The use of SVEC (Synchronization VECtor) list for Synchronized dependent path computations

draft-ietf-pce-pcep-svec-list-05.txt

Abstract

A Path Computation Element (PCE) may be required to perform dependent path computations. Dependent path computations are requests that need to be synchronized in order to meet specific objectives. An example of a dependent request would be a PCE computing a set of services which are required to be diverse (disjointed) from each other. When a PCE computes sets of dependent path computation requests concurrently, it is required to use the Synchronization VECtor (SVEC) list for association among the sets of dependent path computation requests. The SVEC object is optional and carried within the Path Computation Element Protocol (PCEP) PCRequest (PCReq) message.

This document does not specify the PCEP SVEC object or procedure. This informational document clarifies the use of the SVEC list for synchronized path computations when computing dependent requests. The document also describes a number of usage scenarios for SVEC lists within single domain and multi-domain environments.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on December 6, 2010.
1. Introduction

[RFC5440] describes the specifications for PCEP (Path Computation Element communication Protocol). PCEP specifies the communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between two PCEs based on the PCE architecture [RFC4655]. PCEP interactions include path computation requests and path computation replies.
The PCE may be required to compute independent and dependent path requests. Path computation requests are said to be independent if they are not related to each other, and therefore not required to be synchronized. Equally a set of dependent path computation requests, that are required to be synchronized, cannot be performed independently of each other. The Synchronization VECTor (SVEC) with a list of the path computation request identifiers carried within the request message allows the PCC or PCE to specify a list of multiple path computation requests that must be synchronized. Section 1.1 (SVEC Object) describes the SVEC object. Section 1.2 (Application of SVEC Lists) describes the application of SVEC lists in certain scenarios.

This informational document clarifies the handling of dependent and synchronized path computation requests, using the SVEC list, based on the PCE architecture [RFC4655] and PCEP [RFC5440]. The document also describes a number of usage scenarios for SVEC lists within single domain and multi-domain environments. This document is not intended to specify the procedure when using SVEC lists for dependent and synchronized path computation requests.

1.1. SVEC Object

When a PCC or PCE sends path computation requests to a PCE, a PCEP Path Computation Request (PCReq) message may carry multiple requests each of which has a unique path computation request identifier. The SVEC with a list of the path computation request identifiers carried within the request message allows the PCC or PCE to specify a list of multiple path computation requests that must be synchronized, and also allows the specification of any dependency relationships between the paths. The path computation requests listed in the SVEC must be handled in a relation to each other (i.e. synchronized).

[RFC5440] defines two synchronous path computation modes for dependent or independent path computation requests specified by the dependency flags (i.e. Node, Link or SRLG diverse flags) in the SVEC. (See [RFC5440] for more details of dependent, independent and synchronous path computation.)

- A set of independent and synchronized path computation requests,
- A set of dependent and synchronized path computation requests.

These computation modes are exclusive each other in a single SVEC. If one of the dependency flags in a SVEC is set, it indicates a set of synchronous path computation requests has a dependency and does not allow any other path computation requests. In order to be synchronized with other path computation requests with a dependency, it is necessary to associate them.
The aim of the SVEC object carried within a PCReq message is to request the synchronization of M path computation requests. Each path computation request is uniquely identified by the Request-ID-number carried within the respective RP object. The SVEC object also contains a set of flags that specify the synchronization type. The SVEC Object is defined in Section 7.13 (SVEC Object) of [RFC5440].

1.2. Application of SVEC Lists

It is important for the PCE, when performing path computations, to synchronize any path computation requests with a dependency. For example, consider two protected end-to-end services:

- It would be beneficial for each back-up path to be disjointed so they do not share the same links and nodes as the working path.
- Two diverse path computation requests would be needed to compute the working and disjointed protected paths.

If the diverse path requests are computed sequentially, fulfillment of the initial diverse path computation without consideration of the second diverse path computation and disjoint constraint may result in the PCE providing sub-optimal path disjoint results for the protected path, or may fail to meet the end-to-end disjoint requirement altogether.

Additionally, SVEC can be applied to end-to-end diverse path computations that traverse multiple domains. [RFC5441] describes two approaches, synchronous (i.e. simultaneous) and 2-step approaches, for the end-to-end diverse path computation across a chain of domains. The path computation procedure is specified for the 2-step approaches in [RFC5521], but no guidelines are provided for a synchronous approach which is described in this document.

The following scenarios are specifically described within this document:

- Single domain, single PCE, dependent and synchronized path computation request.
- Single domain, multi-PCE, dependent and synchronized path request.
- Multi-domain, dependent and synchronized path computation request, including end-to-end diverse path computation.
The association among multiple SVECs for multiple sets of synchronized dependent path computation is also described in this document, as well as disjoint Virtual Shortest Path Tree (VSPT) encoding rule for end-to-end diverse path computation across domains. Path computation algorithms for these path computation scenarios are out of the scope of this document.

The clarifications and use cases in this document are applicable to the Global Concurrent Optimization (GCO) path computation mechanism specified in [RFC5557]. The GCO application provides the capability to optimize a set of services within the network, in order to maximize efficient use of network resources. A single or set of objective functions (OFs) can be applied to a GCO. To compute a set of such traffic-engineered paths for the GCO application, PCEP supports the synchronous and dependent path computation requests required in [RFC4657].

The SVEC association and the disjoint VSPT described in this document do not require any extension to PCEP messages and object formats, when computing a GCO for multiple or end-to-end diverse paths. In addition, the use of multiple SVECs is not restricted to only SRLG, Node and Link diversity currently defined in the SVEC object [RFC5440], but is also available for other dependent path computation requests.

The SVEC association and disjoint VSPT are available to both single PCE path computation and multi-PCE path computation.

2. Terminology

This document uses PCE terminology defined in [RFC4655], [RFC4875], and [RFC5440].

Associated SVECs: A group of multiple SVECs (Synchronization VECTors), defined in this document, to indicate a set of synchronized or concurrent path computations.

Disjoint VSPT: A set of VSPTs, defined in this document, to indicate a set of virtual diverse path tree.

GCO (Global Concurrent Optimization): A concurrent path computation application, defined in [RFC5557], where a set of TE paths is computed concurrently in order to efficiently utilize network resources.

Synchronized: A set of path computation requests is said to be synchronized if the PCE associates the requests, and does not compute each request independently of each other.

VSPT: Virtual Shortest Path Tree defined in [RFC5441].
3. SVEC association scenarios

This section clarifies several path computation scenarios, in which SVEC association can be applied. Also, any combination of scenarios described in this section could be applicable.

3.1. Synchronized computation for diverse path requests

A PCE may compute two or more point-to-point diverse paths, concurrently, in order to increase the probability of meeting primary and secondary path diversity (or disjointness) objective and network resource optimization objective.

Two scenarios can be considered for the SVEC association of point-to-point diverse paths.

- Two or more end-to-end diverse paths

When concurrent path computation of two or more end-to-end diverse paths is requested, SVEC association is needed among diverse path requests. Note here that each diverse path request consists of primary, secondary, and tertiary and beyond path requests, in which all path requests are grouped with one SVEC association.

Consider two end-to-end services that are to be kept separate by using diverse paths. The path computation requests would need to be associated so that diversity could be assured. Consider further that each of these services requires a backup path that can protect against any failure in the primary path. These backup paths must be computed using requests that are associated with the primary paths giving rise to a set of four associated requests.

- End-to-end primary path and its segmented secondary paths

When concurrent path computation for segment recovery paths, as shown in figure 1, is requested, SVEC association is needed between a primary path and several segmented secondary paths.

\[<---------- primary ---------->\]

\[A------B------C---D------E------F\]

\[\ \ / \ \ / \ \ / \ \ / \ \ /\]

\[P---Q---R \quad X---Y---Z\]

\[<--secondary1--> \quad <--secondary2-->\]

Figure 1: Segment Recovery Paths
In this scenario, we assume that the primary path may be pre-computed, which is used for specifying the segment for secondary paths. Otherwise, the segment for secondary path requests are specified in advance, by using Exclude Route Object (XRO) and/or Include Route Object (IRO) constraints in the primary request.

### 3.2. Synchronized computation for point-to-multipoint path requests

For point-to-multipoint path requests, SVEC association can be applied.

- Two or more point-to-multipoint paths

If a point-to-multipoint paths request is represented as a set of point-to-point paths [ID.pce-p2mp-ext], two or more point-to-multipoint path computation requests can be associated for concurrent path computation, in order to optimize network resources.

- Point-to-multipoint paths and their secondary paths

When concurrent path computation of a point-to-multipoint path and its point-to-point secondary paths [RFC4875], or a point-to-multipoint path and its point-to-multipoint secondary paths is requested, SVEC association is needed among these requests. In this scenario, we use the same assumption as "end-to-end primary path and its segmented secondary paths scenario" in section 3.1.

### 4. SVEC association

This section describes the associations among SVECs in a SVEC list.

#### 4.1. SVEC list

PCEP provides the capability to carry one or more SVEC objects in a PCReq message, and this set of SVEC objects within the PCReq message is termed a SVEC list. Each SVEC object in the SVEC list contains a distinct group of path computation requests. When requesting association among such distinct groups, associated SVECs described in this document are used.

#### 4.2. Associated SVECs

"Associated SVECs" defines that there are relationships among multiple SVECs in SVEC list. Note that there is no automatic association in [RFC5440] between the members of one SVEC and the members of another SVEC in the same SVEC list. The associated SVEC is introduced to associate these SVECs, especially for correlating among SVECs with dependency flags.
Request identifiers in the SVEC objects are used to indicate the association among SVEC objects. If the same request-IDs exist in SVEC objects, this indicates these SVEC objects are associated. When associating among SVEC objects, at least one request identifier must be shared between associated SVECs. The SVEC objects can be associated regardless of the dependency flags in each SVEC object, but it is recommended to use a single SVEC if the dependency flags are not set in all SVEC objects. Similarly, when associating among SVEC objects with dependency flags, it is recommended to construct them using a minimum set of associated SVECs, thus avoiding complex relational associations.

Below is an example of associated SVECs. In this example, the first SVEC is associated with the other SVECs, and all of path computation requests contained in the associated SVECs (i.e. Request-ID#1,#2,#3, #4,#X,#Y,#Z) must be synchronized.

```
<SVEC-list>
   <SVEC> without dependency flags
     Request-ID #1, Request-ID #3, Request-ID #X
   <SVEC> with one or more dependency flags
     Request-ID #1, Request-ID #2
   <SVEC> with one or more dependency flags
     Request-ID #3, Request-ID #4
   <SVEC> without dependency flag
     Request-ID #X, Request-ID #Y, Request-ID #Z
</SVEC-list>
```

4.3. Non-associated SVECs

Non-associated SVECs mean that there are no relationships among SVECs. If none of the SVEC objects in the SVEC list on a PCReq message contains a common request-ID, there is no association between the SVECs and so no association between the requests in one SVEC and the requests in another SVEC.

Below is an example of non-associated SVECs that does not contain any common request-IDs.
5. Processing of SVEC list

5.1. Single PCE, single domain environments

In this environment, there is a single PCE within the domain.

When a PCE receives PCReq messages with more than one SVEC objects in the SVEC list, PCEP has to first check the request-IDs in all SVEC objects in order to identify any associations among them.

If there are no matching request-IDs in the different SVEC objects, these SVEC objects are not associated, and then each set of path computation requests in the non-associated SVEC objects has to be computed separately.

If there are matching request-IDs in the different SVEC objects, these SVEC objects are associated, and then all path computation requests in the associated SVEC objects are treated in a synchronous manner for GCO application.

If the PCE does not have capability to handle the associated SVEC objects, it may send a PCErr message with Error-Type="Capability not supported".

In the case that M path computation requests are sent across multiple PCReq messages, the PCE may start a SyncTimer as recommended in Section 7.13.3 (Handling of the SVEC Object) [RFC5440]. In this case, the associated SVECs should also be handled as described in [RFC5440]. I.E. after receiving the entire set of M path computation requests associated by SVECs, the computation should start at one. If the SyncTimer has expired or the following PCReq messages have been malformed; the PCE should cancel the path computation request and respond to the PCC with the relevant PCErr message.
5.2. Multi-PCE, single domain environments

There are multiple PCEs in a domain, to which PCCs can communicate directly, and PCCs can choose an appropriate PCE for load balanced path computation requests. In this environment it is possible dependent path computation requests are sent to different PCEs.

If a PCC sends path computation requests to a PCE and then sends another path computation requests, which are dependent on the first requests and has been associated by using a SVEC list. There is no method for the PCE to correlate the dependent requests sent to different PCEs. No SVEC object correlation function between the PCEs is specified in [RFC5440]. As indentified, no mechanism exists to resolve this problem and the issue is open for future study. Therefore, a PCC must not send dependent path computation requests associated by SVECs to different PCEs.

5.3. Multi-PCE, multi-domain environments

In this environment, there are multiple domains in which PCEs are located in each domain, and end-to-end dependent paths (i.e. diverse path) is computed using multiple PCEs. Note that we assume a chain of PCEs are pre-determined and the BRPC procedure [RFC5441] is in use.

The SVECs can be applied to end-to-end diverse path computations that traverse multiple domains. [RFC5441] describes two approaches, synchronous (i.e. simultaneous) and 2-step approaches, for the end-to-end diverse path computation across a chain of domains. In the 2-step approaches described in [RFC5521], it is not necessary to use the associated SVECs because any of dependency flags in a SVEC object are not set. On one hand, the simultaneous approach may require the associated SVEC because at least one of dependency flags is required in a SVEC object. Thus, a use case of the simultaneous approach is described in this environment.

When a chain of PCEs located in separate domains are used for simultaneous path computations, additional path computation processing is required. It is described in this document (Section 6).

If the PCReq message contains multiple associated SVEC objects and these SVEC objects contain path computation requests that will be sent to the next PCE along the path computation chain, the following procedures are applied.
When a chain of PCEs is a unique sequence for all of path computation requests in a PCReq message, it is not necessary to re-construct associations among SVEC objects. Thus, the PCReq message is passed to the tail end PCE. When a PCReq message contains more than one SVEC objects with the dependency flag set, the contained SVECs may then be associated. PCEs receiving the associated SVECs must maintain their association, and consider their relationship in path computing after receiving a corresponding PCRep message.

When a chain of PCEs is different, it is required that intermediate PCEs receiving such PCReq messages may re-construct associations among SVEC objects, and then send PCReq messages to corresponding PCEs located in neighboring domains. If the associated SVECs are re-constructed at the intermediate PCE, the PCE must not start its path computation until all PCRep messages have been received from all neighbor PCEs. However, a complex PCE implementation is required for SVEC reconstruction, and waiting mechanisms must be implemented. Therefore, it is not recommended to associate path computation requests with different PCE chains. This is open issue and is currently being discussed in [ID.h-pce] which proposes a hierarchical PCE architecture.

6. End-to-end diverse path computation

In this section, the synchronous approach is provided to compute primary and secondary paths simultaneously.

6.1. Disjoint VSPT

The BRPC procedure constructs a VSPT to inform the enquiring PCE of potential paths to the destination node.

In the end-to-end diverse path computation, diversity (or disjointness) information among the potential paths must be preserved in the VSPT to ensure end-to-end disjoint path. In order to preserve diversity (or disjointness) information, disjoint VSPTs are sent in the PCEP PCRep message. The PCReq containing a SVEC object with the appropriate diverse flag set would signal that the PCE should compute a disjoint VSPT.

A definition of the disjoint VSPT is a collection of VSPTs, in which each VSPT contains a potential set of primary and secondary paths.

Figure 2 shows an example network. Here, transit nodes in domains are not depicted, and PCE1 and PCE2 may be located in border nodes. In this network, there are three VSPTs for the potential set of diverse paths shown in Figure 3, when the primary path and secondary path are requested from S1 to D1. These VSPTs consist of a disjoint
VSPT, which is replied to PCE1. When receiving the disjoint VSPT, PCE1 recognizes the disjoint request and disjoint VSPT information. PCE1 will then continue to process the request and compute the diverse path using the BRPC procedure [RFC5441]. The detail encoding for the disjoint VSPT is described in Section 6.2.

Domain1          Domain2
+----------+     +----------+
|   PCE1   |     |   PCE2   |    S1: Source node
|          BN1---BN4     |    D1: Destination node
|   S1      BN2---BN5  |    D1 |     BN1-BN6: Border nodes
|          BN3---BN6     |
+----------+     +----------+

Figure 2: Example network for diverse path computation

VSPT1:          VSPT2:          VSPT3:
    D1          D1            D1
    / \          / \           / \  
    BN4  BN5     BN4  BN6      BN5  BN6

Figure 2: Disjoint VSPT from PCE2 to PCE1

6.2. Disjoint VSPT encoding

Encoding for disjoint VSPT follows the definition of PCEP message encoding in [RFC5440].

PCEP PCRep message returns a disjoint VSPT as <path list> for each RP object (Request Parameter object). The order of <path> in <path list> among <responses> implies a set of primary EROs (Explicit Route Objects) and secondary EROs.

A PCE sending PCRep with a disjoint VSPT can reply with a partial disjoint VSPT based on its network operation policy, but the order of <path> in <path list> must be aligned correctly.
If confidentiality is required between domains, path key mechanism defined in [RFC5520] is used for a disjoint VSPT.

Detailed disjoint VSPT encoding in Figure 2 is shown below, when a primary path and a secondary path are requested from S1 to D1.

- **Request ID #1 (Primary)**
  - ERO1 BN4(TE route ID)- ...-D1(TE-Router ID)  [for VSPT1]
  - ERO2 BN4(TE route ID)- ...-D1(TE-Router ID)  [for VSPT2]
  - ERO3 BN5(TE route ID)- ...-D1(TE-Router ID)  [for VSPT3]

- **Request ID #2 (Secondary)**
  - ERO4 BN5(TE route ID)- ...-D1(TE-Router ID)  [for VSPT1]
  - ERO5 BN6(TE route ID)- ...-D1(TE-Router ID)  [for VSPT2]
  - ERO6 BN6(TE route ID)- ...-D1(TE-Router ID)  [for VSPT3]

### 6.3. Path computation procedure

For end-to-end diverse path computation, the same mode of operation as BRPC procedure can be applied (i.e. Step 1 to Step n in Section 4.2 [RFC5441]). During this procedure, a question is how to recognize disjoint VSPTs.

The recognition of disjoint VSPT is achieved by the PCE sending PCReq to its neighbor PCE which maintains the path computation request (PCReq) information. If PCReq has one or more SVEC object(s) with the appropriate dependency flags, the received PCRep will contain the disjoint VSPT. If not, the received VSPT is a normal VSPT based on the shortest path computation.

Note that the PCE will apply a suitable algorithm for computing requests with disjoint VSPT. The selection and application of the appropriate algorithm is out of scope in this draft.

### 7. Manageability considerations

This section describes manageability considerations specified in [ID.pce-mngabl-reqs].

#### 7.1. Control of Function and Policy
In addition to [RFC5440], PCEP implementations should allow the PCC to be responsible for mapping the requested paths to computation requests. The PCC should construct the SVECs to identify and associate SVEC relationships.

7.2. Information and Data Models, e.g. MIB modules

There are currently no additional parameters for MIB modules. There is value in a MIB module that details the SVEC association. This work is currently out of scope of this document.

7.3. Liveness Detection and Monitoring

The associated SVEC in this document allows PCEs to compute optimal sets of diverse paths. This type of path computation may require more time to obtain its results. Therefore, it is recommended for PCEP to support PCE monitoring mechanism specified in [ID.pce-monitor].

7.4. Verifying Correct Operation

[RFC5440] provides the sufficient descriptions for this document. So, there are no additional considerations.

7.5. Requirements on Other Protocols and Functional Components

This document does not require any other protocol and functional components.

7.6. Impact on Network Operation

[RFC5440] provides descriptions for the mechanisms discussed in this document. There is value in considering that large associated SVECs will require greater PCE resources, compared to non-associated SVECs. Additionally, the sending of large associated SVECs within multiple PCReq messages will require more network resources. Solving these specific issues is out of scope of this document.

8. Security Considerations

This document describes the usage of SVEC list, and does not have any extensions for PCEP protocol. The security of the procedures described in this document depends on PCEP protocol [RFC5440]. However, a PCE that supports associated SVECs may be open to DoS attack from a rogue PCC. A PCE may be made to queue large numbers of requests waiting for other requests that will never arrive. Additionally a PCE might be made to compute exceedingly complex associated SVEC computations. These DoS attacks may be mitigated with the use of practical SVEC list limits, as well as:
9. Using the same number of simultaneous service provisioning would be recommended.

9. Priority-based multi-queuing mechanism in which path computation requests with a smaller SVEC list are prioritized for path computation processing.

9. Specifying which PCCs may request large SVEC associations through PCE access policy control.

9. IANA Considerations

This document has no specific extension for PCEP messages, objects and its parameters and does not require any registry assignment.

10. References

10.1. Normative References


10.2. Informative References


11. Acknowledgements

The authors would like to thank Adrian Farrel, Julien Meuric and Filippo Cugini for their valuable comments.

12. Authors’ Addresses

Itaru Nishioka
NEC Corp.
1753 Shimonumabe,
Kawasaki, 211-8666,
Japan
Phone: +81 44 396 3287
Email: i-nishioka@cb.jp.nec.com

Daniel King
Old Dog Consulting
United Kingdom
Phone: +44 7790 775187
Email: daniel@olddog.co.uk