This document introduces a simple mechanism to associate a group of Label Switched Paths (LSPs) via an extension to the Path Computation Element Communication Protocol (PCEP) with the purpose of computing diverse paths for those LSPs. The proposed extension allows a PCC to advertise to a PCE the belonging of a particular LSP to a disjoint-group, thus the PCE knows that LSPs in the same group must be disjoint from each other.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

[RFC5440] describes the Path Computation Element communication Protocol (PCEP) which enables the communication between a Path Computation Client (PCC) and a Path Control Element (PCE), or between two PCEs based on the PCE architecture [RFC4655].
PCEP Extensions for Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to enable active control of MPLS-TE and 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 a dynamic network.

[I-D.ietf-pce-association-group] introduces a generic mechanism to create a grouping of LSPs which can then be used to define associations between a set of LSPs and a set of attributes (such as configuration parameters or behaviors) and is equally applicable to the active and passive modes of a stateful PCE [RFC8231] or a stateless PCE [RFC5440].

This document specifies a PCEP extension to signal that a particular group of LSPs should use diverse paths including the requested type of diversity. A PCC can use this extension to signal to a PCE the belonging of a particular LSP to a disjoint-group. When a PCE receives LSP states belonging to the same disjoint-group from some PCCs, the PCE should ensure that the LSPs within the group are disjoint from each other.

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] when, and only when, they appear in all capitals, as shown here.

2. Terminology

The following terminology is used in this document.

LSR: Label Switch Router.

MPLS: Multiprotocol Label Switching.

PCC: Path Computation Client. Any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.


SRLG: Shared Risk Link Group.
3. Motivation

Path diversity is a very common use case in today’s IP/MPLS networks especially for layer 2 transport over MPLS. A customer may request that the operator provide two end-to-end disjoint paths across the IP/MPLS core. The customer may use those paths as primary/backup or active/active.

Different level of disjointness may be offered:

- **Link disjointness**: the paths of the associated LSPs should transit different links (but may use common nodes or different links that may have some shared fate).

- **Node disjointness**: the paths of the associated LSPs should transit different nodes (but may use different links that may have some shared fate).

- **SRLG disjointness**: the paths of the associated LSPs should transit different links that do not share fate (but may use common transit nodes).

- **Node+SRLG disjointness**: the paths of the associated LSPs should transit different links that do not have any common shared fate and should transit different nodes.

The associated LSPs may originate from the same or from different head-end(s) and may terminate at the same or different tail-end(s).
In the figure above, consider that the customer wants to have two disjoint paths between CE1/CE2 and CE3/CE4. From an IP/MPLS network point view, in this example, the CE's are connected to different PEs to maximize their disjointness. When LSPs originate from different head-ends, distributed computation of diverse paths can be difficult. Whereas, computation via a centralized PCE ensures path disjointness correctness and simplicity.

[RFC5440] defines a mechanism for the synchronization of a set of path computation requests by using the SVEC (Synchronization Vector) object, that specifies the list of synchronized requests that can either be dependent or independent. The SVEC object identify the relationship between the set of path computation requests, identified by ‘Request-ID-number’ in RP (Request Parameters) object. [RFC6007] further clarified the use of the SVEC list for synchronized path computations when computing dependent requests as well as described a number of usage scenarios for SVEC lists within single-domain and multi-domain environments.

The SVEC object includes a Flags field that indicates the potential dependency between the set of path computation request in a similar way as the Flags field in the TLVs defined in this document. The path computation request in the PCReq message MAY use both SVEC object to identify the related path computation request as well as to identify the diversity association group. The PCE MUST try to find a path that meets both the constraints. It is possible that the
diversity set in the association group is different from the one in SVEC object, this might be true for the same LSP as well. The PCE would consider both the objects as per the processing rules and aim to find a path that meets both these constraints. In case no such path is possible (or the constraints are incompatible), the PCE MUST send a path computation reply (PCRep) with NO-PATH object indicating path computation failure as per [RFC5440].

The PCEP extension for stateful PCE [RFC8231] defined new PCEP messages - PCrpt, PCUpd and PCInitiate [RFC8281]. These messages uses PLSP-ID in the LSP object for identification. Moreover to allow diversity between LSPs originating from different PCCs, the generic mechanism to create a grouping of LSPs is described in [I-D.ietf-pce-association-group] (that is equally applicable to the active and passive modes of a stateful PCE).

Using PCEP, the PCC could indicate that the disjoint path computation is required, such indication should include disjointness parameters such as the type of disjointness, the disjoint group identifiers, and any customization parameters according to the configured local policy. As mentioned previously, the extension described in [I-D.ietf-pce-association-group] is well suited to associate a set of LSPs with a particular disjoint-group.

The management of the disjoint group-ids will be a key point for the operator as the Association ID field is limited to 65535. The local configuration of IPv4/IPv6 association source, or Global Association Source/Extended Association ID should allow to overcome this limitation as described in [I-D.ietf-pce-association-group]. When a PCC or PCE initiates all the LSPs in a particular disjoint-group, it can set the IPv4/IPv6 association source as one of its own IP address. When disjoint LSPs are initiated from different head-ends, association source could be the PCE address or any other unique value to identify the disjoint association group.
Using the disjoint-group within a PCEP messages may have two purpose:

- Information: in case the PCE is performing the path computation, it may communicate to the PCC the disjoint parameters.
- Configuration: in case the PCC are configured with disjoint requirements, these are communicated to the PCE.

4. Protocol extension

4.1. Association group

As per [I-D.ietf-pce-association-group], LSPs are associated with other LSPs with which they interact by adding them to a common association group. The Association parameters, as described in [I-D.ietf-pce-association-group] as the combination of the mandatory fields Association type, Association ID and Association Source in the ASSOCIATION object, that uniquely identify the association group, uniquely identify the disjoint group. If the optional TLVs - Global
Association Source or Extended Association ID are included, then they are included in combination with mandatory fields to uniquely identifying the association group. This document defines a new Association type, based on the generic Association object -

- Association type = TBD1 ("Disjointness Association Type").

[I-D.ietf-pce-association-group] specify the mechanism for the capability advertisement of the association types supported by a PCEP speaker by defining a ASSOC-Type-List TLV to be carried within an OPEN object. This capability exchange for the association type described in this document (i.e. Disjointness Association Type) MUST be done before using the disjointness association. Thus the PCEP speaker MUST include the Disjointness Association Type (TBD1) in the ASSOC-Type-List TLV before using the disjoint association group in the PCEP messages.

This association type is considered to be both dynamic and operator-configured in nature. The association group could be created by the operator manually on the PCEP peers and the LSPs belonging to this associations is conveyed via PCEP messages to the PCEP peer; or the association group could be created dynamically by the PCEP speaker and both the association group information and the LSPs belonging to the association group is conveyed to the PCEP peer. The Operator-configured Association Range MUST be set for this association-type to mark a range of association identifiers that are used for operator-configured associations to avoid any association identifier clash within the scope of the association source. (Refer [I-D.ietf-pce-association-group].)

A disjoint group can have two or more LSPs. But a PCE may be limited in how many LSPs it can take into account when computing disjointness. If a PCE receives more LSPs in the group than it can handle in its computation algorithm, it SHOULD apply disjointness computation to only a subset of LSPs in the group. The subset of disjoint LSPs will be decided by PCE as a local matter.

Local polices on the PCC or PCE MAY define the computational behavior for the other LSPs in the group. For example, the PCE may provide no path, a shortest path, or a constrained path based on relaxing disjointness, etc.

Associating a particular LSP to multiple disjoint groups is authorized from a protocol perspective, however there is no insurance that the PCE will be able to compute properly the multi-disjointness constraint.
4.2. Disjoint TLVs

The disjoint group MUST carry the following TLV:

- DISJOINTNESS-CONFIGURATION-TLV: Used to communicate some disjointness configuration parameters.

In addition, the disjoint group MAY carry the following TLV:

- DISJOINTNESS-STATUS-TLV: Used to communicate the status of the computed disjointness. This is applicable for messages from PCE to PCC (PCUpd, PCInitiate or PCRep message).

- VENDOR-INFORMATION-TLV: Used to communicate arbitrary vendor specific behavioral information, described in [RFC7470].

The DISJOINTNESS-CONFIGURATION-TLV is shown in the following figure:

```
+----------------------------------+
<p>|       Type = [TBD2]             |</p>
<table>
<thead>
<tr>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flags</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
+--------------------------------+
```

Type: TBD2.

Length: Fixed value of 4 bytes.

Flags:

* L (Link diverse) bit: when set, this indicates that the computed paths within the disjoint group MUST NOT have any link in common.

* N (Node diverse) bit: when set, this indicates that the computed paths within the disjoint group MUST NOT have any node in common.

* S (SRLG diverse) bit: when set, this indicates that the computed paths within the disjoint group MUST NOT share any SRLG (Shared Risk Link Group).

* P (Shortest path) bit: when set, this indicates that the computed path of the LSP SHOULD satisfies all constraints and objective functions first without considering the diversity constraint. This means that an LSP with P flag set should be
placed as if the disjointness constraint has not been configured, while the other LSP in the association with P flag unset should be placed by taking into account the disjointness constraint. Setting P flag changes the relationship between LSPs to a unidirectional relationship (LSP 1 with P=0 depends on LSP 2 with P=1, but LSP 2 with P=1 does not depend on LSP 1 with P=0).

* T (Strict disjointness) bit: when set, if disjoint paths cannot be found, PCE should return no path for LSPs that could not be disjoint. When unset, PCE is allowed to relax disjointness by using either applying a requested objective function or any other behavior if no objective function is requested (e.g.: using a lower disjoint type (link instead of node) or relaxing disjointness constraint at all).

If a PCEP speaker receives a disjoint-group without DISJOINTNESS-CONFIGURATION-TLV, it SHOULD reply with a PCErr Error-type=6 (Mandatory Object missing) and Error-value=TBD7 (DISJOINTNESS-CONFIGURATION-TLV missing).

The DISJOINTNESS-STATUS-TLV uses the same format as the DISJOINTNESS-CONFIGURATION-TLV with a different type TBD3 (in TLV):

Any new flag defined for the DISJOINTNESS-CONFIGURATION-TLV is be automatically applicable to the DISJOINTNESS-STATUS-TLV.

4.3. Disjointness objective functions

An objective function (OF) MAY be applied to the disjointness computation to drive the PCE computation behavior. In this case, the OF-List TLV (defined in ([RFC5541])) is used as an optional TLV in the Association Group Object. The PCEP OF-List TLV allow multiple OF-Codes inside the TLV, a sender SHOULD include a single OF-Code in the OF-List TLV when included in the Association Group, and the receiver MUST consider the first OF-code only and ignore others if included.

To minimize the common shared resources (Node, Link or SRLG) between a set of paths during path computation three new OF codes are proposed:

MSL

* Name: Minimize the number of shared (common) Links.

* Objective Function Code: TBD4
* Description: Find a set of paths such that it passes through the least number of shared (common) links.

MSS

* Name: Minimize the number of shared (common) SRLGs.

* Objective Function Code: TBD5

* Description: Find a set of paths such that it passes through the least number of shared (common) SRLGs.

MSN

* Name: Minimize the number of shared (common) Nodes.

* Objective Function Code: TBD6

* Description: Find a set of paths such that it passes through the least number of shared (common) nodes.

[RFC5440] uses SVEC diversity flag for node, link or SRLG to describe the potential disjointness between the set of path computation requests used in PCEP protocol.

This document defines three new OF codes to maximize diversity as much as possible, in other words, minimize the common shared resources (Node, Link or SRLG) between a set of paths.

It may be interesting to note that the diversity flags in the SVEC object and OF for diversity can be used together. Some example of usage are listed below –

- SVEC object with node-diverse bit=1 - ensure full node-diversity.

- SVEC object with node-diverse bit=1 and OF=MSS - full node diverse with as much as SRLG-diversity as possible.

- SVEC object with domain-diverse bit=1; link diverse bit=1 and OF=MSS - full domain and node diverse path with as much as SRLG-diversity as possible.

- SVEC object with node-diverse bit=1 and OF=MSN - ensure full node-diversity.
As mentioned in Section 4.2, the P-flag (when set) indicates that the computed path of the LSP SHOULD satisfies all constraints and objective functions first without considering the diversity constraint. This could be required in some primary/backup scenarios where the primary path should use the more optimal path available (taking into account the other constraints). When disjointness is computed, it is important for the algorithm to know that it should try to optimize the path of one or more LSPs in the disjoint group (for instance the primary path) while other paths are allowed to be longer (compared to a similar path without the disjointness constraint). Without such a hint, the disjointness algorithm may set a path for all LSPs that may not completely fulfill the customer requirement.

In the figure above, a customer has two dual homed sites (CE1/CE3 and CE2/CE4). Consider, this customer wants two disjoint paths between the two sites. Due to physical meshing, the customer wants to use CE1 and CE2 as primary (and CE3 and CE4 are hosted in a remote site for redundancy purpose).

Without any hint (constraint) provided, the PCE may compute the two disjoint LSPs together, leading to PE1->PE2 using a path PE1->R1->R2->PE2 and PE3->PE4 using PE3->R3->R4->PE4. In this case, even if the disjointness constraint is fulfilled, the path from PE1
to PE2 does not use the best optimal path available in the network (RTD may be higher): the customer requirement is thus not completely fulfilled.

The usage of the P-Flag allows the PCE to know that a particular LSP should be tied to the best path as if the disjointness constraint was not requested.

In our example, if the P-Flag is set to the LSP PE1->PE2, the PCE should use the path PE1->R1->R3->R4->R2->PE2 for this LSP, while the other LSP should be disjoint from this path. The second LSP will be placed on PE3->R5->R6->PE4 as it is allowed to be longer.

Driving the PCE disjointness computation may be done in other ways by for instance setting a metric boundary reflecting an RTD boundary. Other constraints may also be used.

The P-Flag allows a simple expression that the disjointness constraint should not make the LSP worst.

Any constraint added to a path disjointness computation may reduce the chance to find suitable paths. The usage of the P-flag, as any other constraint, may prevent to find a disjoint path. In the example above, if we consider that the router R5 is down, if PE1->PE2 has the P-flag set, there is no room available to place PE3->PE4 (the disjointness constraint cannot be fulfilled). If PE->PE2 has the P-flag unset, the algorithm may be able to place PE1->PE2 on R1->R2 link leaving a room for PE3->PE4 using the R3->R4 link. When using P-flag or any additional constraint on top of the disjointness constraint, the user should be aware that there is less chance to fulfill the disjointness constraint.

Multiple LSPs in the same disjoint group may have the P-flag set. In such a case, those LSPs may not be disjoint from each other but will be disjoint from others LSPs in the group that have the P-flag unset.
In the figure above, we still consider the same previous requirements, so PE1->PE2 LSP should be optimized (P-flag set) while PE3->PE4 should be disjoint and may use a longer path.

Regarding PE1->PE2, there are two paths that are satisfying the constraints (ECMP): PE1->R1->R4->R2->PE2 (path 1) and PE1->R1->R3->R4->R2->PE2 (path 2). An implementation may choose one of the paths or even use both (using both may happen in case Segment Routing TE is used, allowing ECMP).

If the implementation elects only one path, there is a chance that picking up one path may prevent disjointness. In our example, if path 2 is used for PE1->PE2, there is no room left for PE3->PE4 while if path 1 is used, PE3->PE4 can be placed on R3->R4 link.

When P-flag is set for an LSP and when ECMPs are available, an implementation MAY select a path that allows disjointness.

4.5. Disjointness computation issues

There may be some cases where the PCE is not able to provide a set of disjoint paths for one or more LSPs in the association.

When the T-bit is set (Strict disjointness requested), if disjointness cannot be ensured for one or more LSPs, the PCE SHOULD reply with a PCUpd message containing an empty ERO. In addition to
the empty ERO Object, the PCE MAY add the NO-PATH-VECTOR TLV ([RFC5440]) in the LSP Object.

This document adds new bits in the NO-PATH-VECTOR TLV:

- bit "TBD7": when set, the PCE indicates that it could not find a disjoint path for this LSP.
- bit "TBD8": when set, the PCE indicates that it does not support the requested disjointness computation.

When the T-bit is unset, the PCE is allowed to reduce the required level of disjointness. The actual level of disjointness computed by the PCE can be reported through the DISJOINTNESS-STATUS-TLV by setting the appropriate flags in the TLV. While the DISJOINTNESS-CONFIGURATION-TLV defines the expected level of disjointness required by configuration, the DISJOINTNESS-STATUS-TLV defines the actual level of disjointness computed.

There are some cases where the PCE may need to completely relax the disjointness constraint in order to provide a path to all the LSPs that are part of the association. A mechanism that allows the PCE to fully relax a constraint is considered by the authors as more global to PCEP rather than linked to the disjointness use case. As a consequence, it is considered as out of scope of the document.

All LSPs in a particular disjoint group MUST use the same combination of T,S,N,L flags in the DISJOINTNESS-CONFIGURATION-TLV. If a PCE receives PCRpt messages for LSPs belonging to the same disjoint group but having an inconsistent combination of T,S,N,L flags, the PCE SHOULD NOT try to compute disjointness path and SHOULD reply a PCErr with Error-type 26 (Association Error) and Error-Value 6 (Association information mismatch) to all PCCs involved in the disjoint group.

5. Security Considerations

This document defines one new type for association, which do not add any new security concerns beyond those discussed in [RFC5440], [RFC8231] and [I-D.ietf-pce-association-group] in itself.

6. IANA Considerations

6.1. Association object Type Indicators

This document defines the following new association type originally defined in [I-D.ietf-pce-association-group].
6.2. PCEP TLVs

This document defines the following new PCEP TLVs:

<table>
<thead>
<tr>
<th>Value</th>
<th>Name</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD2</td>
<td>DISJOINTNESS-CONFIGURATION-TLV</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>TBD3</td>
<td>DISJOINTNESS-STATUS-TLV</td>
<td>[This I.D.]</td>
</tr>
</tbody>
</table>

IANA is requested to manage the space of flags carried in the DISJOINTNESS-CONFIGURATION-TLV defined in this document, numbering them from 0 as the least significant bit.

New bit numbers may be allocated in future.

IANA is requested to allocate the following bit numbers in the DISJOINTNESS-CONFIGURATION-TLV flag space:

<table>
<thead>
<tr>
<th>Bit Number</th>
<th>Name</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Link disjointness</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>1</td>
<td>Node disjointness</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>2</td>
<td>SRLG disjointness</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>3</td>
<td>Shortest-path</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>4</td>
<td>Strict disjointness</td>
<td>[This I.D.]</td>
</tr>
</tbody>
</table>

6.3. Objective Functions

Three new Objective Functions have been defined. IANA has made the following allocations from the PCEP "Objective Function" sub-registry:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD4</td>
<td>MSL</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>TBD5</td>
<td>MSN</td>
<td>[This I.D.]</td>
</tr>
<tr>
<td>TBD6</td>
<td>MSS</td>
<td>[This I.D.]</td>
</tr>
</tbody>
</table>

6.4. NO-PATH-VECTOR bit Flags

This documents defines new bits for the NO-PATH-VECTOR TLV in the "NO-PATH-VECTOR TLV Flag Field" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry:
Bit Number | Name                                                | Reference
-----------|------------------------------------------------------|----------
TBD7        | Disjoint path not found                              | [This I.D.]
TBD8        | Requested disjointness computation not supported     | [This I.D.]

6.5. PCEP-ERROR codes

IANA is requested to allocate new Error Types and Error Values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

<table>
<thead>
<tr>
<th>Error-Type</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Mandatory Object missing</td>
</tr>
<tr>
<td></td>
<td>Error-value=TBD7: DISJOINTNESS-CONFIGURATION TLV missing</td>
</tr>
</tbody>
</table>

7. Manageability Considerations

7.1. Control of Function and Policy

An operator MUST be allowed to configure the disjointness associations and parameters at PCEP peers and associate it with the LSPs.

7.2. Information and Data Models

[RFC7420] describes the PCEP MIB, there are no new MIB Objects for this document.

7.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

7.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440].

7.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.
7.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440].

8. Acknowledgments

A special thanks to author of [I-D.ietf-pce-association-group], this document borrow some of the text from it. Authors would also like to thank Adrian Farrel for his useful comments.

9. References

9.1. Normative References


9.2. Informative References


Authors' Addresses

Stephane Litkowski
Orange
EMail: stephane.litkowski@orange.com

Siva Sivabalan
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada
EMail: msiva@cisco.com
Colby Barth
Juniper Networks

EMail: cbarth@juniper.net

Dhruv Dhody
Huawei

EMail: dhruv.dhody@huawei.com