TEAS Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 7, 2017

Y. Zhuang
Q. Wu
H. Chen
Huawei
A. Farrel
Juniper Networks
January 3, 2017

Architecture for Scheduled Use of Resources
draft-ietf-teas-scheduled-resources-02

Abstract

Time-scheduled reservation of traffic engineering (TE) resources can be used to provide resource booking for TE Label Switched Paths so as to better guarantee services for customers and to improve the efficiency of network resource usage into the future. This document provides a framework that describes and discusses the architecture for the scheduled reservation of TE resources. This document does not describe specific protocols or protocol extensions needed to realize this service.

Table of Contents

1. Introduction

Traffic Engineering Label Switched Paths (TE-LSPs) are connection oriented tunnels in packet and non-packet networks [RFC3209], [RFC3945]. TE-LSPs may reserve network resources for use by the traffic they carry, thus providing some guarantees of service delivery and allowing a network operator to plan the use of the resources across the whole network.

In some technologies (such as wavelength switched optical networks) the resource is synonymous with the label that is switched on the path of the LSP so that it is not possible to establish an LSP that can carry traffic without assigning a concrete resource to the LSP. In other technologies (such as packet switched networks) the resources assigned to an LSP are a measure of the capacity of a link that is dedicated for use by the traffic on the LSP. In all cases, network planning consists of selecting paths for LSPs through the

network so that there will be no contention for resources; LSP establishment is the act of setting up an LSP and reserving resources within the network; and network optimization or re-optimization is the process of re-positioning LSPs in the network to make the unreserved network resources more useful for potential future LSPs while ensuring that the established LSPs continue to fulfill their objectives.

It is often the case that it is known that an LSP will be needed at some time in the future. While a path for that LSP could be computed using knowledge of the currently established LSPs and the currently available resources, this does not give any degree of certainty that the necessary resources will be available when it is time to set up the new LSP. Yet setting up the LSP ahead of the time when it is needed (which would guarantee the availability of the resources) is wasteful since the network resources could be used for some other purpose in the meantime.

Similarly, it may be known that an LSP will no longer be needed after some future time and that it will be torn down releasing the network resources that were assigned to it. This information can be helpful in planning how a future LSP is placed in the network.

Time-Scheduled (TS) reservation of TE resources can be used to provide resource booking for TE-LSPs so as to better guarantee services for customers and to improve the efficiency of network resource usage into the future. This document provides a framework that describes and discusses the architecture for the scheduled reservation of TE resources. This document does not describe specific protocols or protocol extensions needed to realize this service.

2. Problem Statement

2.1. Provisioning TE-LSPs and TE Resources

TE-LSPs in existing networks are provisioned using RSVP-TE as a signaling protocol [RFC3209] [RFC3473], by direct control of network elements such as in the Software Defined Networking (SDN) paradigm, and using the PCE Communication Protocol (PCEP) [RFC5440] as a control protocol.

TE resources are reserved at the point of use. That is, the resources (wavelengths, timeslots, bandwidth, etc.) are reserved for use on a specific link and are tracked by the Label Switching Routers (LSRs) at the end points of the link. Those LSRs learn which resources to reserve during the LSP setup process.

The use of TE resources can be varied by changing the parameters of the LSP that uses them, and the resources can be released by tearing down the LSP.

2.2. Selecting the Path of an LSP

Although TE-LSPs can determine their paths hop-by-hop using the shortest path toward the destination to route the signaling protocol messages [RFC3209], in practice this option is not applied because it does not look far enough ahead into the network to verify that the desired resources are available. Instead, the full length of the path of an LSP is computed ahead of time either by the head-end LSR of a signaled LSP, or by Path Computation Element (PCE) functionality in a dedicated server or built into network management software [RFC4655].

Such full-path computation is applied in order that an end-to-end view of the available resources in the network can be used to determine the best likelihood of establishing a viable LSP that meets the service requirements. Even in this situation, however, it is possible that two LSPs being set up at the same time will compete for scarce network resources meaning that one or both of them will fail to be established. This situation is avoided by using a centralized PCE that is aware of the LSP setup requests that are in progress.

2.3. Planning Future LSPs

LSPs may be established "on demand" when the requester determines that a new LSP is needed. In this case, the path of the LSP is computed as described in Section 2.2.

However, in many situations, the requester knows in advance that an LSP will be needed at a particular time in the future. For example, the requester may be aware of a large traffic flow that will start at a well-known time, perhaps for a database synchronization or for the exchange of content between streaming sites. Furthermore, the requester may also know for how long the LSP is required before it can be torn down.

The set of requests for future LSPs could be collected and held in a central database (such as at a Network Management System - NMS): when the time comes for each LSP to be set up the NMS can ask the PCE to compute a path and can then request the LSP to be provisioned. This approach has a number of drawbacks because it is not possible to determine in advance whether it will be possible to deliver the LSP since the resources it needs might be used by other LSPs in the network. Thus, at the time the requester asks for the future LSP,

the NMS can only make a best-effort guarantee that the LSP will be set up at the desired time.

A better solution, therefore, is for the requests for future LSPs to be serviced at once. The paths of the LSPs can be computed ahead of time and converted into reservations of network resources during specific windows in the future.

2.4. Looking at Future Demands on TE Resources

While path computation as described in Section 2.2 takes account of the currently available network resources, and can act to place LSPs in the network so that there is the best possibility of future LSPs being accommodated, it cannot handle all eventualities. It is simple to construct scenarios where LSPs that are placed one at a time lead to future LSPs being blocked, but where foreknowledge of all of the LSPs would have made it possible for them all to be set up.

If, therefore, we were able to know in advance what LSPs were going to be requested we could plan for them and ensure resources were available. Furthermore, such an approach enables a commitment to be made to a service user that an LSP will be set up and available at a specific time.

This service can be achieved by tracking the current use of network resources and also a future view of the resource usage. We call this Time-Scheduled TE (TS-TE) resource reservation.

2.5. Requisite State Information

In order to achieve the TS-TE resource reservation, the use of resources on the path needs to be scheduled. Scheduling state is used to indicate when resources are reserved and when they are available for use.

A simple information model for one piece of scheduling state is as follows:

```
{
  link id;
  resource id or reserved capacity;
  reservation start time;
  reservation end time
}
```

The resource that is scheduled can be link capacity, physical resources on a link, CPU utilization, memory, buffers on an interfaces, etc. The resource might also be the maximal unreserved

bandwidth of the link over a time interval. For any one resource there could be multiple pieces of scheduling state, and for any one link, the timing windows might overlap.

There are multiple ways to realize this information model and different ways to store the data. The resource state could be expressed as a start time and an end time as shown above, or could be expressed as a start time and a duration. Multiple periods, possibly of different lengths, may be associated with one reservation request, and a reservation might repeat on a regular cycle. Furthermore, the current state of network reservation could be kept separate from the scheduled usage, or everything could be merged into a single TS database.

This scheduling state information can be used by applications to book resources for future or now, so as to maximize chance of services being delivered. Also, it can avoid contention for resources of LSPs.

Note that it is also necessary to store the information about future LSPs. This information is held to allow the LSPs to be instantiated when they are due and using the paths/resources that have been computed for them, but also to provide correlation with the TS-TE resource reservations so that it is clear why resources were reserved allowing pre-emption and handling release of reserved resources in the event of cancellation of future LSPs.

3. Architectural Concepts

This section examines several important architectural concepts that lead to design decisions that will influence how networks can achieve TS-TE in a scalable and robust manner.

3.1. Where is Scheduling State Held?

The scheduling state information described in Section 2.5 has to be held somewhere. There are two places where this makes sense:

- o In the network nodes where the resources exist;

- o In a central scheduling controller where decisions about resource allocation are made.

The first of these makes policing of resource allocation easier. It means that many points in the network can request immediate or scheduled LSPs with the associated resource reservation and that all such requests can be correlated at the point where the resources are

allocated. However, this approach has some scaling and technical problems:

- The most obvious issue is that each network node must retain the full time-based state for all of its resources. In a busy network with a high arrival rate of new LSPs and a low hold time for each LSP, this could be a lot of state. Yet network nodes are normally implemented with minimal spare memory.

- In order that path computation can be performed, the computing entity normally known as a Path Computation Element (PCE) [RFC4655] needs access to a database of available links and nodes in the network, and of the TE properties of the links. This database is known as the Traffic Engineering Database (TED) and is usually populated from information advertised in the IGP by each of the network nodes or exported using BGP-LS [RFC7752]. To be able to compute a path for a future LSP the PCE needs to populate the TED with all of the future resource availability: if this information is held on the network nodes it must also be advertised in the IGP. This could be a significant scaling issue for the IGP and the network nodes as all of the advertised information is held at every network node and must be periodically refreshed by the IGP.

- When a normal node restarts it can recover resource reservation state from the forwarding hardware, from Non-Volatile Random-Access Memory (NVRAM), or from adjacent nodes through the signaling protocol [RFC5063]. If scheduling state is held at the network nodes it must also be recovered after the restart of a network node. This cannot be achieved from the forwarding hardware because the reservation will not have been made, could require additional expensive NVRAM, or might require that all adjacent nodes also have the scheduling state in order to re-install it on the restarting node. This is potentially complex processing with scaling and cost implications.

Conversely, if the scheduling state is held centrally it is easily available at the point of use. That is, the PCE can utilize the state to plan future LSPs and can update that stored information with the scheduled reservation of resources for those future LSPs. This approach also has several issues:

- If there are multiple controllers then they must synchronize their stored scheduling state as they each plan future LSPs, and must have a mechanism to resolve resource contention. This is relatively simple and is mitigated by the fact that there is ample processing time to re-plan future LSPs in the case of resource contention.

- If other sources of immediate LSPs are allowed (for example, other controllers or autonomous action by head-end LSRs) then the changes in resource availability caused by the setup or tear down of these LSPs must be reflected in the TED (by use of the IGP as currently) and may have an impact of planned future LSPs. This impact can be mitigated by re-planning future LSPs or through LSP preemption.

- If other sources of planned LSPs are allowed, they can request path computation and resource reservation from the centralized PCE using PCEP [RFC5440].

- If the scheduling state is held centrally at a PCE, the state must be held and restored after a system restart. This is relatively easy to achieve on a central server that can have access to non-volatile storage. The PCE could also synchronize the scheduling state with other PCEs after restart. See Section 4.2 for details.

- Of course, a centralized system must store information about all of the resources in the network. In a busy network with a high arrival rate of new LSPs and a low hold time for each LSP, this could be a lot of state. This is multiplied by the size of the network measured both by the number of links and nodes, and by the number of trackable resources on each link or at each node. The challenge may be mitigated by the centralized server being dedicated hardware, but the problem of collecting the information from the network is only solved if the central server has full control of the booking of resources and the establishment of new LSPs.

Thus the architectural conclusion is that scheduling state should be held centrally at the point of use and not in the network devices.

3.2. What State is Held?

As already described, the PCE needs access to an enhanced, time-based TED. It stores the traffic engineering (TE) information such as bandwidth for every link for a series of time intervals. There are a few ways to store the TE information in the TED. For example, suppose that the amount of the unreserved bandwidth at a priority level for a link is Bj in a time interval from time Tj to Tk (k = j+1), where j = 0, 1, 2, ...

```
        Bandwidth
         ^
         |                                   B3
         |          B1                       ___________
         |          __________
         |B0                                            B4
         |__________          B2                        _________
         |                    ________________
         |
        -+-------------------------------------------------------> Time
         |T0        T1        T2              T3        T4
```

Figure 1: A Plot of Bandwidth Usage against Time

The unreserved bandwidth for the link can be represented and stored in the TED as [T0, B0], [T1, B1], [T2, B2], [T3, B3], ... as shown in Figure 1.

But it must be noted that service requests for future LSPs are known in terms of the LSPs whose paths are computed and for which resources are scheduled. For example, if the requester of a future LSP decides to cancel the request or to modify the request, the PCE must be able to map this to the resources that were reserved. When the LSP or the request for the LSP with a number of time intervals is cancelled, the PCE must release the resources that were reserved on each of the links along the path of the LSP in every time intervals from the TED. If the bandwidth reserved on a link for the LSP is B from time T2 to T3 and the unreserved bandwidth on the link is B2 from T2 to T3, B is added to the link for the time interval from T2 to T3 and the unreserved bandwidth on the link from T2 to T3 will be B2 + B.

This suggests that the PCE needs an LSP Database (LSP-DB) [I-D.ietf-pce-stateful-pce] that contains information not only about LSPs that are active in the network, but also those that are planned. The information for an LSP stored in the LSP-DB includes for each time interval that applies to the LSP: the time interval, the paths computed for the LSP satisfying the constraints in the time interval, and the resources such as bandwidth reserved for the LSP in the time interval. See also Section 2.3

It is an implementation choice how the TED and LSP-DB are stored both for dynamic use and for recovery after failure or restart, but it may be noted that all of the information in the scheduled TED can be recovered from the active network state and from the scheduled LSP-DB.

4. Architecture Overview

The architectural considerations and conclusions described in the previous section lead to the architecture described in this section and illustrated in Figure 2. The interfaces and interactions shown on the figure and labeled (a) through (f) are described in Section 4.1.

```
          -------------------
         | Service Requester |
          -------------------
                     ^
                    a|
                     v
                  -------   b   --------
                 |       |<--->| LSP-DB |
                 |       |      --------
                 |  PCE  |
                 |       |  c    -----
                 |       |<---->| TED |
                  -------        -----
                  ^     ^
                  |     |
                 d|     |e
                  |     |
            ------+-----+--------------------
                  |     |          Network
                  |     --------
                  |    | Router |
                  v     --------
                -----          -----
               | LSR |<------>| LSR |
                -----     f    -----
```

Figure 2: Reference Architecture for Scheduled Use of Resources

4.1. Service Request

As shown in Figure 2, some component in the network requests a service. This may be an application, an NMS, an LSR, or any component that qualifies as a Path Computation Client (PCC). We show this on the figure as the "Service Requester" and it sends a request to the PCE for an LSP to be set up at some time (either now or in the future). The request, indicated on Figure 2 by the arrow (a), includes all of the parameters of the LSP that the requester wishes to supply such as bandwidth, start time, and end time. Note that the

requester in this case may be the LSR shown in the figure or may be a distinct system.

The PCE enters the LSP request in its LSP-DB (b), and uses information from its TED (c) to compute a path that satisfies the constraints (such as bandwidth) for the LSP in the time interval from the start time to the end time. It updates the future resource availability in the TED so that further path computations can take account of the scheduled resource usage. It stores the path for the LSP into the LSP-DB (b).

When it is time (i.e., at the start time) for the LSP to be set up, the PCE sends a PCEP Initiate request to the head end LSR (d) providing the path to be signaled as well as other parameters such as the bandwidth of the LSP.

As the LSP is signaled between LSRs (f) the use of resources in the network is updated and distributed using the IGP. This information is shared with the PCE either through the IGP or using BGP-LS (e), and the PCE updates the information stored in its TED (c).

After the LSP is set up, the head end LSR sends a PCEP LSP State Report (PCRpt message) to the PCE (d). The report contains the resources such as bandwidth usage for the LSP. The PCE updates the status of the LSP in the LSP-DB according to the report.

When an LSP is no longer required (either because the Service Requester has cancelled the request, or because the LSP's scheduled lifetime has expired) the PCE can remove it. If the LSP is currently active, the PCE instructs the head-end LSR to tear it down (d), and the network resource usage will be updated by the IGP and advertised back to the PCE through the IGP or BGP-LS (e). Once the LSP is no longer active, the PCE can remove it from the LSP-DB (b).

4.2. Initialization and Recovery

When a PCE in the architecture shown in Figure 2 is initialized, it must learn state from the network, from its stored databases, and potentially from other PCEs in the network.

The first step is to get an accurate view of the topology and resource availability in the network. This would normally involve reading the state direct from the network via the IGP or BGP-LS (e), but might include receiving a copy of the TED from another PCE. Note that a TED stored from a previous instantiation of the PCE is unlikely to be valid.

Next, the PCE must construct a time-based TED to show scheduled resource usage. How it does this is implementation specific and this document does not dictate any particular mechanism: it may recover a time-based TED previously saved to non-volatile storage, or it may reconstruct the time-based TED from information retrieved from the LSP-DB previously saved to non-volatile storage. If there is more than one PCE active in the network, the recovering PCE will need to synchronize the LSP-DB and time-based TED with other PCEs (see Section 4.3).

Note that the stored LSP-DB needs to include the intended state and actual state of the LSPs so that when a PCE recovers it is able to determine what actions are necessary.

4.3. Synchronization Between PCEs

If there is more than one PCE that supports scheduling active in the network, it is important to achieve some consistency between the scheduled TED and scheduled LSP-DB held by the PCEs.

[RFC7399] answers various questions around synchronization between the PCEs. It should be noted that the time-based "scheduled" information adds another dimension to the issue of synchronization between PCEs. It should also be noted that a deployment may use a primary PCE and the have other PCEs as backup, where a backup PCE can take over only in the event of a failure of the primary PCE. Alternatively, the PCEs may share the load at all times. The choice of the synchronization technique is largely dependent on the deployment of PCEs in the network.

One option for ensuring that multiple PCEs use the same scheduled information is simply to have the PCEs driven from the same shared database, but it is likely to be inefficient and interoperation between multiple implementations will be harder.

Another option is for each PCE to be responsible for its own scheduled database and to utilize some distributed database synchronization mechanism to have consistent information. Depending on the implementation, this could be efficient, but interoperation between heterogeneous implementations is still hard.

A further approach is to utilize PCEP messages to synchronize the scheduled state between PCEs. This approach would work well if the number of PCEs which support scheduling is small, but as the number increases considerable message exchange needs to happen to keep the scheduled databases synchronized. Future solutions could also utilize some synchronization optimization techniques for efficiency. Another variation would be to request information from other PCEs for

a particular time slice, but this might have impact on the optimization algorithm.

5. Multi-Domain Considerations

Multi-domain path computation usually requires some form of cooperation between PCEs each of which has responsibility for determining a segment of the end-to-end path in the domain for which it has computationonal responsiblity. When computing a scheduled path, resources need to be booked in all of the domains that the path will cross so that they are available when the LSP is finlly signalled.

Per-domain path computation [RFC5152] is not an appropriate mechanism when a scheduled LSP is being computed because the computation requests at downstream PCEs are only triggered by signaling. However, a similar mechanism could be used where cooperating PCEs exchange PCReq messages for a scheduled LSP as shown in Figure 3. In this case the service requester asks for a scheduled LSP that will span two domains (a). PCE1 computes a path across Domain 1 and reserves the resources, and also asks PCE2 to compute and reserve in Domain 2 (b). PCE2 may return a full path, or could return a path key [RFC5520]. When it is time for LSP setup PCE1 triggers the head-end LSR (c) and the LSP is signaled (d). If a path key is used, the entry LSR in Domain 2 will consult PCE2 for the path expansion (e) before completing signaling (f).

```
          -------------------
         | Service Requester |
          -------------------
               ^
              a|
               v
            ------         b          ------
           |      |<---------------->|      |
           | PCE1 |                  | PCE2 |
           |      |                  |      |
            ------                    ------
            ^                         ^
            |                         |
           c|                        e|
            |                         |
        ----+-----------------    ----+-----------------
       |    |        Domain 1 |  |    |        Domain 2 |
       |    v                 |  |    v                 |
       |  -----   d   -----   |  |   -----   f   -----  |
       | | LSR |<--->| LSR |<-+--+->| LSR |<--->| LSR | |
       |  -----       -----   |  |   -----       -----  |
        ----------------------    ----------------------
```

Figure 3: Per-Domain Path Computation for Scheduled LSPs

Another mechanism for PCE cooperation in multi-domain LSP setup is Backward- Recursive Path Computation (BRPC) [RFC5441]. This approach relies on the downstream domain supply a variety of potential paths to the upstream domain. Although BRPC can arrive at a more optimal end-to-end path than per-domain path computation, it is not well suited to LSP scheduling because the downstream PCE would need to reserve resources on all of the potential paths and then release those that the upstream PCE announced it did not plan to use.

Finally we should consider hierarchical PCE (H-PCE) [RFC6805]. This mode of operatation is similar to that shown in Figure 3, but a parent PCE is used to coordinate the requests to the child PCEs resulting in better visibility of the end-to-end path and better coordination of the resource booking. The sequenced flow of control is shown in Figure 4.

```
          -------------------
         | Service Requester |
          -------------------
               ^
              a|
               v
            --------
           |        |
           | Parent |
           |  PCE   |
           |        |
            --------
               ^ ^         b
              b| |_______________________
               |                         |
               v                         v
            ------                    ------
           |      |                  |      |
           | PCE1 |                  | PCE2 |
           |      |                  |      |
            ------                    ------
            ^                         ^
            |                         |
           c|                        e|
            |                         |
        ----+-----------------    ----+-----------------
       |    |        Domain 1 |  |    |        Domain 2 |
       |    v                 |  |    v                 |
       |  -----   d   -----   |  |   -----    f   -----  |
       | | LSR |<--->| LSR |<-+--+->| LSR |<--->| LSR | |
       |  -----       -----   |  |   -----       -----  |
        ----------------------    ----------------------
```

Figure 4: Hierarchical PCE for Path Computation for Scheduled LSPs

6. Security Considerations

The protocol implications of scheduled resources are unchanged from "on-demand" LSP computation and setup. A discussion of securing PCEP is found in [RFC5440] and work to extend that security is provided in [I-D.ietf-pce-pceps]. Furthermore, the path key mechanism described in [RFC5520] can be used to enhance privacy and security.

Similarly, there is no change to the security implications for the signaling of scheduled LSPs. A discussion of the security of the signaling protocols that would be used is found in [RFC5920].

However, the use of scheduled LSPs extends the attack surface for a PCE-enabled TE system by providing a larger (logically infinte) window during which an attack can be initiated or planned. That is, if bogus scheduled LSPs can be requested, they can be entered into the LSP-DB, then a large number of LSPs could be launched, or significant network resources could be blocked. Of course, additional authorization could be applied for access to LSP scheduling, and diagnostic tools could inspect the LSP DB to spot attacks.

7. IANA Considerations

This architecture document makes no request for IANA action.

8. Acknowledgements

This work has benefited from the discussions of resource scheduling over the years. In particular the DRAGON project [DRAGON] and [I-D.yong-ccamp-ason-gmpls-autobw-service] both of which provide approaches to auto-bandwidth services in GMPLS networks.

Mehmet Toy, Lei Liu, and Khuzema Pithewan contributed the earlier version of [I-D.chen-teas-frmwk-tts]. We would like to thank the authors of that draft on Temporal Tunnel Services.

Thanks to Michael Scharf and Daniele Ceccarelli for useful comments on this work.

9. Contributors

The following people contributed to discussions that led to the development of this document:

Dhruv Dhody
Email: dhruv.dhody@huawei.com

10. Informative References

[DRAGON] National Science Foundation, "http://www.maxgigapop.net/wp-content/uploads/The-DRAGON-Project.pdf".

[I-D.chen-teas-frmwk-tts]
Chen, H., Toy, M., Liu, L., and K. Pithewan, "Framework for Temporal Tunnel Services", draft-chen-teas-frmwk-tts-01 (work in progress), March 2016.

[I-D.ietf-pce-pceps]
Lopez, D., Dios, O., Wu, W., and D. Dhody, "Secure Transport for PCEP", draft-ietf-pce-pceps-11 (work in progress), January 2017.

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-18 (work in progress), December 2016.

[I-D.yong-ccamp-ason-gmpls-autobw-service]
Yong, L. and Y. Lee, "ASON/GMPLS Extension for Reservation and Time Based Automatic Bandwidth Service", draft-yong-ccamp-ason-gmpls-autobw-service-00 (work in progress), October 2006.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <http://www.rfc-editor.org/info/rfc3209>.

[RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <http://www.rfc-editor.org/info/rfc3473>.

[RFC3945] Mannie, E., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3945, DOI 10.17487/RFC3945, October 2004, <http://www.rfc-editor.org/info/rfc3945>.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <http://www.rfc-editor.org/info/rfc4655>.

[RFC5063] Satyanarayana, A., Ed. and R. Rahman, Ed., "Extensions to GMPLS Resource Reservation Protocol (RSVP) Graceful Restart", RFC 5063, DOI 10.17487/RFC5063, October 2007, <http://www.rfc-editor.org/info/rfc5063>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <http://www.rfc-editor.org/info/rfc5152>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <http://www.rfc-editor.org/info/rfc5440>.

[RFC5441] Vasseur, JP., Ed., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, DOI 10.17487/RFC5441, April 2009, <http://www.rfc-editor.org/info/rfc5441>.

[RFC5520] Bradford, R., Ed., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, DOI 10.17487/RFC5520, April 2009, <http://www.rfc-editor.org/info/rfc5520>.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <http://www.rfc-editor.org/info/rfc5920>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <http://www.rfc-editor.org/info/rfc6805>.

[RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <http://www.rfc-editor.org/info/rfc7399>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <http://www.rfc-editor.org/info/rfc7752>.

Authors' Addresses

Yan Zhuang
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: zhuangyan.zhuang@huawei.com

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Huaimo Chen
Huawei
Boston, MA
US

Email: huaimo.chen@huawei.com

Adrian Farrel
Juniper Networks

Email: afarrel@juniper.net