

Yang model for requesting Path Computation

draft-ietf-teas-yang-path-computation-07
IETF 106 – Singapore

Italo Busi (Huawei)
Sergio Belotti (Nokia)
Daniele Ceccarelli (Ericsson)
Victor Lopez, Oscar Gonzales de Dios (Telefonica)
Michael Scharf (individual)
Anurag Sharma
Yan Shi (China Unicom)
Ricard Vilalta (CTTC)
Karthik Sethuraman (NEC)

Summary of changes from v06

- Added text describing in a new section 5.3
 - describing the proposed solution for open issue #49 to allow requesting path computation for protected tunnels
- No YANG modifications
 - waiting for WG feedbacks on the proposed approach

Open Issue #49: Path computation for protected tunnels (1)

- Addressing 2 different use cases:
 - the request is to compute both the working and the protection paths for a tunnel that does not exist yet
 - the request is to compute the protection path to add protection for an existing tunnel.
- Taking into account the feedbacks from previous discussions:
 - two path requests within the same RPC: one for the working path and other for the protecting path, with some association

Open Issue #49: Path computation for protected tunnels (2)

	Option 1 (Single Tunnel Request)	Option 2 (Two Path Requests)
PROS	<ul style="list-style-type: none"> tighter alignment with TE tunnel model and operations Implicit support of all associations of paths within a tunnel: e.g. unidir and bidir paths (issue #43) 	<ul style="list-style-type: none"> just add new attributes to the existing YANG model the model already supports requesting the computation of multiple paths with one RPC
CONS	<ul style="list-style-type: none"> Major and heavy restructure of existing YANG model impacts on synchronization vector (SVEC) model unclear different approach than PCEP (using SVEC and ASSOCIATION objects) 	<ul style="list-style-type: none"> need to define new mechanisms to associate paths to be used in the same tunnel e.g. unidir and bidir paths #43 duplication of tunnel parameters in multiple requests: e.g. source, destination, bandwidth

The proposed solution in v07 is addressing these drawbacks

Objectives of the proposal

The proposal permits:

1. associating multiple path requests intended to be used within the same tunnel
2. avoiding repeating the same set of per-tunnel parameters on all the requested paths that are intended to belong to the same tunnel
 - the server can easily understand what attributes are intended to be configured per-tunnel and what attributes are intended to be configured per-path

Basic Yang tree concept (1)

```
+---- path-request* [request-id]
| +--- request-id                uint32
| +---- (tunnel-information)?
| | +----:(tunnel-association)
| | | +---- (tunnel-exist)?
| | | | +----:(tunnel-ref)
| | | | | +---- tunnel-ref          leafref
| | | | | +----:(tunnel-association-id)
| | | | | +---- tunnel-association-id  uint32
| | | <...>
| | +----:(tunnel-attributes)
| | <...>
+---- tunnel-associations* [tunnel-association-id]
| +---- tunnel-association-id?      uint32
| <...>
```

- The (tunnel-association) case, associates multiple paths by either
 - Referencing an existing tunnel: e.g., when computing the protection path to add protection for an existing tunnel
 - Referencing an entry to the new tunnel-associations list when computing multiple paths for a tunnel that does not exist yet: tunnel attributes (e.g. tunnel-name, source/destination TTP, encoding and switching-type) are provided here
- The (tunnel-association) case also provides information about the role of the path being requesting within tunnel (primary or secondary)

Basic Yang tree concept (2)

```
+---- path-request* [request-id]
|   +--- request-id                uint32
|   +---- (tunnel-information)?
|   |   +----:(tunnel-association)
|   |   |   <...>
|   |   +----:(tunnel-attributes)
|   |   |   <...>
```

- The (tunnel-attributes) case provides the set of tunnel attributes (e.g. tunnel-name, source/destination TTP, encoding and switching-type) in case there is no need to associated multiple path requests (e.g., path computation for an unprotected tunnel which does not exist yet)
- The server will have all the information to know how to create a tunnel within the operational DS, when requested (alignment with the tunnel model is strengthened)

Open Issues status

- GitHub Repository
<https://github.com/rvilalta/ietf-te-path-computation>
- Tracking Open Issues, discussions and resolutions linked to YANG model
 - Still 10 open, 5 specific for path computation RPC
 - 3 out of them can be addressed with the proposal for issue #49 on protected tunnel
 - » #43 on bidirectional tunnels
 - » #65 on bidirectional path with asymmetric path properties
 - 1 editorial (review terminology)
 - 1 pending the YANG model becoming stable (example of path computation request)

Next Steps

- Resolve current open issues
 - Continue cooperation with TE Tunnel model authors
- Provide guidance for technology specific augmentations
 - Synch up with OTN tunnel model authors, WSON and flex-grid tunnel authors
- Plan to request YANG doctor review at IETF 108