Path Table based Routing Mechanism in Software-Defined Optical Transport Networks (SD-OTN)
draft-li-sdnrg-path-table-routing-in-sd-otn-06

Status of this Memo

This Internet-Draft is submitted 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 October 05, 2019.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.
Abstract

The dynamic establishment and removal of an end-to-end light-path consume a lot of time which are also the main burden of control plane in optical transport networks. With the frequent arrival and departure of services, the control plane needs to handle a large number of control and management messages to conduct path computation, resource reservation and cross connection configuration. This draft proposes a novel routing mechanism based on Path Table for software-defined optical transport networks (SD-OTN). The Path Table reserves partial activated established light-paths that can be directly used by subsequent requests for subsequent services. This new routing mechanism can reduce the network load and save routing time for some services.
1. Introduction

This document describes the principle of Path Table based routing mechanism for software-defined optical transport networking (SD-OTN). With the development of bandwidth intensive applications such as cloud services, high definition, social networking, big data, etc., optical transport networks which take the advantage of high-capacity and low propagation delay become an important infrastructure for data transmission. The optical transport networks need to dynamically establish end-to-end light-path for each service. The established end-to-end light-path is used to transfer data for a period of time in the optical transport networks. The dynamic establishment and removal of an end-to-end light-path is implemented by control plane. The traditional control plane adopts GMPLS protocol which is composed of OSPF module, LRM module, RSVP module, PCE module, etc. The OSPF module or PCE module conducts path computation and RSVP module realizes resource reservation. Recently, the software defined network (SDN) which takes advantage of centralized control is proposed. In SDN architecture, the control plane is extracted from the data plane and realized in a centralized controller. The controller communicates with data plane through multiple protocols such as OpenFlow and PCEP. SDN is now being applied to the optical transport networks using ROADM and multiple modulation levels programmable transponders. SDN-enabled optical transport networks are called software-defined optical transport network (SD-OTN), which is expected to be more open, programmable, and application aware. In SDON, the establishment and removal of an end-to-end light-path is implemented by the centralized controller. When one request arrives, the controller will compute the path and distribute the cross connection message by southbound protocol for optical transport networks. The dynamic establishment and removal of an end-to-end light-path consumes a lot of time and increases the control and managing burden on optical transport network. This draft proposes a novel Path Table based routing mechanism for software-defined optical transport networks (SD-OTN). The Path Table reserves partial activated established light-paths that can be used by subsequent requests for subsequent services. If a subsequent service matches a light-path entry in the Path Table, the controller does not need to establish a new end-to-end light-path for this request and only assign the existing light-path to this request for data transmission. This new routing mechanism can reduce the network load and save a lot of path computation time for some services.

2. Conventions used in this document

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].
The following acronyms are used in this document.

SDN: Software-Defined Networking
SDON: Software-Defined Optical Networks
GMPLS: Generalized Multi-Protocol Label Switching
PCE: Path Calculation Element
RSVP: Resource Reservation Protocol
OSPF: Open Shortest Path First
TED: Traffic Engineering Database
ROADM: Reconfigurable Optical Add and Drop Multiplexer

3. Motivation of Path Table based Routing Mechanism

The traffic requests arrive and depart from the network dynamically in a random manner. When a request arrives, the control plane must immediately find a light-path available for the traffic demand. With the frequent arrival and departure of service requests, the optical transport networks need to dynamically establish and remove the end-to-end light-path for each service request. The establishment and removal of end-to-end light-path not only consume much time but also will occupy the bandwidth of management channel. The software module in the controller and optical devices need to produce and handle too many messages to conduct path calculation and cross connection configuration. In view of the flow table used at each OpenFlow-enabled switch, we propose the Path Table for SD-OTN. Path Table preserves the existing established light-paths which are not used by any request currently. Taking advantage of Path Table, the controller will first search the suitable light-path in the Path Table for each request. If there exist one suitable light-path matching the current request, the controller assigns the matched light-path to this request. Otherwise, the controller will establish one end-to-end light-path for this request. From the perspective of time statistical point of traffics, the Path Table will get enormous benefits.

4. Path Table based Routing Architecture

This section first gives an overview of the architecture of Path Table based routing model.
4.1. Architecture

Figure 1 below shows the component of a centralized controller with PCE. PCE is a special computational component which can support large, multi-domain, multi-region and multi-layer network constraint-based path computation. The OpenFlow protocol is used to exchange message between optical devices and controller. Service requests of provisioning end-to-end light-path are received by the controller. The controller will transmit the request to the PCE modular. The PCE modular operates on the Path Table and TED to respond with the requested path.

![Path Table based routing architecture](image)

4.2. Path Table

The Path Table consists of a lot of path entries. Each path entry represents one already established light-path in the SD-OTN and is identified by its source-destination node and bandwidth.
Each path entry (as presented in Table 1) contains:

Match Fields: match the service requests. It consists of the source node, destination node and bandwidth.

Source: the node at which the traffic uploads.

Destination: the node at which the traffic downloads.

Bandwidth: the assigned spectrum or wavelengths for the service request.

Priority: matching precedence of the path entry.

Counters: updated when requests are matched.

Instructions: to send the request to next table or assign the matched light-path to this request.

Timeouts: maximum amount of time or idle time before path is expired by the controller.
4.3. Matching

Once a request arrives, the controller performs the functions as shown in Figure 2. The controller starts with performing a table lookup in the first path table and may perform table lookups in other path tables. The source node, destination node and bandwidth are extracted from the request. Light-path match fields used for table lookups depend on the request's source-destination and bandwidth.

A request matches a path table entry if the source node, destination node and bandwidth all match fields used for the lookup match those defined in the path table entry. If a path table entry field has a value of ANY (field omitted), it matches all possible values. The request is matched with the path table and only the highest priority light-path entry that matches the request must be selected.
counters associated with the selected path entry must be updated and
the action included in the selected path entry must be carried out.
If there are multiple matched light-path entries with the same
highest priority, the selected light-path entry is explicitly
undefined.

4.4. Table-miss

Every path table must support a table-miss light-path entry to
process table misses. The table-miss flow entry specifies how to
process requests unmatched by other light-path entries in the path
table, may send the request to PCE or other controller. The table-
miss light-path entry is identified by its match and its priority, it
wildcards all match fields and has the lowest priority. The match of
the table-miss light-path entry must support sending the request to
the PCE. It also may support sending the request to the other
controller. The table-miss path entry behaves in the same way as any
other path entry, but it does exist by default in each path table.

4.5. Path Removal

Path entries are removed from path table in two ways, either via the
controller, or the light-path (path entry) expiry mechanism.

The path expiry mechanism is run dependently in the controller and it
is based on the state and configuration of path entries. Each path
entry has an idle timeout and a hard timeout associated with it. If the
hard timeout field is non-zero, the controller must note the path
entry’s arrival time, at it may need to evict the entry later. A non-
zero hard timeout field causes the path entry to be removed after the
given number of seconds, regardless of how many requests it has
matched. A non-zero idle timeout field causes the path entry to be
removed when it has matched no paths in the given number of seconds.
The controller must implement path expiry and remove path entries
from the path table when one of their timeouts is exceeded.

The controller may actively remove path entries from path table. Path
entries may be removed from path tables when controller needs to
reclaim resources. The controller selects the removed path depending
on path entry parameters, resource mapping in the controller and
other internal controller constraints.

4.6. Counters
Counters are maintained for each path table and path entry. Counters may be implemented in software and maintained by polling hardware counters with more limited ranges. Table 2 contains the set of counters defined by our suggestions. A controller is not required to support all counters, just those marked "required" in Table 2. Duration refers to the amount of time the path entry has been installed in the controller.

<table>
<thead>
<tr>
<th>Counter</th>
<th>Bits</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Per Flow Entry</td>
<td></td>
<td>-------------</td>
</tr>
<tr>
<td>Reference Count (active entries)</td>
<td>32</td>
<td>Required</td>
</tr>
<tr>
<td>Path Lookups</td>
<td>64</td>
<td>Optional</td>
</tr>
<tr>
<td>Path Matches</td>
<td>64</td>
<td>Optional</td>
</tr>
<tr>
<td>Per Path Entry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Received requests</td>
<td>64</td>
<td>Optional</td>
</tr>
<tr>
<td>Duration (seconds)</td>
<td>32</td>
<td>Optional</td>
</tr>
<tr>
<td>Duration (nanoseconds)</td>
<td>32</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Table 2 List of counters

4.7. Instructions

Each path entry contains a set of instructions which are executed when a request matches the entry. A controller is not required to support all instruction types, just those marked "Required Instruction" below.

Optional Instruction: send the request to other controller.

Required Instruction: assign the matched light-path to this request.

Required Instruction: transmit this request to PCE

The instruction set associated with a path entry contains a maximum of one instruction of each type.
5. Security Considerations

TBD

6. IANA Considerations

This document makes no request of IANA.

7. References

7.1. Normative References


8. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot.
Authors’ Addresses

Xin Li
Beijing University of Posts and Telecommunications
Email: xinli@bupt.edu.cn

Yu Zhou
Beijing University of Posts and Telecommunications
Email: shaxiaoziningyi@bupt.edu.cn

Dajiang Wang
ZTE Corporation
Email: wang.dajiang@zte.com.cn

Zhenyu Wang
ZTE Corporation
Email: wang.zhenyul@zte.com.cn

Jiayu Wang
ZTE Corporation
Email: wang.jiayu1@zte.com.cn

Wenzhe Li
Beijing University of Posts and Telecommunications
Email: lwz@bupt.edu.cn

Shan Yin
Beijing University of Posts and Telecommunications
Email: yinshan@bupt.edu.cn