:

o Defines a new ERO subobject, a new RRO subobject and new PCEP error codes.

o Specifies how two PCEP speakers can establish a PCEP session that can carry information about SR-TE paths.

o Specifies processing rules for the ERO subobject.

o Defines a new path setup type to be used in the PATH_SETUP_TYPE and PATH_SETUP_TYPE_CAPABILITY TLVs ([I-D.ietf-pce-lsp-setup-type]).

o Defines a new sub-TLV for the PATH_SETUP_TYPE_CAPABILITY TLV.

The extensions specified in this document complement the existing PCEP specifications to support SR-TE paths. As such, the PCEP messages (e.g., Path Computation Request, Path Computation Reply, Path Computation Report, Path Computation Update, Path Computation Initiate, etc.,) MUST be formatted according to [RFC5440], [RFC8231], [I-D.ietf-pce-pce-initiated-lsp], and any other applicable PCEP specifications.

4. SR-Specific PCEP Message Extensions

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable length body made up of mandatory and/or optional objects. This document does not require any changes in the format of the PCReq and PCRep messages specified in [RFC5440],

PCInitiate message specified in [I-D.ietf-pce-pce-initiated-lsp], and PCRpt and PCUpd messages specified in [RFC8231].

5. Object Formats

5.1. The OPEN Object

5.1.1. The SR PCE Capability sub-TLV

This document defines a new Path Setup Type (PST) for SR, as follows:

o PST = 1: Path is setup using Segment Routing Traffic Engineering.

A PCEP speaker SHOULD indicate its support of the function described in this document by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN object with this new PST included in the PST list.

This document also defines the SR-PCE-CAPABILITY sub-TLV. PCEP speakers use this sub-TLV to exchange information about their SR capability. If a PCEP speaker includes PST=1 in the PST List of the PATH-SETUP-TYPE-CAPABILITY TLV then it MUST also include the SR-PCE-CAPABILITY sub-TLV inside the PATH-SETUP-TYPE-CAPABILITY TLV.

The format of the SR-PCE-CAPABILITY sub-TLV is shown in the following figure:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |            Type=26            |            Length=4           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |         Reserved              |     Flags   |L|      MSD      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: SR-PCE-CAPABILITY sub-TLV format

The code point for the TLV type is 26. The TLV length is 4 octets.

The 32-bit value is formatted as follows. The "Maximum SID Depth" (1 octet) field (MSD) specifies the maximum number of SIDs (MPLS label stack depth in the context of this document) that a PCC is capable of imposing on a packet. The "Reserved" (2 octets) field is unused, and MUST be set to zero on transmission and ignored on reception. The "Flags" field is 1 octect long, and this document defines the following flag:

o L-flag: A PCC sets this flag to 1 to indicate that it does not impose any limit on the MSD.

5.1.2. Exchanging the SR PCE Capability

A PCC indicates that it is capable of supporting the head-end functions for SR-TE LSP by including the SR-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCE. A PCE indicates that it is capable of computing SR-TE paths by including the SR-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCC.

If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a PST list containing PST=1, but the SR-PCE-CAPABILITY sub-TLV is absent, then the PCEP speaker MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value TBD1 (to be assigned by IANA) (Missing PCE-SR-CAPABILITY sub-TLV) and MUST then close the PCEP session. If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a SR-PCE-CAPABILITY sub-TLV, but the PST list does not contain PST=1, then the PCEP speaker MUST ignore the SR-PCE-CAPABILITY sub-TLV.

The number of SIDs that can be imposed on a packet depends on the PCC's data plane's capability. If a PCC sets the L flag to 1 then the MSD is not used and MUST be set to zero. If a PCE receives an SR-PCE-CAPABILITY sub-TLV with the L flag set to 1 then it MUST ignore the MSD field and MUST assume that the sender can impose a SID stack of any depth. If a PCC sets the L flag to zero, then it sets the MSD field to the maximum number of SIDs that it can impose on a packet. If a PCE receives an SR-PCE-CAPABILITY sub-TLV with the L flag and MSD both set to zero then it MUST assume that the PCC is not capable of imposing a SID stack of any depth and hence is not SR-TE capable, unless it learns a non-zero MSD for the PCC through some other means.

Note that the MSD value exchanged via the SR-PCE-CAPABILITY sub-TLV indicates the SID/label imposition limit for the PCC node. However, if a PCE learns the MSD value of a PCC node via different means, e.g routing protocols, as specified in:
[I-D.ietf-isis-segment-routing-msd];
[I-D.ietf-ospf-segment-routing-msd];
[I-D.ietf-idr-bgp-ls-segment-routing-msd], then it ignores the MSD value in the SR-PCE-CAPABILITY sub-TLV. Furthermore, whenever a PCE learns the MSD for a link via different means, it MUST use that value for that link regardless of the MSD value exchanged in the SR-PCE-CAPABILITY sub-TLV.

Once an SR-capable PCEP session is established with a non-zero MSD value, the corresponding PCE MUST NOT send SR-TE paths with a number

of SIDs exceeding that MSD value. If a PCC needs to modify the MSD value, it MUST close the PCEP session and re-establish it with the new MSD value. If a PCEP session is established with a non-zero MSD value, and the PCC receives an SR-TE path containing more SIDs than specified in the MSD value, the PCC MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value 3 (Unsupported number of Segment ERO subobjects). If a PCEP session is established with an MSD value of zero, then the PCC MAY specify an MSD for each path computation request that it sends to the PCE, by including a "maximum SID depth" metric object on the request, as defined in Section 5.5.

The L flag and MSD value inside the SR-PCE-CAPABILITY sub-TLV are meaningful only in the Open message sent from a PCC to a PCE. As such, a PCE MUST set the L flag and MSD value to zero in an outbound message to a PCC. Similarly, a PCC MUST ignore any MSD value received from a PCE. If a PCE receives multiple SR-PCE-CAPABILITY sub-TLVs in an Open message, it processes only the first sub-TLV received.

5.2. The RP/SRP Object

In order to setup an SR-TE LSP using SR, RP or SRP object MUST include PATH-SETUP-TYPE TLV, specified in [I-D.ietf-pce-lsp-setup-type], with the PST set to 1 (path setup using SR-TE).

The LSP-IDENTIFIERS TLV MAY be present for the above PST type.

5.3. ERO Object

An SR-TE path consists of one or more SID(s) where each SID MAY be associated with the identifier that represents the node or adjacency corresponding to the SID. This identifier is referred to as the 'Node or Adjacency Identifier' (NAI). As described later, a NAI can be represented in various formats (e.g., IPv4 address, IPv6 address, etc). Furthermore, a NAI is used for troubleshooting purposes and, if necessary, to derive SID value as described below.

The ERO object specified in [RFC5440] is used to carry SR-TE path information. In order to carry SID and/or NAI, this document defines a new ERO subobject referred to as "SR-ERO subobject" whose format is specified in the following section. An ERO object carrying an SR-TE path consists of one or more ERO subobject(s), and MUST carry only SR-ERO subobject(s). Note that an SR-ERO subobject does not need to have both SID and NAI. However, at least one of them MUST be present.

When building the MPLS label stack from ERO, a PCC MUST assume that SR-ERO subobjects are organized as a last-in-first-out stack. The first subobject relative to the beginning of ERO contains the information about the topmost label. The last subobject contains information about the bottommost label.

5.3.1. SR-ERO Subobject

An SR-ERO subobject consists of a 32-bit header followed by the SID and the NAI associated with the SID. The SID is a 32-bit number. The size of the NAI depends on its respective type, as described in the following sections.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |  ST   |     Flags     |F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              SID                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        NAI (variable)                       //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: SR-ERO Subobject format

The fields in the SR-ERO Subobject are as follows:

The 'L' Flag indicates whether the subobject represents a loose-hop in the LSP [RFC3209]. If this flag is unset, a PCC MUST not overwrite the SID value present in the SR-ERO subobject. Otherwise, a PCC MAY expand or replace one or more SID value(s) in the received SR-ERO based on its local policy.

Type is the type of the SR-ERO subobject. This document defines the SR-ERO subobject type, and requests a new codepoint from IANA.

Length contains the total length of the subobject in octets, including the L, Type and Length fields. Length MUST be at least 8, and MUST be a multiple of 4. As mentioned earlier, an SR-ERO subobject MUST have at least SID or NAI. The length should take into consideration SID or NAI only if they are not null. The flags described below used to indicate whether SID or NAI field is null.

SID Type (ST) indicates the type of information associated with the SID contained in the object body. When ST value is 0, SID MUST NOT be null and NAI MUST be null. Other ST values are described later in this document.

Flags is used to carry any additional information pertaining to SID. Currently, the following flag bits are defined:

* M: When this bit is set, the SID value represents an MPLS label stack entry as specified in [RFC5462] where only the label value is specified by the PCE. Other fields (TC, S, and TTL) fields MUST be considered invalid, and PCC MUST set these fields according to its local policy and MPLS forwarding rules.

* C: When this bit as well as the M bit are set, then the SID value represents an MPLS label stack entry as specified in [RFC5462], where all the entry's fields (Label, TC, S, and TTL) are specified by the PCE. However, a PCC MAY choose to override TC, S, and TTL values according its local policy and MPLS forwarding rules.

* S: When this bit is set, the SID value in the subobject body is null. In this case, the PCC is responsible for choosing the SID value, e.g., by looking up its TED using the NAI which, in this case, MUST be present in the subobject.

* F: When this bit is set, the NAI value in the subobject body is null.

SID is the Segment Identifier.

NAI contains the NAI associated with the SID. Depending on the value of ST, the NAI can have different format as described in the following section.

5.3.2. NAI Associated with SID

This document defines the following NAIs:

'IPv4 Node ID' is specified as an IPv4 address. In this case, ST value is 1, and the Length is 8 or 12 depending on either SID or NAI or both are included in the subobject.

'IPv6 Node ID' is specified as an IPv6 address. In this case, ST and Length are 2, and Length is 8, 20, or 24 depending on either SID or NAI or both are included in the subobject.

'IPv4 Adjacency' is specified as a pair of IPv4 addresses. In this case, ST value is 3. The Length is 8, 12, or 16 depending on either SID or NAI or both are included in the subobject, and the format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Local IPv4 address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Remote IPv4 address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: NAI for IPv4 adjacency

'IPv6 Adjacency' is specified as a pair of IPv6 addresses. In this case, ST valie is 4. The Length is 8, 36 or 40 depending on whether SID or NAI or both included in the subobject,and the format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//               Local IPv6 address (16 bytes)                 //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//               Remote IPv6 address (16 bytes)                //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: NAI for IPv6 adjacency

'Unnumbered Adjacency with IPv4 NodeIDs' is specified as a pair of Node ID / Interface ID tuples. In this case, ST value is 5. The Length is 8, 20, or 24 depending on whether SID or NAI or both included in the subobject, and the format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Local Node-ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Local Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Remote Node-ID                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Remote Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: NAI for Unnumbered adjacency with IPv4 Node IDs

5.3.3. ERO Processing

A PCEP speaker that does not recognize the SR-ERO subobject in PCRep, PCInitiate, PCUpd or PCRpt messages MUST reject the entire PCEP message and MUST send a PCErr message with Error-Type=3 ("Unknown Object") and Error-Value=2 ("Unrecognized object Type") or Error-Type=4 ("Not supported object") and Error-Value=2 ("Not supported object Type"), defined in [RFC5440].

When the SID represents an MPLS label (i.e. the M bit is set), its value (20 most significant bits) MUST be larger than 15, unless it is special purpose label, such as an Entropy Label Indicator (ELI). If a PCEP speaker receives an invalid value, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error Value = 2 ("Bad label value"). If both M and C bits of an SR-ERO subobject are set, and if a PCEP speaker finds erroneous setting in one or more of TC, S, and TTL fields, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 4 ("Bad label format").

If a PCC receives a stack of SR-ERO subobjects, and the number of stack exceeds the maximum number of SIDs that the PCC can impose on the packet, it MAY send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 3 ("Unsupported number of Segment ERO subobjects").

When a PCEP speaker detects that all subobjects of ERO are not identical, and if it does not handle such ERO, it MUST send a PCErr

message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 5 ("Non-identical ERO subobjects").

If a PCEP speaker receives an SR-ERO subobject in which both SID and NAI are absent, it MUST consider the entire ERO object invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 6 ("Both SID and NAI are absent in ERO subobject").

When a PCEP speaker receives an SR-ERO subobject in which ST is 0, SID MUST be present and NAI MUST NOT be present(i.e., S-flag MUST be 0, F-flag MUST be 1, and the Length MUST be 8). Otherwise, it MUST consider the entire ERO object invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 11 ("Malformed object"). The PCEP speaker MAY include the malformed SR-ERO object in the PCErr message as well.

5.4. RRO Object

A PCC can record SR-TE LSP and report the LSP to a PCE via RRO. An RRO object contains one or more subobjects called "SR-RRO subobjects" whose format is shown below:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Length    |  ST   |     Flags     |F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              SID                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        NAI (variable)                       //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: SR-RRO Subobject format

The format of SR-RRO subobject is the same as that of SR-ERO subobject without L flag.

A PCC MUST assume that SR-RRO subobjects are organized such that the first subobject relative to the beginning of RRO contains the information about the topmost label, and the last subobject contains information about the bottommost label of the SR-TE LSP.

5.4.1. RRO Processing

Processing rules of SR-RRO subobject are identical to those of SR-ERO subobject.

If a PCEP speaker receives an SR-RRO subobject in which both SID and NAI are absent, it MUST consider the entire RRO object invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 7 ("Both SID and NAI are absent in RRO subobject").

If a PCE detects that all subobjects of RRO are not identical, and if it does not handle such RRO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 10 ("Non-identical RRO subobjects").

5.5. METRIC Object

If a PCEP session is established with an MSD value of zero, then the PCC MAY specify the MSD for an individual path computation request using the METRIC object defined in [RFC5440]. This document defines a new type for the METRIC object to be used for this purpose as follows:

- T = 11: Maximum SID Depth of the requested path.

The PCC sets the metric-value to the MSD for this path. The PCC MUST set the B (bound) bit to 1 in the METRIC object, which specifies that the SID depth for the computed path MUST NOT exceed the metric-value.

If a PCEP session is established with a non-zero MSD value, then the PCC MUST NOT send an MSD METRIC object. If the PCE receives a path computation request with an MSD METRIC object on a session with a non-zero MSD value then it MUST consider the request invalid and send a PCErr with Error-Type = 10 ("Reception of an invalid object") and Error-Value 9 ("Default MSD is specified for the PCEP session").

6. Backward Compatibility

A PCEP speaker that does not support the SR PCEP capability cannot recognize the SR-ERO or SR-RRO subobjects. As such, it MUST send a PCEP error with Error-Type = 4 (Not supported object) and Error-Value = 2 (Not supported object Type) as per [RFC5440].

Some implementations, which are compliant with an earlier version of this specification, do not send the PATH-SETUP-TYPE-CAPABILITY TLV in their OPEN objects. Instead, to indicate that they support SR, these implementations include the SR-CAPABILITY-TLV as a top-level TLV in the OPEN object. Unfortunately, some of these implementations made it into the field before this document was published in its final form. Therefore, if a PCEP speaker receives an OPEN object in which the SR-CAPABILITY-TLV appears as a top-level TLV, then it MUST interpret this as though the sender had sent a PATH-SETUP-TYPE-

CAPABILITY TLV with a PST list of (0, 1) (that is, both RSVP-TE and SR-TE PSTs are supported) and with the SR-CAPABILITY-TLV as a sub-TLV. If a PCEP speaker receives an OPEN object in which both the SR-CAPABILITY-TLV and PATH-SETUP-TYPE-CAPABILITY TLV appear as top-level TLVs, then it MUST ignore the top-level SR-CAPABILITY-TLV and process only the PATH-SETUP-TYPE-CAPABILITY TLV.

7. Management Considerations

7.1. Policy

PCEP implementation:

- o Can enable SR PCEP capability either by default or via explicit configuration.

- o May generate PCEP error due to unsupported number of SR-ERO or SR-RRO subobjects either by default or via explicit configuration.

7.2. The PCEP Data Model

A PCEP MIB module is defined in [RFC7420]n eeds be extended to cover additional functionality provided by [RFC5440] and [I-D.ietf-pce-pce-initiated-lsp]. Such extension will cover the new functionality specified in this document.

8. Security Considerations

The security considerations described in [RFC5440] and [I-D.ietf-pce-pce-initiated-lsp] are applicable to this specification. No additional security measure is required.

9. IANA Considerations

9.1. PCEP Objects

This document defines a new sub-object type for the PCEP explicit route object (ERO), and a new sub-object type for the PCEP record route object (RRO). The code points for sub-object types of these objects is maintained in the RSVP parameters registry, under the EXPLICIT_ROUTE and ROUTE_RECORD objects. IANA is requested to confirm the early allocation of the following code points in the RSVP Parameters registry for each of the new sub-object types defined in this document.

Object	Sub-Object	Sub-Object Type
EXPLICIT_ROUTE	SR-ERO (PCEP-specific)	36
ROUTE_RECORD	SR-RRO (PCEP-specific)	36

9.2. PCEP-Error Object

IANA is requested to confirm the early allocation of the code-points in the PCEP-ERROR Object Error Types and Values registry for the following new error-values:

Error-Type	Meaning	
10	Reception of an invalid object.	
	Error-value = 2:	Bad label value
	Error-value = 3:	Unsupported number of Segment ERO subobjects
	Error-value = 4:	Bad label format
	Error-value = 5:	Non-identical ERO subobjects
	Error-value = 6:	Both SID and NAI are absent in ERO subobject
	Error-value = 7:	Both SID and NAI are absent in RRO subobject
	Error-value = 9:	Default MSD is specified for the PCEP session
	Error-value = 10:	Non-identical RRO subobjects
	Error-value = TBD1:	Missing PCE-SR-CAPABILITY sub-TLV

Note to IANA: this draft originally had an early allocation for Error-value=11 (Malformed object) in the above list. However, we have since moved the definition of that code point to draft-ietf-pce-lsp-setup-type and we included an instruction in that draft for you to update the reference in the indicated registry. Please ensure that this has happened when you process the present draft.

Note to IANA: the final Error-value (Missing PCE-SR-CAPABILITY sub-TLV) in the above list was defined after the early allocation took place, and so does not currently have a code point assigned. Please

assign a code point from the indicated registry and replace each instance of "TBD1" in this document with the allocated code point.

9.3. PCEP TLV Type Indicators

IANA is requested to confirm the early allocation of the following code point in the PCEP TLV Type Indicators registry.

Value	Meaning	Reference
26	SR-PCE-CAPABILITY	This document

9.4. New Path Setup Type

[I-D.ietf-pce-lsp-setup-type] requests that IANA creates a sub-registry within the "Path Computation Element Protocol (PCEP) Numbers" registry called "PCEP Path Setup Types". IANA is requested to allocate a new code point within this registry, as follows:

Value	Description	Reference
1	Traffic engineering path is setup using Segment Routing.	This document

9.5. New Metric Type

IANA is requested to confirm the early allocation of the following code point in the PCEP METRIC object T field registry:

Value	Description	Reference
11	Segment-ID (SID) Depth.	This document

10. Contributors

The following people contributed to this document:

- Lakshmi Sharma
- Jan Medved
- Edward Crabbe
- Robert Raszuk
- Victor Lopez

11. Acknowledgements

We thank Ina Minei, George Swallow, Marek Zavodsky, Dhruv Dhody, Ing-Wher Chen and Tomas Janciga for the valuable comments.

12. References

12.1. Normative References

[I-D.ietf-idr-bgp-ls-segment-routing-msd]
Tantsura, J., Chunduri, U., Mirsky, G., and S. Sivabalan, "Signaling Maximum SID Depth using Border Gateway Protocol Link-State", draft-ietf-idr-bgp-ls-segment-routing-msd-01 (work in progress), October 2017.

[I-D.ietf-isis-segment-routing-extensions]
Previdi, S., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and j. jefftant@gmail.com, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-13 (work in progress), June 2017.

[I-D.ietf-isis-segment-routing-msd]
Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling MSD (Maximum SID Depth) using IS-IS", draft-ietf-isis-segment-routing-msd-04 (work in progress), June 2017.

[I-D.ietf-ospf-segment-routing-extensions]
Psenak, P., Previdi, S., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", draft-ietf-ospf-segment-routing-extensions-21 (work in progress), October 2017.

[I-D.ietf-ospf-segment-routing-msd]
Tantsura, J., Chunduri, U., Aldrin, S., and P. Psenak, "Signaling MSD (Maximum SID Depth) using OSPF", draft-ietf-ospf-segment-routing-msd-05 (work in progress), June 2017.

[I-D.ietf-pce-lsp-setup-type]
Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying path setup type in PCEP messages", draft-ietf-pce-lsp-setup-type-05 (work in progress), October 2017.

[I-D.ietf-pce-pce-initiated-lsp]
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-11 (work in progress), October 2017.

[I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-12 (work in progress), June 2017.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

[RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <https://www.rfc-editor.org/info/rfc5462>.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <https://www.rfc-editor.org/info/rfc7420>.

[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>.

12.2. Informative References

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <https://www.rfc-editor.org/info/rfc3473>.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, DOI 10.17487/RFC3477, January 2003, <https://www.rfc-editor.org/info/rfc3477>.

[RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <https://www.rfc-editor.org/info/rfc4657>.

Authors' Addresses

Siva Sivabalan
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada

Email: msiva@cisco.com

Clarence Filsfils
Cisco Systems, Inc.
Pegasus Parc
De kleetlaan 6a, DIEGEM BRABANT 1831
BELGIUM

Email: cfilsfil@cisco.com

Jeff Tantsura
Individual
444 San Antonio Rd, 10A
Palo Alto, CA 94306
USA

Email: jefftant.ietf@gmail.com

Wim Henderickx
Nokia
Copernicuslaan 50
Antwerp 2018, CA 95134
BELGIUM

Email: wim.henderickx@alcatel-lucent.com

Jon Hardwick
Metaswitch Networks
100 Church Street
Enfield, Middlesex
UK

Email: jonathan.hardwick@metaswitch.com