Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

S. Alvarez
S. Sivabalan
Z. Ali
Cisco Systems, Inc.
L. Tomotaki
Verizon
V. Lopez
Telefonica I+D

February 14, 2014

PCE Path Profiles
draft-alvarez-pce-path-profiles-01

Abstract

This document describes extensions to the Path Computation Element (PCE) Communication Protocol (PCEP) to signal path profile identifiers. A profile represents a list of path parameters or policies that a PCEP peer may invoke on a remote peer using an opaque identifier. When a path computation client (PCC) initiates a path computation request, the PCC can signal profile identifiers to invoke path parameters or policies defined on the PCE which would influence the path computation. Similarly, when a PCE initiates or updates a path, the PCE can signal profile identifiers to invoke path parameters or policies defined on the PCC which would influence the path setup.

Table of Contents

1. Introduction

1.1. 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].

2. Path Profiles

A path profile represents a list of path parameters or policies that a PCEP peer may invoke on a remote peer using a profile identifier. The receiving peer interprets the identifier according to a local path profile definition. The PATH-PROFILE object defined in Section 4.2 can signal one or more profile identifiers. PCEP carries profile identifiers as opaque values. PCEP peers do not exchange the details of a path profile. The PCE may be stateful or stateless.

3. Procedures

3.1. Capability Advertisement

PCEP peers advertise their capability to support path profile identifiers during the session initialization phase. They include the PATH-PROFILE-CAPABILITY TLV defined in Section 4.1 as part of the OPEN object. A PCEP peer can only signal path profile identifiers if both peers advertised this capability. A peer MUST send a PCErr message with Error-Type=4 (Not supported object), Error-value=1 (Not supported object class) and close the session if it receives a message with a path profile identifier, it supports the extensions in this document and both peers did not advertise this capability.

3.2. PCC-Initiated Paths

A PCC MAY include a PATH-PROFILE object when sending a PCReq message. The PCE uses the path profile identifier to select path parameters or path policies to fulfill the request. The means by which the PCC learns about a particular path profile identifier and decides to include it in a PCReq message are outside the scope of this document. Similarly, the means by which the PCE selects a set of parameters or policies based on the profile identifier for a specific request are outside the scope of this document. The P flag of the PATH-PROFILE object MUST be set.

A PCE may receive a path computation request with an unknown or invalid path profile identifier. The PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=1 (Unknown path profile) if the path profile identifier is not known to the PCE. The PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=2 (Invalid path profile) if the PCE knows about the path profile identifier, but considers the request invalid. As an example, the profile may be invalid because of the path type, the PCEP session type or the originating PCC. The PCEP-ERROR object SHOULD include the path profile identifiers that generated the error condition.

The PCE will determine whether to consider any additional optional objects included in a PCReq message based on policy. As illustrated in Section 3.2.1 and Section 3.2.2, the PCC MAY include other optional objects along with a PATH-PROFILE object as part of a path computation request. The PCC will use the processing-rule (P) flag in the common object header to signal whether it considers those objects mandatory or optional when the PCE performs path computation. Those objects may overlap with the path parameters that the PCE associates with the path profile identifier.

PCE policy may place different kinds of restrictions on PCReq messages that include a PATH-PROFILE object and additional parameters. A PCE MUST send an error message if it receives a request with optional objects signaled as mandatory (P flag = 1) for path computation and PCE policy does not allow such behavior from the originating PCC. In that case, the PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=3 (Unexpected mandatory object). If the objects are signaled as optional (P flag = 0) for path computation, the PCE will decide based on policy whether to consider them or not. When sending the PCRep message for the request, the PCE will use the ignore (I) flag in the common object header to indicate to the PCC whether an object was ignored.

3.2.1. Point-to-Point Paths

[RFC5440] defines the basic structure of a PCReq message for point-to-point paths. This documents extends the message format as follows:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
```

where:

```
<svec-list>::=<SVEC>[<svec-list>]
<request-list>::=<request>[<request-list>]

<request>::= <RP>
             <END-POINTS>
             [<PATH-PROFILE>]
             [<path-computation>]
```

where:

<path-computation> is the list of optional objects used for path computation as defined initially in [RFC5440] and modified in subsequent PCEP extensions.

If present in a PCReq message, the PATH-PROFILE object MUST be the first optional object in the request portion of the message.

3.2.2. Point-to-Multipoint Paths

[RFC6006] defines the basic structure of a PCReq message for point-to-multipoint paths. This documents extends the message format as follows:

TBD

3.3. PCE-Initiated Paths

A PCE MAY include a PATH-PROFILE object when sending a PCInitiate message as defined in [I-D.ietf-pce-pce-initiated-lsp]. The PCC uses the path profile identifier to select path parameters or path policies to be applied during the instantiation of the path. The means by which the PCE learns about a particular path profile identifier and decides to include it in a PCInitiate message are outside the scope of this document. Similarly, the means by which the PCC selects a set of parameters or policies based on the profile identifier for a specific path are outside the scope of this document.

A PCC may receive a path instantiation request with an unknown or invalid path profile identifier. The PCC sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=1 (Unknown path profile) if the path profile identifier is not known to the PCC. The PCC sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=2 (Invalid path profile) if the PCC knows about the path profile identifier, but considers the request invalid. As an example, the profile may be invalid because of the path type, the PCEP session type or the originating PCE. The PCEP-ERROR object SHOULD include the path profile identifiers that generated the error condition.

[I-D.ietf-pce-pce-initiated-lsp] defines the basic structure of a PCInitiate message. This documents extends the message format as follows:

```
<PCInitiate Message> ::= <Common Header>
                           <PCE-initiated-lsp-list>
Where:

  <PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                               [<PCE-initiated-lsp-list>]

  <PCE-initiated-lsp-request> ::= (<PCE-initiated-lsp-instantiation>|
                                   <PCE-initiated-lsp-deletion>)

  <PCE-initiated-lsp-instantiation> ::= <SRP>
                                        <LSP>
                                        <END-POINTS>
                                        <ERO>
                                        [PATH-PROFILE>
                                        [<attribute-list>]

  <PCE-initiated-lsp-deletion> ::= <SRP>
                                   <LSP>
```

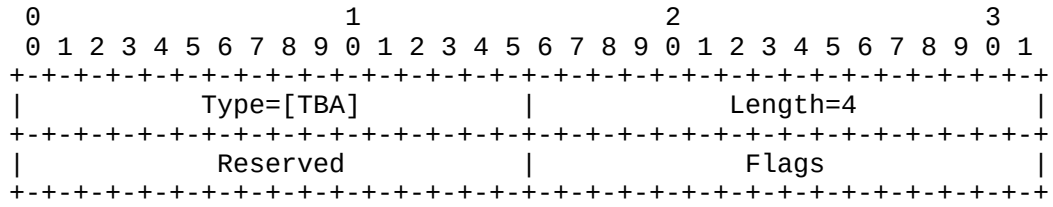
where:

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

4. Object Extensions

4.1. OPEN Object

This documents defines a new optional PATH-PROFILE-CAPABILITY TLV in the OPEN object.



PATH-PROFILE-CAPABILITY TLV

Figure 1

Reserved (16 bits):

MUST be set to zero on transmission and ignored on receipt.

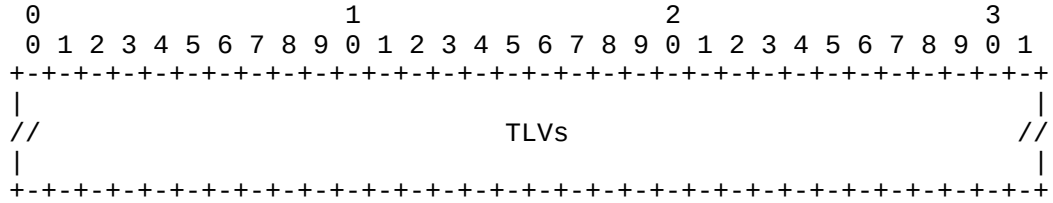
Flags (16 bits):
Unassigned bits are considered reserved. They MUST be set to zero on transmission and ignored on receipt. No flags are currently defined.

4.2. PATH-PROFILE Object

The PATH-PROFILE object may be carried in PCReq, PCInitiate and PCUpd messages.

PATH-PROFILE Object-Class is [TBA].

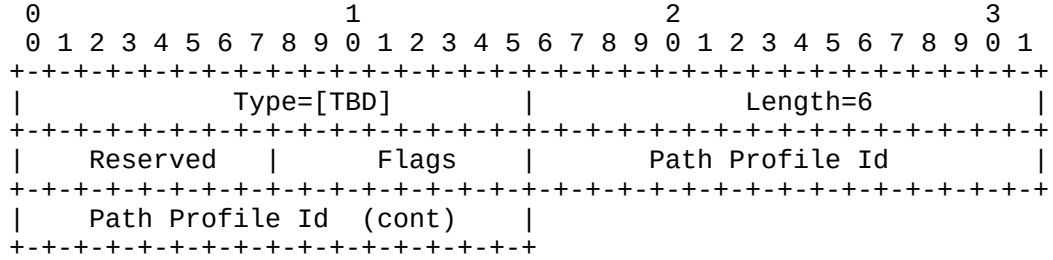
PATH-PROFILE Object-Type is 1.



PATH-PROFILE Object

Figure 2

The PATH-PROFILE object has a variable length and contains one or more PATH-PROFILE-ID TLVs.



PATH-PROFILE-ID TLV

Figure 3

Reserved (8 bits):

MUST be set to zero on transmission and ignored on receipt.

Flags (8 bits):
Unassigned bits are considered reserved. They MUST be set to zero on transmission and ignored on receipt. No flags are currently defined.

Path Profile Id (32 bits):
(non-zero) unsigned path profile identifier.

5. Error Codes for PATH-PROFILE Object

Error-Type	Meaning	Error-Value
<TBA>	PATH-PROFILE Error	1: Unknown path profile
		2: Invalid path profile
		3: Unexpected mandatory object

6. Acknowledgements

The authors would like to thank Clarence Filsfils for his valuable comments.

7. IANA Considerations

IANA is requested to assign the following code points.

PATH-PROFILE-CAPABILITY TLV

PATH-PROFILE Object-Class

PATH-PROFILE Object-Type

PATH-PROFILE Error-Type

8. Security Considerations

TBD

9. References

9.1. Normative References

[I-D.ali-pce-remote-initiated-gmpls-lsp]
Ali, Z., Sivabalan, S., Filsfils, C., Varga, R., Lopez, V., Dios, O., and X. Zhang, "Path Computation Element Communication Protocol (PCEP) Extensions for remote-initiated GMPLS LSP Setup", draft-ali-pce-remote-initiated-gmpls-lsp-02 (work in progress), October 2013.

[I-D.ietf-pce-gmpls-pcep-extensions]
Margaria, C., Dios, O., and F. Zhang, "PCEP extensions for GMPLS", draft-ietf-pce-gmpls-pcep-extensions-08 (work in progress), July 2013.

[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-00 (work in progress), December 2013.

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

[I-D.sivabalan-pce-segment-routing]
Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E., and R. Raszuk, "PCEP Extensions for Segment Routing", draft-sivabalan-pce-segment-routing-02 (work in progress), October 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

9.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

Authors' Addresses

Santiago Alvarez
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134
USA

Email: saalvare@cisco.com

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

Email: msiva@cisco.com

Zafar Ali
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, ON K2K-3E8
Canada

Email: zali@cisco.com

Luis Tomotaki
Verizon
400 International
Richardson, TX 75081
US

Email: luis.tomotaki@verizon.com

Victor Lopez
Telefonica I+D
c/ Don Ramon de la Cruz 84
Madrid 28006
Spain

Email: vlopez@tid.es