

INTERNET-DRAFT
Intended Status: Informational draft
Expires: April 4, 2014

Arunkumar Arumuga Nainar
Tata Communications Ltd
October 1, 2013

Dynamic Path Selection (DPS) Based on Application
draft-aumuganainar-rtgwg-dps-00

Abstract

The document describes a network design architecture for routing packets via different paths available in the network based on application port number. Primarily, this is targeted for Enterprise customers who have built up redundancy at their WAN edge but are suffering from a congested primary link whilst the secondary is idle.

The objective of this architecture is as follows

- 1) Offload bulky application on to the secondary link
- 2) Achieve the above with out introducing asymmetric routing in the network

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/lid-abstracts.html>

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

Copyright and License Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1	Introduction	3
1.1	Terminology	4
2.	DPS Architecture Overview.	4
3.	DPS Signaling:-	4
4.	DPS Profile Based Packet Filter	10
5.	DPS Routing Frame Work:-	12
6.	DPS Fault-detection mechanism	14
8.	Implementation Details.	14
7.	Summary	16
8	Security Considerations	17
9	IANA Considerations	17
10	References	17
10.1	Normative References	17
10.2	Informative References	17
	Authors' Addresses	17

1 Introduction

The high availability puzzle can be resolved by building in resiliency to network designs. Whilst active/backup routing schemes are sufficient to create redundancy with low convergence times the following deficiencies and customer demands are not addressed comprehensively.

1. IP routing is essentially best path based. This will lead to underutilized or over utilized links.
2. WAN application performance could be adversely impacted due to congestion whilst the backup link remains idle. Techniques such as DiffServ QoS do address the problem effectively, but those approaches address only the symptoms and not the root cause.
3. Half of the network resources that the end customer has paid for, always remains unused. This is a matter of huge concern for small and mid-size customers as WAN circuit costs are very high and recurring.

Existing Solutions

One way to address the above problems is to load balance the traffic across the available links. To enable load balancing, there are several methods that are available today such as the following.

1. Equal Cost Load balancing
2. GLBP (Global Load Balancing Protocol) based load balancing
3. Optimized Edge Routing (OER) - Cisco proprietary feature
4. Policy based routing

However all these techniques can only be implemented at per-hop level. This would mean load balancing techniques need to be applied on each and every device that the traffic passes through. Failure to do so, might result in asymmetric routing and out of order packets. This invariably results in serious application performance issues.

Proposed solution:-

To address this problem, a new architecture called Dynamic Path Selection or DPS is being proposed. DPS provides the frame work for separating applications that have different QoS requirements and sends them along two different paths in the network. By sending different applications on different links, DPS will be able to

successfully address all the issues reported above with out compromising network availability.

1.1 Terminology

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].

2. DPS Architecture Overview.

The objective of DPS is to achieve end-to-end application separation with out introducing asymmetric routing within the network. In order to ensure the above objectives, we should have a comprehensive mechanism to achieve the following tasks.

Task 1: Any two sites participating in DPS will have to agree on a common set of applications that it will send using either the primary routing path or the secondary routing path (also called a DPS path). This happens in the control plane and will be implemented at the time routing information is exchanged. Please refer to DPS Signaling section for more details.

Task 2: At the time of forwarding the packets, packet should be filtered based on application and the capabilities of remote sites. Packets should than be pushed in to appropriate paths. Please refer to DPS Profile Based Packet filter section for more details.

Task 3: If the packet is pushed in to a DPS path, it should always use the secondary link end to end. This is achieved by building an overlay VPN network (called DPS Routing Domain) over the normal IP/MPLS network using commonly available technologies such as DMVPN (Dynamic Multipoint VPN) tunnels and VRF (Virtual Routing and Forwarding) instances. Please refer to DPS Routing Frame Work section for more details.

Task 4: A comprehensive fault detection mechanism should be put in place to detect the faults in the DPS domain. In such a case, the DPS traffic should be re-routed via the normal routing domain. Please refer to the DPS Fault-Detection & Recovery mechanism section for more details.

3. DPS Signaling:-

DPS Signaling will enable sites to actively exchange their DPS

capabilities dynamically and agree on which set of applications that it will treat as critical and non-critical. DPS architecture assumes existence of dual links on sites that are participating in DPS. For the sake of discussion, the applications to be transported across the first link (also called a primary link) are termed a critical applications and the set of applications that need to be transported across the second link (also called a secondary link) are termed non-critical applications.

In order to achieve the above objective, the Network Manager will be required to define the application profile. Information defined in the application profile will be communicated to all participating sites and a decision will be taken locally based on the profile information received for forwarding the packet.

Definition of DPS Profile:-

A DPS profile is defined as a non-overlapping applications that is treated as critical. The Network Manager will be free to define multiple DPS profiles as long as the application defined in them does not overlap with any of the previously defined DPS profiles.

For example:-

```
Profile 1: { Citrix, SAP, RTP, H.325 }
Profile 2: { FTP , HTTP }
Profile 3: { SMTP, POP3 }
```

.

.

So on and so forth...

Examples quoted above are purely arbitrary and in practice, the definition will be left to the discretion of Network Managers. Any application that is not a member of the critical application set will be treated as non-critical.

Note: Alternatively customers/Network managers can also define non-critical application. In such a application that is not a member of non-critical application set will be treated as Critical.

The definition is valid as long as no application is a member of more than 1 profile. A site on the network can be defined to conform to one or more profiles. In such a case, the list of applications that the given site can potentially treat as critical is the union of all the profiles that it conforms to.

Critical application set for site X = Union of all the conforming profiles.

DPS path selection is unidirectional. In order to avoid asymmetric routing, we must ensure any two participating sites should define a common set of applications as critical. In such a case, if X and Y are two participating sites, then:

Critical Application Set for (X, Y) Pair = Critical Application Set for Site (X) \cap Critical Application Set for Site (Y)

Note: Any application that is not a member of the Critical Application set will be treated as non-critical and will go over the DPS path.

Special Case:-

It is very much possible that there could be a site within the network that does not have DPS capability. For example:

1. Site might be a small site and might not have dual links and hence DPS will not be applicable to them.
2. When a network is being migrated, the sites that have not been migrated to the new network may not understand DPS and hence should not be treated as a DPS capable site.

In such cases routing to and from the sites will have to follow normal IP routing path. To handle this special case, a default profile will be defined called Profile 0:

Profile 0: { } is a null set.

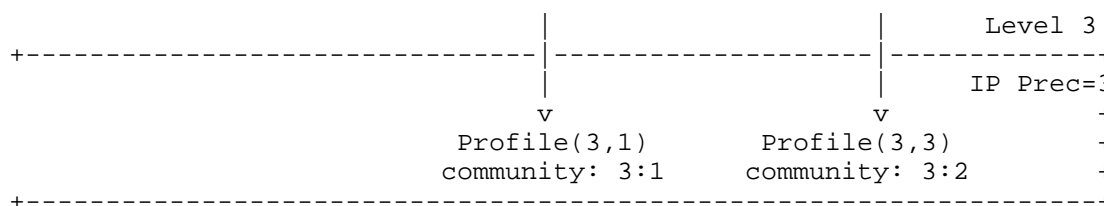
When a DPS capable site X communicates with a non-DPS Capable site Z then:

Critical Application Set for (X,Z) pair =
Critical Application Set for Site (X) \cap Critical Application Set for Site (Z)
= { } or a Null set.

The behavior for Null set is that all traffic will be treated as critical and will be routed via normal routing domain.

Hierarchical model for associating profiles to the site.

In order to aid the following objectives, a hierarchical model based



Usage of non-IP Precedence based marking could possibly extend the depth of the tree. Couple of mechanism are suggested as possible alternatives and listed below.

1. IP DSCP based marking scheme (up to 64 levels possible).
2. QOS Group based marking scheme (up to 100 levels possible).

However marking tree depth or DPS level using IP DSCP or QOS group is not possible using tools currently available in operating systems of networking devices such as Cisco's IOS. It will require minimum amount of code-development effort to take advantage of the above schemes. Till that time, IP Precedence will be used for implementing the framework on a production network and all implementations until that time will be subjected to the known restriction associated with IP Precedence.

In the above tree structure, a site can be associated with any of the profiles located in any of the levels. Under such a scenario, the critical application set is defined by following equation:

$$\text{Critical Application Set for give Node } i,j = \text{Profile}(i,j) \cup \text{Profile}(\text{Parent of Profile}(i,j)) \text{ for all values of } i,j$$

In order to translate the tree structure in to actual deployment practice, each node or profile will be associated with a standard BGP community and each level will be associated with an IP precedence value. The choice of BGP community is arbitrary and is determined by the administrator. The IP precedence value chosen will be equal to the level at which the profile is located. Because DPS signaling relies on BGP community, when the network is deployed, it is mandatory that the primary link of the DPS capable site should run BGP and all the underlying providers support transport of BGP communities.

When a site advertises its routing information, it advertises the community associated with its own profile and all its parents' as well. It should be noted that at any given level, a profile will send only one community (along with the community list of its parent).

Once the communities are sent, the receiving site will interpret the communities. The interpretation of communities is limited to the communities that the given site advertises. Other communities are silently ignored. A site will receive a BGP prefix and associate an IP precedence to the prefix based on the highest level of the matching communities.

For example if a site is in Level N, then it will use following algorithm to associate an IP Precedence for the receiving profile.

```
If Level N community is present , then Set IP Precedence to N
If Level N-1 community is present then Set IP Precedence to N-1
.
.
.
If Level 2 community is a present then Set IP Precedence to 2
If Level 1 community is a present then Set IP Precedence to 1
If there is no matching community at all Set IP Precedence to 0
```

The deployment of above DPS Signaling Mechanism leverages an existing feature called QoS Policy Propagation via BGP (QPPB). This is commonly used feature on networking devices and it is used for propagating QOS marking information in the BGP advertisements. Even though it is not designed to carry DPS signaling, the QPPB functionality is leveraged to achieve DPS signaling. This would mean no additional code changes are required to be done on network devices to achieve this.

Note:- All of the above happen in the control plane (before the packet gets forwarded). However the actual marking happens when the packet hits the site's primary LAN interface. A packet will be remarked as the rules set above using QPPB. Once the packet is marked, then the packet will taken through profile based filtering where the decision will be taken about which routing domain will be referred to while forwarding the packet. Practical Illustration of DPS Profiles

Consider a small network consisting of 20 sites. The sites' profiles are categorized in to 3 types with the below configuration:

- * Type 1: Primary: 10 Mbps; Secondary: 2 Mbps
- * Type 2: Primary: 2 Mbps; Secondary: 8 Mbps/800 Kbps DSL
- * Type 3: Primary: 8 Mbps/800 Kbps DSL; Secondary: None

Common applications used on the network are Citrix, SAP, SMTP, FTP & HTTP. Among which Citrix and SAP are very critical to the business and needs to be protected.

The Network Manager wants to restrict Citrix and SAP to the primary link and the rest to the secondary link. This works well on Type 2 sites. These are small sites predominantly consisting of thin client. However on Type 1 sites are large sites with thick client. Users utilise applications such as SMTP and Lotus notes more than SAP and Citrix. Here a problem is noticed. There is high congestion on the 2 Mbps secondary link. SMTP and FTP are business traffic but by nature they are bulky. Because Type 1 sites have a large number of thick clients, the portion of this traffic is also high. Hence there is the desire to offload SMTP and FTP on to the large 10Mbps link.

Based on the above scenario Profile tree can be built as follows.

Profile 0: { } - This is null set ; BGP Community: None and Precedence = 0.

Profile 1: {Citrix, SAP } with BGP Community : 100:1 and Precedence = 1.

Profile 2: {SMTP, FTP} with BGP Community : 100:2 and Precedence = 2.

This configuration will result in following:

Case 1: When Type 1 talks to Type 1 Site:
Critical Application = {Citrix, SAP, SMTP, FTP}

Case 2: When Type 1 talks to Type 2 Site:
Critical Application = {Citrix, SAP}

Case 3: When Type 2 talks to Type 2 Site:
Critical Application = {Citrix, SAP}

Case 4: When Type 1 talks to Type 3 Site:
Critical Application = { }

Case 5: When Type 2 talks to Type 3 Site:
Critical Application = { }

Case 6: When Type 3 talks to Type 3 Site:
Critical Application = { }

4. DPS Profile Based Packet Filter

DPS Profile Based Packet Filter attempts to filter packets based on DPS profiles and pushes them in to the relevant DPS routing domain or the normal routing domain. It happens in two steps:

> STEP 1:- Colour or mark the packet based on DPS capabilities of the destination site as per the rules set by DPS Signaling.

> STEP 2:- Filter the packets based on application and the DPS capabilities of the source-destination pair.

STEP 1: Colouring or Marking of Packets.

The actual marking happens when the packet hits the routers LAN interface. The packet will be remarked as per the rules set during the DPS signaling by QPPB. Once the packet is marked, the packet will be taken through profile based filtering where the decision to forward it to the relevant routing domain will be taken.

Design Note: Because QPPB remarks the traffic, Trust based QoS model will not be supported when DPS is turned on in a given site. However, QoS can still be applied on DPS capable sites; this is achieved by performing explicit classification and marking at the router before applying QoS policies on the out bound interface.

Note: Current DPS implementation supports only IP Precedence based markings. However with a little bit of development effort other mechanisms such as QoS group can also be adopted. When this is done, restrictions on trust based QoS model will cease to exist. Here the packet is appropriately coloured so that we can pass this through a profile based filter.

1. Application of the incoming packet is an element of Critical Application Set for (X,Z) then it will be push to normal routing domain.

2. Otherwise it will be pushed to DPS routing domain.

3.Special condition rule also applies here, i.e. if Critical Application Set for (X,Z) is a null set then packet will be pushed to normal routing domain.

This Profile based filter will be applied on the LAN interface of the router. Once the traffic hits the primary router, the traffic gets separated as DPS traffic or as normal traffic and gets pushed to appropriate routing domain. Implementation models for Profile based filter is done through two common features/technologies:

1. Packet filters (Access Control List) based on TCP and UDP application port numbers and IP Precedence.

2. Policy based Routing (PBR).

PBR will use simple next hop feature to push the traffic in to the DPS domain (please refer to DPS Routing Framework section for more details). However in case of single router, dual circuit scenario, a modified version of PBR will be used. Here, PBR will be used to select the VRF domain based on which packet will have to be routed. This feature is called VRF selection based on PBR and it is common feature used on most of networking devices including Cisco.

It should be noted that there are several restrictions on PBR match criteria in most implementations such as matching IP Precedence using extend ACLs is not supported. However this mechanism has been tested and implemented in Cisco's software based routing platforms such as ISRs.

Also during our implementation, we have found that PBR had huge impact on routers performance. Hence future implementations based on sleek model using Layer 4 port numbers and IP Precedence could be done to make these processes more efficient.

5. DPS Routing Framework:-

DPS Routing framework provides overlay routing domain for routing packets that belong to non-critical applications. DPS framework assumes the following:

1. Customer sites consist of redundant routers and redundant links. The first link (also called a primary link) will connect to Router 1 (also called a primary router) and will be used to carry traffic belonging to critical applications. Primary link will also carry all the traffic destined for sites that do not support DPS. The second link (also called a secondary link) will connect to Router 2 (also called a secondary router) and will be used to carry traffic belonging to non-critical applications.

2. DPS routing framework also assumes that BGP is enabled across the primary link and the network provider supports transport of BGP communities end to end.

In order to create a DPS routing framework two new interfaces/sub-interfaces will be configured and their details are listed below.

1. Dynamic multipoint tunnel interface (DMVPN tunnel interface). This will be created on the secondary router. The DMVPN tunnel is a point-to-multipoint tunnel interface commonly used in IP Networks for creating any-to-any overlay VPNs.

Source Address of the DMVPN tunnel will only be advertised via secondary link. At the primary router these source addresses will be

filtered out. This ensures that any traffic coming out of tunnel interface will leave the local site via the secondary link and enter the destination site via its secondary link

2. In addition to the tunnel interface, one more sub-interface will be created across the back to back link between the primary and secondary router.

In order to secure the normal and DPS routing domain, new virtual routing and forwarding instances (VRF) will be created on the secondary router. Both the DMVPN tunnel interface and the DPS back to back sub-interface on the secondary router will be assigned to the VRF.

Routing protocols will be enabled on the newly created interface and separate routing protocol instances will be run across the DPS domain. Following peers will be established across these interfaces:

1. 1st peering will be established across DPS back to back interface between primary and secondary router.

2. 2nd peering across DMVPN hub. It should be noted that though routing information is exchanged only with DMVPN hub device, traffic flow will be always happen directly between the spokes. This capability is defined by Next Hop Resolution Protocol (NHRP # RFC 2332) and it is built in to DMVPN tunnel technology. This capability is leveraged to provide any to any communication on the DPS Frame work.

Design Note:- In order to increase the availability of the DPS routing domains it is suggested to host additional DMVPN hubs. In such a case each DPS site will have two peering points via DMVPN tunnel interfaces.

All the LAN routes are pushed in to the DPS domain via peering established across back to back sub interface. This is then propagated across the entire network via a DMVPN tunnel interface. VRF configured on the secondary router ensures that DPS and normal routing information do not get mixed up with each other. If the DPS routing domain is built around the above guidelines, we can ensure that the packet will leave the local site via its secondary link and enter the remote site again via the secondary link.

The above design assumes two routers being used. However the design could be a single router, two circuits scenario as well. In such a case, there is no need for the DPS back to back sub-interface. The rest of details remain the same for the single router scenario.

6. DPS Fault-detection mechanism

As with any networks, faults can happen in a DPS routing domain. DPS by design has got several single points of failure. However DPS has been equipped with sound fault detection and recovery mechanisms. Fault detection and recovery mechanisms will dynamically allow a given router to detect faults that might have happened anywhere (local and remote faults) on the DPS domain. Once the fault is detected the packet is ejected out of the DPS domain and pushed on to the normal routing domain.

Fault detection is enabled through dynamic routing information exchanged via a routing protocol. A fault can happen any where within the site such as:

1. Secondary link could have failed.
2. Back to back link connecting primary and secondary router could have failed.
3. LAN interface on the primary router could have failed.

All of the above failures will result in routing information being withdrawn from the routing table. If a route for a given DPS capable site is not present in the DPS routing table then it is considered a fault.

To enable fault recovery, DPS uses a default static route to push the traffic out of the DPS domain and in to the normal routing domain. During the event default route is used inside the routing domain, we will have to use one or more summary route that encompasses all the LAN routes used with in the network instead of default static routes. This will enable DPS to push the traffic in to the DMVPN tunnel if a more specific route is available. In case a more specific route is not available (this might happen due to local or remote fault) it will use default static route to pop out of DPS domain and back out to the primary router and route via the normal routing domain.

8. Implementation Details.

This architecture has been developed using exiting features available in Cisco IOS. Details are given below.

- 1) DPS Signaling :- QPPB
- 2) Profile based Filter :- PBR and Extended ACL
- 3) Routing Framework :- OSPF, DMVPN and VRF

4) Fault Recovery :- Static Routing

All the components are put to gather as described in previous sections and has been thoroughly tested in labs and also implemented in the field. Current implementations are done using Cisco routers and IOS version 15.0M. OSPF has been used as routing protocol inside the DPS domain and it has been tweaked so that it scales well in large deployments. During lab testing, we were able to scale well using this architecture where it was tested up to 500 sites with 5000 prefixes. In the production environment, several implementations were done with largest one consisting of 300 sites & 2000 prefixes. Following are the challenges that we faced during this implementation. Some of them will require additional development effort:

1. Lack of trust based QoS model. This restriction is particularly important in converged environment where voice and data shares the same infrastructure space. Here customers wanted their providers to support trust based markings. Due to reliance of IP precedence based coloring for identifying DPS capabilities trust model could not be supported.

2. Matching using Extended ACLs based on IP Precedence inside the PBR was also a challenge. All hardware switching based platforms such as Cisco's Catalyst platforms failed during lab testing. However software switching based platforms such as Cisco's ISRs performed really well both in lab and also in the production environment.

3. PBR based filters had severe restriction on throughput of software based routing platform. Additional development work is required to accomplish light weight profile based filters.

To a greater extent, large scale implementation is possible in the present form with out any modifications on any networking hardware that supports the above mentioned features (eg: Cisco IOS). However, with little bit of development effort, we will be able to overcome some of the shortcomings as well. These are listed below

1) Lack of support for trust model has been a major drawback in the current architecture. Though QPPB can mark, QOS-GROUP field, it can not be matched inside a PBR. IOS in its current form only allows classification based on QoS-Group only on output policy. If support can be added for matching QOS-Group inside a PBR then we can do the coloring based on QoS-Group instead of IP Precedence. Hence trust model can be easily supported.

2) PBR is currently used for Profile based filtering. however through put of the device is very much limited when this feature is turned

on. Since filtering is only done on IP Precedence and Application port-number, special filters could be developed to speed up this operations. This could improve the performance of the application even better.

7. Summary

By summarizing all the four components, true end to end application based routing scheme could be achieved. Such DPS frame work has the following advantages:

1. Give lots of room for Network Manager to determine which path should be used for which application.
2. This is very scalable framework.
3. Trouble shooting the setup is easy and simple since it is based on simple routing.
4. DPS capable sites can co-exists with non DPS sites and this capability provides enough room for phased migration. Hence DPS technology adoption is easy and simple.
5. It should be noted that DPS frame work and signaling, needs to be understood only by edge devices and all the devices in middle such as provider routers need not be aware of DPS.

```
Definitions and code {  
  line 1  
  line 2  
}
```

Special characters examples:

The characters , , ,

However, the characters \0, \&, \% , \" are displayed.

.ti 0 is displayed in text instead of used as a directive.

.\" is displayed in document instead of being treated as a comment

C:\dir\subdir\file.ext Shows inclusion of backslash \".

8 Security Considerations

TBD

9 IANA Considerations

TBD

10 References

10.1 Normative References

- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC1776] Crocker, S., "The Address is the Message", RFC 1776, April 1 1995.
- [TRUTHS] Callon, R., "The Twelve Networking Truths", RFC 1925, April 1 1996.

10.2 Informative References

- [EVILBIT] Bellovin, S., "The Security Flag in the IPv4 Header", RFC 3514, April 1 2003.
- [RFC5513] Farrel, A., "IANA Considerations for Three Letter Acronyms", RFC 5513, April 1 2009.
- [RFC5514] Vyncke, E., "IPv6 over Social Networks", RFC 5514, April 1 2009.

11 Acknowledgements

The authors would like to thank Hesham Moussa for his review and comments.

Authors' Addresses

Arunkumar Arumuga Nainar
Tata Communications (UK)
1st Floor
20 Old Bailey

INTERNET DRAFT Dynamic Path Selection Based on Application October 01, 2013

London EC4M 7AN
United Kingdom

EMail: arun.arumuganainar@tatacommunications.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 22, 2014

Pierre Francois
Institute IMDEA Networks
Clarence Filsfils
Ahmed Bashandy
Cisco Systems, Inc.
Bruno Decraene
Stephane Litkowski
Orange
November 18, 2013

Topology Independent Fast Reroute using Segment Routing
draft-francois-segment-routing-ti-lfa-00

Abstract

This document presents a Fast Reroute (FRR) approach aimed at providing link and node protection of node and adjacency segments within the Segment Routing (SR) framework. This FRR behavior builds on proven IP-FRR concepts being LFAs, remote LFAs (RLFA), and remote LFAs with directed forwarding (DLFA). It extends these concepts to provide guaranteed coverage in any IGP network. We accommodate the FRR discovery and selection approaches in order to establish protection over post-convergence paths from the point of local repair, dramatically reducing the operator's need to control the tie-breaks among various FRR options.

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on May 22, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	4
3. Intersecting P-Space and Q-Space with post-convergence paths	5
3.1. P-Space property computation for a resource X	5
3.2. Q-Space property computation for a link S-F, over post-convergence paths	5
3.3. Q-Space property computation for a node F, over post-convergence paths	6
4. EPC Repair Tunnel	6
4.1. The repair node is a direct neighbor	6
4.2. The repair node is a PQ node	6
4.3. The repair is a Q node, neighbor of the last P node	7
4.4. Connecting distant P and Q nodes along post-convergence paths	7
5. Protecting segments	7
5.1. The active segment is a node segment	7
5.2. The active segment is an adjacency segment	7
5.2.1. Protecting [Adjacency, Adjacency] segment lists	8
5.2.2. Protecting [Adjacency, Node] segment lists	8
6. References	8
Authors' Addresses	9

1. Introduction

Segment Routing aims at supporting services with tight SLA guarantees [1]. This document provides local repair mechanisms using SR, capable of restoring end-to-end connectivity in the case of a sudden failure of a link or a node, with guaranteed coverage properties.

Using segment routing, there is no need to establish TLDP sessions with remote nodes in order to take advantage of the applicability of remote LFAs (RLFA) or remote LFAs with directed forwarding (DLFA) [2]. As a result, preferring LFAs over RLFAs or DLFAs, as well as minimizing the number of RLFA or DLFA repair nodes is not required. Using SR, there is no need to create state in the network in order to enforce an explicit FRR path. As a result, we can use optimized detour paths for each specific destination and for each possible failure in the network without creating additional forwarding state.

Building on such an easier forwarding environment, the FRR behavior suggested in this document tailors the repair paths over the post-convergence path from the PLR to the protected destination.

As the capacity of the post-convergence path is typically planned by the operator to support the post-convergence routing of the traffic for any expected failure, there is much less need for the operator to tune the decision among which protection path to choose. The protection path will automatically follow the natural backup path that would be used after local convergence. This also helps to reduce the amount of path changes and hence service transients: one transition (pre-convergence to post-convergence) instead of two (pre-convergence to FRR and then post-convergence).

We provide an EPC-FRR approach that achieves guaranteed coverage against link or node failure, in any IGP network, relying on the flexibility of SR.

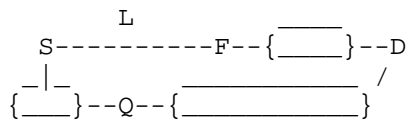


Figure 1: EPC Protection

We use Figure 1 to illustrate the EPC-FRR approach.

The Point of Local Repair (PLR), S , needs to find a node Q (a repair node) that is capable of safely forwarding the traffic to a destination D affected by the failure of the protected link L , or node F . The PLR also needs to find a way to reach Q without being affected by the convergence state of the nodes over the paths it wants to use to reach Q .

In Section 2 we define the main notations used in the document. They are in line with [2].

In Section 3, we suggest to compute the P-Space and Q-Space properties defined in Section 2, for the specific case of nodes lying over the post-convergence paths towards the protected destinations. The failure of a link $S-F$ as well as the failure of a neighbor F is discussed.

Using the properties defined in Section 3, we describe how to compute protection lists that encode a loopfree post-convergence towards the destination, in Section 4.

Finally, we define the segment operations to be applied by the PLR to ensure consistency with the forwarding state of the repair node, in Section 5.

2. Terminology

We define the main notations used in this document as the following.

We refer to "old" and "new" topologies as the LSDB state before and after the considered failure.

$SPT_old(R)$ is the Shortest Path Tree rooted at node R in the initial state of the network.

$SPT_new(R, X)$ is the Shortest Path Tree rooted at node R in the state of the network after the resource X has failed.

$Dist_old(A,B)$ is the distance from node A to node B in $SPT_old(A)$.

$Dist_new(A,B, X)$ is the distance from node A to node B in $SPT_new(A, X)$.

The P-Space $P(R,X)$ of a node R w.r.t. a resource X (e.g. a link $S-F$, or a node F) is the set of nodes that are reachable from R without passing through X . It is the set of nodes that are not downstream of

X in $SPT_old(R)$.

The Extended P-Space $P'(R,X)$ of a node R w.r.t. a resource X is the set of nodes that are reachable from R or a neighbor of R, without passing through X.

The Q-Space $Q(D,X)$ of a destination node D w.r.t. a resource X is the set of nodes which do not use X to reach D in the initial state of the network. In other words, it is the set of nodes which have D in their P-Space w.r.t. S-F (or F).

A symmetric network is a network such that the IGP metric of each link is the same in both directions of the link.

3. Intersecting P-Space and Q-Space with post-convergence paths

In this section, we suggest to determine the P-Space and Q-Space properties of the nodes along on the post-convergence paths from the PLR to the protected destination and compute an SR-based explicit path from P to Q when they are not adjacent. Such properties will be used in Section 4 to compute the EPC-FRR repair list.

3.1. P-Space property computation for a resource X

A node N is in $P(R, X)$ if it is not downstream of X in $SPT_old(R)$.

A node N is in $P'(R,X)$ if it is not downstream of X in $SPT_old(N)$, for at least one neighbor N of R.

3.2. Q-Space property computation for a link S-F, over post-convergence paths

We want to determine which nodes on the post-convergence from the PLR to the destination D are in the Q-Space of destination D w.r.t. link S-F.

This can be found by intersecting the post-convergence path to D, assuming the failure of S-F, with $Q(D, S-F)$.

The post-convergence path to D requires to compute $SPT_new(S, S-F)$.

A node N is in $Q(D,S-F)$ if it is not downstream of S-F in $rSPT_old(D)$.

3.3. Q-Space property computation for a node F, over post-convergence paths

We want to determine which nodes on the post-convergence from the PLR to the destination D are in the Q-Space of destination D w.r.t. node F.

This can be found by intersecting the post-convergence path to D, assuming the failure of F with $Q(D, F)$.

The post-convergence path to D requires to compute $SPT_{new}(S, F)$.

A node N is in $Q(D, F)$ if it is not downstream of F in $rSPT_{old}(D)$.

4. EPC Repair Tunnel

The EPC repair tunnel consists of an outgoing interface and a list of segments (repair list) to insert on the SR header. The repair list encodes the explicit post-convergence path to the destination, which avoids the protected resource X.

The EPC repair tunnel is found by intersecting $P(S, X)$ and $Q(D, X)$ with the post-convergence path to D and computing the explicit SR-based path $EP(P, Q)$ from P to Q when these nodes are not adjacent along the post convergence path. The EPC repair list is expressed generally as $(Node_SID(P), EP(P, Q))$.

Most often, the EPC repair list has a simpler form, as described in the following sections.

4.1. The repair node is a direct neighbor

When the repair node is a direct neighbor, the outgoing interface is set to that neighbor and the repair segment list is empty.

This is comparable to an LFA FRR repair.

4.2. The repair node is a PQ node

When the repair node is in $P(S, X)$, the repair list is made of a single node segment to the repair node.

This is comparable to an RLFA repair tunnel.

4.3. The repair is a Q node, neighbor of the last P node

When the repair node is adjacent to $P(S,X)$, the repair list is made of two segments: A node segment to the adjacent P node, and an adjacency segment from that node to the repair node.

This is comparable to a DLFA repair tunnel.

4.4. Connecting distant P and Q nodes along post-convergence paths

In some cases, there is no adjacent P and Q node along the post-convergence path. However, the PLR can perform additional computations to compute a list of segments that represent a loopfree path from P to Q.

5. Protecting segments

In this section, we explain how a protecting router S processes the active segment of a packet upon the failure of its primary outgoing interface.

The behavior depends on the type of active segment to be protected.

5.1. The active segment is a node segment

The active segment is kept on the SR header, unchanged (1). The repair list is inserted at the head of the list. The active segment becomes the first segment of the inserted repair list.

A future version of the document will describe the FRR behavior when the active segment is a node segment destined to F, and F has failed.

Note (1): If the SRGB at the repair node is different from the SRGB at the PLR, then the active segment must be updated to fit the SRGB of the repair node.

5.2. The active segment is an adjacency segment

We define hereafter the FRR behavior applied by S for any packet received with an active adjacency segment S-F for which protection was enabled. We distinguish the case where this active segment is followed by another adjacency segment from the case where it is followed by a node segment.

5.2.1. Protecting [Adjacency, Adjacency] segment lists

If the next segment in the list is an Adjacency segment, then the packet has to be conveyed to F.

To do so, S applies a "NEXT" operation on Adj(S-F) and then two consecutive "PUSH" operations: first it pushes a node segment for F, and then it pushes a protection list allowing to reach F while bypassing S-F.

Upon failure of S-F, a packet reaching S with a segment list matching [adj(S-F),adj(M),...] will thus leave S with a segment list matching [RT(F),node(F),adj(M)], where RT(F) is the repair tunnel for destination F.

5.2.2. Protecting [Adjacency, Node] segment lists

If the next segment in the stack is a node segment, say for node T, the packet segment list matches [adj(S-F),node(T),...].

A first solution would consist in steering the packet back to F while avoiding S-F, similarly to the previous case. To do so, S applies a "NEXT" operation on Adj(S-F) and then two consecutive "PUSH" operations: first it pushes a node segment for F, and then it pushes a repair list allowing to reach F while bypassing S-F.

Upon failure of S-F, a packet reaching S with a segment list matching [adj(S-F),node(T),...] will thus leave S with a segment list matching [RT(F),node(F),node(T)].

Another solution is to not steer the packet back via F but rather follow the new shortest path to T. In this case, S just needs to apply a "NEXT" operation on the Adjacency segment related to S-F, and push a repair list redirecting the traffic to a node Q, whose path to node segment T is not affected by the failure.

Upon failure of S-F, packets reaching S with a segment list matching [adj(L), node(T), ...], would leave S with a segment list matching [RT(Q),node(T), ...].

6. References

- [1] Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe, "Segment Routing Architecture", draft-filsfils-rtgwg-segment-routing-00 (work in progress), June 2013.

- [2] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, January 2010.
- [3] Filsfils, C., Francois, P., Shand, M., Decraene, B., Uttaro, J., Leymann, N., and M. Horneffer, "Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks", RFC 6571, June 2012.
- [4] Bryant, S., Filsfils, C., Previdi, S., Shand, M., and S. Ning, "Remote LFA FRR", draft-ietf-rtgwg-remote-lfa-02 (work in progress), May 2013.
- [5] Bryant, S., Filsfils, C., Previdi, S., and M. Shand, "IP Fast Reroute using tunnels", draft-bryant-ipfrr-tunnels-03 (work in progress), November 2007.

Authors' Addresses

Pierre Francois
Institute IMDEA Networks
Leganes
ES

Email: pierre.francois@imdea.org

Clarence Filsfils
Cisco Systems, Inc.
Brussels
BE

Email: cfilsfil@cisco.com

Ahmed Bashandy
Cisco Systems, Inc.
San Jose
US

Email: bashandy@cisco.com

Bruno Decraene
Orange
Issy-les-Moulineaux
FR

Email: bruno.decraene@orange.com

Stephane Litkowski
Orange
FR

Email: bruno.decraene@orange.com

RTGWG
Internet-Draft
Intended status: Informational
Expires: January 16, 2014

S. Ning
Tata Communications
D. McDysan
Verizon
E. Osborne
Cisco
L. Yong
Huawei USA
C. Villamizar
Outer Cape Cod Network
Consulting
July 15, 2013

Advanced Multipath Framework in MPLS
draft-ietf-rtgwg-cl-framework-04

Abstract

This document specifies a framework for support of Advanced Multipath in MPLS networks. As defined in this framework, an Advanced Multipath consists of a group of homogenous or non-homogenous links that have the same forward adjacency (FA) and can be considered as a single TE link or an IP link when advertised into IGP routing.

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on January 16, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	4
1.1.	Background	4
1.2.	Architecture Summary	4
1.3.	Conventions used in this document	5
1.4.	Terminology	5
1.5.	Document Issues	5
2.	Advanced Multipath Key Characteristics	7
2.1.	Flow Identification	7
2.1.1.	Flow Identification Granularity	8
2.1.2.	Flow Identification Summary	9
2.1.3.	Flow Identification Using Entropy Label	9
2.2.	Advanced Multipath in Control Plane	10
2.3.	Advanced Multipath in Data Plane	13
3.	Architecture Tradeoffs	14
3.1.	Scalability Motivations	14
3.2.	Reducing Routing Information and Exchange	15
3.3.	Reducing Signaling Load	15
3.3.1.	Reducing Signaling Load using LDP MPTP	16
3.3.2.	Reducing Signaling Load using Hierarchy	16
3.3.3.	Using Both LDP MPTP and RSVP-TE Hierarchy	17
3.4.	Reducing Forwarding State	17
3.5.	Avoiding Route Oscillation	17
4.	New Challenges	18
4.1.	Control Plane Challenges	19
4.1.1.	Delay and Jitter Sensitive Routing	19
4.1.2.	Local Control of Traffic Distribution	20
4.1.3.	Path Symmetry Requirements	20
4.1.4.	Requirements for Contained LSP	21
4.1.5.	Retaining Backwards Compatibility	21
4.2.	Data Plane Challenges	22
4.2.1.	Very Large LSP	22
4.2.2.	Very Large Microflows	23
4.2.3.	Traffic Ordering Constraints	23
4.2.4.	Accounting for IP and LDP Traffic	23
4.2.5.	IP and LDP Limitations	24
5.	Existing Mechanisms	25

5.1. Link Bundling	25
5.2. Classic Multipath	26
6. Mechanisms Proposed in Other Documents	27
6.1. Loss and Delay Measurement	27
6.2. Link Bundle Extensions	28
6.3. Pseudowire Flow and MPLS Entropy Labels	28
6.4. Multipath Extensions	29
7. Required Protocol Extensions and Mechanisms	29
7.1. Brief Review of Requirements	29
7.2. Proposed Document Coverage	30
7.2.1. Component Link Grouping	31
7.2.2. Delay and Jitter Extensions	31
7.2.3. Path Selection and Admission Control	32
7.2.4. Dynamic Multipath Balance	32
7.2.5. Frequency of Load Balance	33
7.2.6. Inter-Layer Communication	33
7.2.7. Packet Ordering Requirements	33
7.2.8. Minimally Disruption Load Balance	34
7.2.9. Path Symmetry	34
7.2.10. Performance, Scalability, and Stability	35
7.2.11. IP and LDP Traffic	35
7.2.12. LDP Extensions	35
7.2.13. Pseudowire Extensions	36
7.2.14. Multi-Domain Advanced Multipath	36
7.3. Framework Requirement Coverage by Protocol	36
7.3.1. OSPF-TE and ISIS-TE Protocol Extensions	37
7.3.2. PW Protocol Extensions	37
7.3.3. LDP Protocol Extensions	37
7.3.4. RSVP-TE Protocol Extensions	37
7.3.5. RSVP-TE Path Selection Changes	37
7.3.6. RSVP-TE Admission Control and Preemption	37
7.3.7. Flow Identification and Traffic Balance	37
8. IANA Considerations	38
9. Security Considerations	38
10. Acknowledgments	38
11. References	39
11.1. Normative References	39
11.2. Informative References	39
Authors' Addresses	42

1. Introduction

Advanced Multipath functional requirements are specified in [I-D.ietf-rtgwg-cl-requirement]. Advanced Multipath use cases are described in [I-D.ietf-rtgwg-cl-use-cases]. This document specifies a framework to meet these requirements.

This document describes an Advanced Multipath framework in the context of MPLS networks using an IGP-TE and RSVP-TE MPLS control plane with GMPLS extensions [RFC3209] [RFC3630] [RFC3945] [RFC5305].

Specific protocol solutions are outside the scope of this document, however a framework for the extension of existing protocols is provided. Backwards compatibility is best achieved by extending existing protocols where practical rather than inventing new protocols. The focus is on examining where existing protocol mechanisms fall short with respect to [I-D.ietf-rtgwg-cl-requirement] and on the types of extensions that will be required to accommodate functionality that is called for in [I-D.ietf-rtgwg-cl-requirement].

1.1. Background

Classic multipath, including Ethernet Link Aggregation has been widely used in today's MPLS networks [RFC4385][RFC4928]. Classic multipath using non-Ethernet links are often advertised using MPLS Link bundling. A link bundle [RFC4201] bundles a group of homogeneous links as a TE link to make IGP-TE information exchange and RSVP-TE signaling more scalable. An Advanced Multipath allows bundling non-homogenous links together as a single logical link.

An Advanced Multipath is a single logical link in MPLS network that contains multiple parallel component links between two MPLS LSR. Unlike a link bundle [RFC4201], the component links in an Advanced Multipath can have different properties such as cost, capacity, delay, or jitter.

1.2. Architecture Summary

Networks aggregate information, both in the control plane and in the data plane, as a means to achieve scalability. A tradeoff exists between the needs of scalability and the needs to identify differing path and link characteristics and differing requirements among flows contained within further aggregated traffic flows. These tradeoffs are discussed in detail in Section 3.

Some aspects of Advanced Multipath requirements present challenges for which multiple solutions may exist. In Section 4 various challenges and potential approaches are discussed.

A subset of the functionality called for in [I-D.ietf-rtgwg-cl-requirement] is available through MPLS Link Bundling [RFC4201]. Link bundling and other existing standards applicable to Advanced Multipath are covered in Section 5.

The most straightforward means of supporting Advanced Multipath requirements is to extend MPLS protocols and protocol semantics and in particular to extend link bundling. Extensions which have already been proposed in other documents which are applicable to Advanced Multipath are discussed in Section 6.

A goal of most new protocol work within IETF is to reuse existing protocol encapsulations and mechanisms where they meet requirements and extend existing mechanisms. This approach minimizes additional complexity while meeting requirements and tends to preserve backwards compatibility to the extent it is practical to do so. These goals are considered in proposing a framework for further protocol extensions and mechanisms in Section 7.

1.3. 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].

1.4. Terminology

Terminology defined in [I-D.ietf-rtgwg-cl-requirement] is used in this document. The additional terms defined in [I-D.ietf-rtgwg-cl-use-cases] are also used.

The abbreviation IGP-TE is used as a shorthand indicating either OSPF-TE [RFC3630] or ISIS-TE [RFC5305].

1.5. Document Issues

This subsection exists solely for the purpose of focusing the RTGWG meeting and mailing list discussions on areas within this document that need attention in order for the document to achieve the level of quality necessary to advance the document through the IETF process. This subsection will be removed before work group last call.

The following issues need to be resolved.

1. The feasibility of symmetric paths for all flows is questionable. The only case where this is practical is where LSP are smaller than component links and where classic link bundling (not using the all-ones component) is used. Perhaps the emphasis on this

(mis)feature should be reduced in the requirements document. See Section 4.1.3.

2. There is a tradeoff between supporting delay optimized routing and avoiding oscillation. This may be sufficiently covered, but a careful review by others and comments would be beneficial.
3. Any measurement of jitter (delay variation) that is used in route decision is likely to cause oscillation. Trying to optimize a path to reduce jitter may be a fools errand. How do we say this in the draft or does the existing text cover it adequately?
4. RTGWG needs to consider the possibility of using multi-topology IGP extensions in IP and LDP routing where the topologies reflect differing requirements (see Section 4.2.5). This idea is similar to TOS routing, which has been discussed for decades but has never been deployed. One possible outcome of discussion would be to declare TOS routing out of scope in the requirements document.
5. The following referenced drafts have expired:
 - A. [I-D.ospf-cc-stlv]
 - B. [I-D.villamizar-mpls-multipath-extn]

A replacement for [I-D.ospf-cc-stlv] is expected to be submitted. [I-D.villamizar-mpls-multipath-extn] is expected to emerge in a simplified form, removing extensions for which existing workarounds are considered adequate based on feedback at a prior IETF.
6. Clarification of what we intend to do with Multi-Domain Advanced Multipath is needed in Section 7.2.14.
7. The following topics in the requirements document are not addressed. Since they are explicitly mentioned in the requirements document some mention of how they are supported is needed in this document.
 - A. Migration (incremental deployment) may not be adequately covered in Section 4.1.5. It might also be necessary to say more here on performance, scalability, and stability as it related to migration. Comments on this from co-authors or the WG?
 - B. We may need a performance section in this document to specifically address #DR6 (fast convergence), and #DR7 (fast worst case failure convergence). We do already have

scalability discussion and make a recommendation for a separate document. At the very least the performance section would have to say "no worse than before, except were there was no alternative to make it very slightly worse" (in a bit more detail than that). It might also be helpful to better define the nature of the performance criteria implied by #DR6 and #DR7.

The above list has been in this document for the better part of a year with very little discussion (or none) of the above issues on the RTGWG mailing list.

2. Advanced Multipath Key Characteristics

[I-D.ietf-rtgwg-cl-requirement] defines external behavior of Advanced Multipath. The overall framework approach involves extending existing protocols in a backwards compatible manner and reusing ongoing work elsewhere in IETF where applicable, defining new protocols or semantics only where necessary. Given the requirements, and this approach of extending MPLS, Advanced Multipath key characteristics can be described in greater detail than given requirements alone.

2.1. Flow Identification

Traffic mapping to component links is a data plane operation. Control over how the mapping is done may be directly dictated or constrained by the control plane or by the management plane. When unconstrained by the control plane or management plane, distribution of traffic is entirely a local matter. Regardless of constraints or lack of constraints, the traffic distribution is required to keep packets belonging to individual flows in sequence and meet QoS criteria specified per LSP by either signaling or management [RFC2475] [RFC3260].

Key objectives of the traffic distribution are to not overload any component link, and to be able to perform local recovery when a subset of component links fails.

The network operator may have other objectives such as placing a bidirectional flow or LSP on the same component link in both direction, bounding delay and/or jitter, Advanced Multipath energy saving, and etc. These new requirements are described in [I-D.ietf-rtgwg-cl-requirement].

Examples of means to identify a flow may in principle include:

1. an LSP identified by an MPLS label,
2. a pseudowire (PW) [RFC3985] identified by an MPLS PW label,
3. a flow or group of flows within a pseudowire (PW) [RFC6391] identified by an MPLS flow label,
4. a flow or flow group in an LSP [RFC6790] identified by an MPLS entropy label,
5. all traffic between a pair of IP hosts, identified by an IP source and destination pair,
6. a specific connection between a pair of IP hosts, identified by an IP source and destination pair, protocol, and protocol port pair,
7. a layer-2 conversation within a pseudowire (PW), where the identification is PW payload type specific, such as Ethernet MAC addresses and VLAN tags within an Ethernet PW [RFC4448]. This is feasible but not practical (see below).

Although in principle a layer-2 conversation within a pseudowire (PW), may be identified by PW payload type specific information, in practice this is impractical at LSP midpoints when PW are carried. The PW ingress may provide equivalent information in a PW flow label [RFC6391]. Therefore, in practice, item #8 above is covered by [RFC6391] and may be dropped from the list.

2.1.1. Flow Identification Granularity

An LSR must at least be capable of identifying flows based on MPLS labels. Most MPLS LSP do not require that traffic carried by the LSP are carried in order. MPLS-TP is a recent exception. If it is assumed that no LSP require strict packet ordering of the LSP itself (only of flows within the LSP), then the entire label stack can be used as flow identification. If some LSP may require strict packet ordering but those LSP cannot be distinguished from others, then only the top label can be used as a flow identifier. If only the top label is used (for example, as specified by [RFC4201] when the "all-ones" component described in [RFC4201] is not used), then there may not be adequate flow granularity to accomplish well balanced traffic distribution and it will not be possible to carry LSP that are larger than any individual component link.

The number of flows can be extremely large. This may be the case when the entire label stack is used and is always the case when IP addresses are used in provider networks carrying Internet traffic.

Current practice for native IP load balancing at the time of writing were documented in [RFC2991] and [RFC2992]. These practices as described, make use of IP addresses.

The common practices described in [RFC2991] and [RFC2992] were extended to include the MPLS label stack and the common practice of looking at IP addresses within the MPLS payload. These extended practices require that pseudowires use a PWE3 Control Word and are described in [RFC4385] and [RFC4928]. Additional detail on current multipath practices can be found in the appendices of [I-D.ietf-rtgwg-cl-use-cases].

Using only the top label supports too coarse a traffic balance. Prior to MPLS Entropy Label [RFC6790] using the full label stack was also too coarse. Using the full label stack and IP addresses as flow identification provides a sufficiently fine traffic balance, but is capable of identifying such a high number of distinct flows, that a technique of grouping flows, such as hashing on the flow identification criteria, becomes essential to reduce the stored state, and is an essential scaling technique. Other means of grouping flows may be possible.

2.1.2. Flow Identification Summary

In summary:

1. Load balancing using only the MPLS label stack provides too coarse a granularity of load balance.
2. Tracking every flow is not scalable due to the extremely large number of flows in provider networks.
3. Existing techniques, IP source and destination hash in particular, have proven in over two decades of experience to be an excellent way of identifying groups of flows.
4. If a better way to identify groups of flows is discovered, then that method can be used.
5. IP address hashing is not required, but use of this technique is strongly encouraged given the technique's long history of successful deployment.

2.1.3. Flow Identification Using Entropy Label

MPLS Entropy Label [RFC6790] provides a means of making use of the entropy from information that would require deeper packet inspection, such as inspection of IP addresses, and putting that entropy in the

form of a hashed value into the label stack. Midpoint LSR that understand the Entropy Label Indicator can make use of only label stack information but still obtain a fine load balance granularity.

2.2. Advanced Multipath in Control Plane

An Advanced Multipath is advertised as a single logical interface between two connected routers, which forms forwarding adjacency (FA) between the routers. The FA is advertised as a TE-link in a link state IGP, using either OSPF-TE or ISIS-TE. The IGP-TE advertised interface parameters for the Advanced Multipath can be preconfigured by the network operator or be derived from its component links. Advanced Multipath advertisement requirements are specified in [I-D.ietf-rtgwg-cl-requirement].

In IGP-TE, an Advanced Multipath is advertised as a single TE link between two connected routers. This is similar to a link bundle [RFC4201]. Link bundle applies to a set of homogenous component links. Advanced Multipath allows homogenous and non-homogenous component links. Due to the similarity, and for backwards compatibility, extending link bundling is viewed as both simple and as the best approach.

In order for a route computation engine to calculate a proper path for a LSP, it is necessary for Advanced Multipath to advertise the summarized available bandwidth as well as the maximum bandwidth that can be made available for single flow (or single LSP where no finer flow identification is available). If an Advanced Multipath contains some non-homogeneous component links, the Advanced Multipath also should advertise the summarized bandwidth and the maximum bandwidth for single flow per each homogeneous component link group.

Both LDP [RFC5036] and RSVP-TE [RFC3209] can be used to signal a LSP over an Advanced Multipath. LDP cannot be extended to support traffic engineering capabilities [RFC3468].

When an LSP is signaled using RSVP-TE, the LSP MUST be placed on the component link that meets the LSP criteria indicated in the signaling message.

When an LSP is signaled using LDP, the LSP MUST be placed on the component link that meets the LSP criteria, if such a component link is available. LDP does not support traffic engineering capabilities, imposing restrictions on LDP use of Advanced Multipath. See Section 4.2.5 for further details.

If the Advanced Multipath solution is based on extensions to IGP-TE and RSVP-TE, then in order to meet requirements defined in

[I-D.ietf-rtgwg-cl-requirement], the following derived requirements MUST be met.

1. An Advanced Multipath MAY contain non-homogeneous component links. The route computing engine MAY select one group of component links for a LSP. The The route computing engine MUST accommodate service objectives for a given LSP when selecting a group of component links for a LSP.
2. The routing protocol MUST make a grouping of component links available in the TE-LSDB, such that within each group all of the component links have similar characteristics (the component links are homogeneous within a group).
3. The route computation used in RSVP-TE MUST be extended to include only the capacity of groups within an Advanced Multipath which meet LSP criteria.
4. The signaling protocol MUST be able to indicate either the criteria, or which groups may be used.
5. An Advanced Multipath MUST place each LSP on a component link or group which meets or exceeds the LSP criteria.

Advanced Multipath capacity is aggregated capacity. LSP capacity MAY be larger than individual component link capacity. Any aggregated LSP can determine a bounds on the largest microflow that could be carried and this constraint can be handled as follows.

1. If no information is available through signaling, management plane, or configuration, the largest microflow is bound by one of the following:
 - A. the largest single LSP if most traffic is RSVP-TE signaled and further aggregated,
 - B. the largest pseudowire if most traffic is carrying pseudowire payloads that are aggregated within RSVP-TE LSP,
 - C. or the largest interface or component link capacity carrying IP or LDP if a large amount of IP or LDP traffic is contained within the aggregate.

If a very large amount of traffic being aggregated is IP or LDP, then the largest microflow is bound by the largest component link on which IP traffic can arrive. For example, if an LSR is acting as an LER and IP and LDP traffic is arriving on 10 Gb/s edge interfaces, then no microflow larger than 10 Gb/s will be present

on the RSVP-TE LSP that aggregate traffic across the core, even if the core interfaces are 100 Gb/s interfaces.

2. The prior conditions provide a bound on the largest microflow when no signaling extensions indicate a bounds. If an LSP is aggregating smaller LSP for which the largest expected microflow carried by the smaller LSP is signaled, then the largest microflow expected in the containing LSP (the aggregate) is the maximum of the largest expected microflow for any contained LSP. For example, RSVP-TE LSP may be large but aggregate traffic for which the source or sink are all 1 Gb/s or smaller interfaces (such as in mobile applications in which cell sites backhauls are no larger than 1 Gb/s). If this information is carried in the LSP originated at the cell sites, then further aggregates across a core may make use of this information.
3. The IGP must provide the bounds on the largest microflow that an Advanced Multipath can accommodate, which is the maximum capacity on a component link that can be made available by moving other traffic. This information is needed by the ingress LER for path determination.
4. A means to signal an LSP whose capacity is larger than individual component link capacity is needed [I-D.ietf-rtgwg-cl-requirement] and also signal the largest microflow expected to be contained in the LSP. If a bounds on the largest microflow is not signaled there is no means to determine if an LSP which is larger than any component link can be subdivided into flows and therefore should be accepted by admission control.

When a bidirectional LSP request is signaled over an Advanced Multipath, if the request indicates that the LSP must be placed on the same component link, the routers of the Advanced Multipath MUST place the LSP traffic in both directions on a same component link. This is particularly challenging for aggregated capacity which makes use of the label stack for traffic distribution. The two requirements are mutually exclusive for any one LSP. No one LSP may be both larger than any individual component link and require symmetrical paths for every flow. Both requirements can be accommodated by the same Advanced Multipath for different LSP, with any one LSP requiring no more than one of these two features.

Individual component link may fail independently. Upon component link failure, an Advanced Multipath MUST support a minimally disruptive local repair, preempting any LSP which can no longer be supported. Available capacity in other component links MUST be used to carry impacted traffic. The available bandwidth after failure MUST be advertised immediately to avoid looped crankback.

When an Advanced Multipath is not able to transport all flows, it preempts some flows based upon holding priority and informs the control plane of these preempted flows. To minimize impact on traffic, the Advanced Multipath MUST support soft preemption [RFC5712]. The network operator SHOULD enable soft preemption. This action ensures the remaining traffic is transported properly. FR#10 requires that the traffic be restored. FR#12 requires that any change be minimally disruptive. These two requirements are interpreted to include preemption among the types of changes that must be minimally disruptive.

2.3. Advanced Multipath in Data Plane

The data plane must identify groups of flows. Flow identification is covered in Section 2.1. Having identified groups of flows the groups must be placed on individual component links. This step following flow group identification is called traffic distribution or traffic placement. The two steps together are known as traffic balancing or load balancing.

Traffic distribution may be determined by or constrained by control plane or management plane. Traffic distribution may be changed due to component link status change, subject to constraints imposed by either the management plane or control plane. The distribution function is local to the routers in which an Advanced Multipath belongs to and its implementation is not specified here.

When performing traffic placement, an Advanced Multipath does not differentiate multicast traffic vs. unicast traffic.

In order to maintain scalability, existing data plane forwarding retains state associated with the top label only. Using UHP (UHP is the absence of the more common PHP), zero or more labels may be POPed and packet and byte counters incremented prior to processing what becomes the top label after the POP operations are completed. Flow group identification may be a parallel step in the forwarding process. Data plane forwarding makes use of the top label to select an Advanced Multipath, or a group of components within an Advanced Multipath or for the case where an LSP is pinned (see [RFC4201]), a specific component link. For those LSP for which the LSP selects only the Advanced Multipath or a group of components within an Advanced Multipath, the load balancing makes use of the set of component links selected based on the top label, and makes use of the flow group identification to select among that group.

The simplest traffic placement techniques uses a modulo operation after computing a hash. This techniques has significant disadvantages. The most common traffic placement techniques uses the

a flow group identification as an index into a table. The table provides an indirection. The number of bits of hash is constrained to keep table size small. While this is not the best technique, it is the most common. Better techniques exist but they are outside the scope of this document and some are considered proprietary.

Requirements to limit frequency of load balancing can be adhered to by keeping track of when a flow group was last moved and imposing a minimum period before that flow group can be moved again. This is straightforward for a table approach. For other approaches it may be less straightforward.

3. Architecture Tradeoffs

Scalability and stability are critical considerations in protocol design where protocols may be used in a large network such as today's service provider networks. Advanced Multipath is applicable to networks which are large enough to require that traffic be split over multiple paths. Scalability is a major consideration for networks that reach a capacity large enough to require Advanced Multipath.

Some of the requirements of Advanced Multipath could potentially have a negative impact on scalability. This section is about architectural tradeoffs, many motivated by the need to maintain scalability and stability, a need which is reflected in [I-D.ietf-rtgwg-cl-requirement], specifically in DR#6 and DR#7.

3.1. Scalability Motivations

In the interest of scalability, information is aggregated in situations where information about a large amount of network capacity or a large amount of network demand provides is adequate to meet requirements. Routing information is aggregated to reduce the amount of information exchange related to routing and to simplify route computation (see Section 3.2).

In an MPLS network large routing changes can occur when a single fault occurs. For example, a single fault may impact a very large number of LSP traversing a given link. As new LSP are signaled to avoid the fault, resources are consumed elsewhere, and routing protocol announcements must flood the resource changes. If protection is in place, there is less urgency to converging quickly. If multiple faults occur that are not covered by shared risk groups (SRG), then some protection may fail, adding urgency to converging quickly even where protection is deployed.

Reducing the amount of information allows the exchange of information

during a large routing change to be accomplished more quickly and simplifies route computation. Simplifying route computation improves convergence time after very significant network faults which cannot be handled by preprovisioned or precomputed protection mechanisms. Aggregating smaller LSP into larger LSP is a means to reduce path computation load and reduce RSVP-TE signaling (see Section 3.3).

Neglecting scaling issues can result in performance issues, such as slow convergence. Neglecting scaling in some cases can result in networks which perform so poorly as to become unstable.

3.2. Reducing Routing Information and Exchange

Link bundling provides a means of aggregating control plane information. Even where the all-ones component link supported by link bundling is not used, the amount of control information is reduced by the number of component links in a bundle.

Fully deaggregating link bundle information would negate this benefit. If there is a need to deaggregate, such as to distinguish between groups of links within specified ranges of delay, then no more deaggregation than is necessary should be done.

For example, in supporting the requirement for heterogeneous component links, it makes little sense to fully deaggregate link bundles when adding support for groups of component links with common attributes within a link bundle can maintain most of the benefit of aggregation while adequately supporting the requirement to support heterogeneous component links.

Routing information exchange is also reduced by making sensible choices regarding the amount of change to link parameters that require link readvertisement. For example, if delay measurements include queuing delay, then a much more coarse granularity of delay measurement would be called for than if the delay does not include queuing and is dominated by geographic delay (speed of light delay).

3.3. Reducing Signaling Load

Aggregating traffic into very large hierarchical LSP in the core very substantially reduces the number of LSP that need to be signaled and the number of path computations any given LSR will be required to perform when a network fault occurs.

In the extreme, applying MPLS to a very large network without hierarchy could exceed the 20 bit label space. For example, in a network with 4,000 nodes, with 2,000 on either side of a cutset, would have 4,000,000 LSP crossing the cutset. Even in a degree four

cutset, an uneven distribution of LSP across the cutset, or the loss of one link would result in a need to exceed the size of the label space. Among provider networks, 4,000 access nodes is not at all large. Hierarchy is an absolute requirement if all access nodes were interconnected in such a network.

In less extreme cases, having each node terminate hundreds of LSP to achieve a full mesh creates a very large computational load. Computational complexity is a function of the number of nodes (N) and links (L) in a topology, and the number of LSP that need to be set up. In the common case where L is proportional to N (relatively constant node degree with growth), the time complexity of one CSPF computation is order($N \log N$). If each node must perform order(N) computations when a fault occurs, then the computational load increases as order($N^2 \log N$) as the number of nodes increases (where " \wedge " is the power of operator and " N^2 " is read "N-squared"). In practice at the time of writing, this imposes a limit of a few hundred nodes in a full mesh of MPLS LSP before the computational load is sufficient to result in unacceptable convergence times.

Two solutions are applied to reduce the amount of RSVP-TE signaling. Both involve subdividing the MPLS domain into a core and a set of regions.

3.3.1. Reducing Signaling Load using LDP MPTP

LDP can be used for edge-to-edge LSP, using RSVP-TE to carry the LDP intra-core traffic and also optionally also using RSVP-TE to carry the LDP intra-region traffic within each region. LDP does not support traffic engineering, but does support multipoint-to-point (MPTP) LSP, which require less signaling than edge-to-edge RSVP-TE point-to-point (PTP) LSP. A drawback of this approach is the inability to use RSVP-TE protection (FRR or GMPLS protection) against failure of the border LSR sitting at a core/region boundary.

3.3.2. Reducing Signaling Load using Hierarchy

When the number of nodes grows too large, the amount of RSVP-TE signaling can be reduced using the MPLS PSC hierarchy [RFC4206]. A core within the hierarchy can divide the topology into M regions of on average N/M nodes. Within a region the computational load is reduced by more than M^2 . Within the core, the computational load generally becomes quite small since M is usually a fairly small number (a few tens of regions) and each region is generally attached to the core in typically only two or three places on average.

Using hierarchy improves scaling but has two consequences. First, hierarchy effectively forces the use of platform label space. When a

containing LSP is rerouted, the labels assigned to the contained LSP cannot be changed but may arrive on a different interface. Second, hierarchy results in much larger LSP. These LSP today are larger than any single component link and therefore force the use of the all-ones component in link bundles.

3.3.3. Using Both LDP MPTP and RSVP-TE Hierarchy

It is also possible to use both LDP and RSVP-TE hierarchy. MPLS networks with a very large number of nodes may benefit from the use of both LDP and RSVP-TE hierarchy. The two techniques are certainly not mutually exclusive.

3.4. Reducing Forwarding State

Both LDP and MPLS hierarchy have the benefit of reducing the amount of forwarding state. Using the example from Section 3.3, and using MPLS hierarchy, the worst case generally occurs at borders with the core.

For example, consider a network with approximately 1,000 nodes divided into 10 regions. At the edges, each node requires 1,000 LSP to other edge nodes. The edge nodes also require 100 intra-region LSP. Within the core, if the core has only 3 attachments to each region the core LSR have less than 100 intra-core LSP. At the border cutset between the core and a given region, in this example there are 100 edge nodes with inter-region LSP crossing that cutset, destined to 900 other edge nodes. That yields forwarding state for on the order of 90,000 LSP at the border cutset. These same routers need only reroute well under 200 LSP when a multiple fault occurs, as long as only links are affected and a border LSR does not go down.

Interior to the core, the forwarding state is greatly reduced. If inter-region LSP have different characteristics, it makes sense to make use of aggregates with different characteristics. Rather than exchange information about every inter-region LSP within the intra-core LSP it makes more sense to use multiple intra-core LSP between pairs of core nodes, each aggregating sets of inter-region LSP with common characteristics or common requirements.

3.5. Avoiding Route Oscillation

Networks can become unstable when a feedback loop exists such that moving traffic to a link causes a metric such as delay to increase, which then causes traffic to move elsewhere. For example, the original ARPANET routing used a delay based cost metric and proved prone to route oscillations [DBP].

Delay may be used as a constraint in routing for high priority traffic, when this high priority traffic makes a minor contribution to total load, such that the movement of the high priority traffic has a small impact on the delay experienced by other high priority traffic. The safest way to measure delay is to make measurements based on traffic which is prioritized such that it is queued ahead of the lower priority traffic which will be affected if high priority traffic is moved. The amount of high priority traffic must be constrained to consume a fraction of link capacities with the remaining capacity available to lower priority traffic.

Any measurement of jitter (delay variation) that is used in route decision is likely to cause oscillation. Jitter that is caused by queuing effects and cannot be measured using a very high priority measurement traffic flow.

It may be possible to find links with constrained queuing delay or jitter using a theoretical maximum or a probability based bound on queuing delay or jitter at a given priority based on the types and amounts of traffic accepted and combining that theoretical limit with a measured delay at very high priority. Using delay or jitter as path metrics without creating oscillations is challenging.

Instability can occur due to poor performance and interaction with protocol timers. In this way a computational scaling problem can become a stability problem when a network becomes sufficiently large.

4. New Challenges

New technical challenges are posed by [I-D.ietf-rtgwg-cl-requirement] in both the control plane and data plane.

Among the more difficult challenges are the following.

1. The requirements related to delay or jitter conflict with requirements for scalability and stability (see Section 4.1.1),
2. The combination of ingress control over LSP placement and retaining an ability to move traffic as demands dictate can pose challenges and such requirements can even be conflicting (see Section 4.1.2),
3. Path symmetry requires extensions and is particularly challenging for very large LSP (see Section 4.1.3),
4. Accommodating a very wide range of requirements among contained LSP can lead to inefficiency if the most stringent requirements

are reflected in aggregates, or reduce scalability if a large number of aggregates are used to provide a too fine a reflection of the requirements in the contained LSP (see Section 4.1.4),

5. Backwards compatibility is somewhat limited due to the need to accommodate legacy multipath interfaces which provide too little information regarding their configured default behavior, and legacy LSP which provide too little information regarding their LSP requirements (see Section 4.1.5),
6. Data plane challenges include those of accommodating very large LSP, large microflows, traffic ordering constraints imposed by a subset of LSP, and accounting for IP and LDP traffic (see Section 4.2).

4.1. Control Plane Challenges

Some of the control plane requirements are particularly challenging. Handling large flows which aggregate smaller flows must be accomplished with minimal impact on scalability. Potentially conflicting are requirements for jitter and requirements for stability. Potentially conflicting are the requirements for ingress control of a large number of parameters, and the requirements for local control needed to achieve traffic balance across an Advanced Multipath. These challenges and potential solutions are discussed in the following sections.

4.1.1. Delay and Jitter Sensitive Routing

Delay and jitter sensitive routing are called for in [I-D.ietf-rtgwg-cl-requirement] in requirements FR#2, FR#7, FR#8, FR#9, FR#15, FR#16, FR#17, FR#18. Requirement FR#17 is particularly problematic, calling for constraints on jitter.

A tradeoff exists between scaling benefits of aggregating information, and potential benefits of using a finer granularity in delay reporting. To maintain the scaling benefit, measured link delay for any given Advanced Multipath SHOULD be aggregated into a small number of delay ranges. IGP-TE extensions MUST be provided which advertise the available capacities for each of the selected ranges.

For path selection of delay sensitive LSP, the ingress SHOULD bias link metrics based on available capacity and select a low cost path which meets LSP total path delay criteria. To communicate the requirements of an LSP, the ERO MUST be extended to indicate the per link constraints. To communicate the type of resource used, the RRO SHOULD be extended to carry an identification of the group that is

used to carry the LSP at each link bundle hop.

4.1.2. Local Control of Traffic Distribution

Many requirements in [I-D.ietf-rtgwg-cl-requirement] suggest that a node immediately adjacent to a component link should have a high degree of control over how traffic is distributed, as long as network performance objectives are met. Particularly relevant are FR#18 and FR#19.

The requirements to allow local control are potentially in conflict with requirement FR#21 which gives full control of component link select to the LSP ingress. While supporting this capability is mandatory, use of this feature is optional per LSP.

A given network deployment will have to consider this set of conflicting requirements and make appropriate use of local control of traffic placement and ingress control of traffic placement to best meet network requirements.

4.1.3. Path Symmetry Requirements

Requirement FR#21 in [I-D.ietf-rtgwg-cl-requirement] includes a provision to bind both directions of a bidirectional LSP to the same component. This is easily achieved if the LSP is directly signaled across an Advanced Multipath. This is not as easily achieved if a set of LSP with this requirement are signaled over a large hierarchical LSP which is in turn carried over an Advanced Multipath. The basis for load distribution in such as case is the label stack. The labels in either direction are completely independent.

This could be accommodated if the ingress, egress, and all midpoints of the hierarchical LSP make use of an entropy label in the distribution, and the ingress use a fixed value per contained LSP in the entropy label. A solution for this problem may add complexity with very little benefit. There is little or no true benefit of using symmetrical paths rather than component links of identical characteristics.

Traffic symmetry and large LSP capacity are a second pair of conflicting requirements. Any given LSP can meet one of these two requirements but not both. A given network deployment will have to make appropriate use of each of these features to best meet network requirements.

4.1.4. Requirements for Contained LSP

[I-D.ietf-rtgwg-cl-requirement] calls for new LSP constraints. These constraints include frequency of load balancing rearrangement, delay and jitter, packet ordering constraints, and path symmetry.

When LSP are contained within hierarchical LSP, there is no signaling available at midpoint LSR which identifies the contained LSP let alone providing the set of requirements unique to each contained LSP. Defining extensions to provide this information would severely impact scalability and defeat the purpose of aggregating control information and forwarding information into hierarchical LSP. For the same scalability reasons, not aggregating at all is not a viable option for large networks where scalability and stability problems may occur as a result.

As pointed out in Section 4.1.3, the benefits of supporting symmetric paths among LSP contained within hierarchical LSP may not be sufficient to justify the complexity of supporting this capability.

A scalable solution which accommodates multiple sets of LSP between given pairs of LSR is to provide multiple hierarchical LSP for each given pair of LSR, each hierarchical LSP aggregating LSP with common requirements and a common pair of endpoints. This is a network design technique available to the network operator rather than a protocol extension. This technique can accommodate multiple sets of delay and jitter parameters, multiple sets of frequency of load balancing parameters, multiple sets of packet ordering constraints, etc.

4.1.5. Retaining Backwards Compatibility

Backwards compatibility and support for incremental deployment requires considering the impact of legacy LSR in the role of LSP ingress, and considering the impact of legacy LSR advertising ordinary links, advertising Ethernet LAG as ordinary links, and advertising link bundles.

Legacy LSR in the role of LSP ingress cannot signal requirements which are not supported by their control plane software. The additional capabilities supported by other LSR has no impact on these LSR. These LSR however, being unaware of extensions, may try to make use of scarce resources which support specific requirements such as low delay. To a limited extent it may be possible for a network operator to avoid this issue using existing mechanisms such as link administrative attributes and attribute affinities [RFC3209].

Legacy LSR advertising ordinary links will not advertise attributes

needed by some LSP. For example, there is no way to determine the delay or jitter characteristics of such a link. Legacy LSR advertising Ethernet LAG pose additional problems. There is no way to determine that packet ordering constraints would be violated for LSP with strict packet ordering constraints, or that frequency of load balancing rearrangement constraints might be violated.

Legacy LSR advertising link bundles have no way to advertise the configured default behavior of the link bundle. Some link bundles may be configured to place each LSP on a single component link and therefore may not be able to accommodate an LSP which requires bandwidth in excess of the size of a component link. Some link bundles may be configured to spread all LSP over the all-ones component. For LSR using the all-ones component link, there is no documented procedure for correctly setting the "Maximum LSP Bandwidth". There is currently no way to indicate the largest microflow that could be supported by a link bundle using the all-ones component link.

Having received the RRO, it is possible for an ingress to look for the all-ones component to identify such link bundles after having signaled at least one LSP. Whether any LSR collects this information on legacy LSR and makes use of it to set defaults, is an implementation choice.

4.2. Data Plane Challenges

Flow identification is briefly discussed in Section 2.1. Traffic distribution is briefly discussed in Section 2.3. This section discusses issues specific to particular requirements specified in [I-D.ietf-rtgwg-cl-requirement].

4.2.1. Very Large LSP

Very large LSP may exceed the capacity of any single component of an Advanced Multipath. In some cases contained LSP may exceed the capacity of any single component. These LSP may make use of the equivalent of the all-ones component of a link bundle, or may use a subset of components which meet the LSP requirements.

Very large LSP can be accommodated as long as they can be subdivided (see Section 4.2.2). A very large LSP cannot have a requirement for symmetric paths unless complex protocol extensions are proposed (see Section 2.2 and Section 4.1.3).

4.2.2. Very Large Microflows

Within a very large LSP there may be very large microflows. A very large microflow is one which cannot be further subdivided and contributes a very large amount of capacity. Flows which cannot be subdivided must be no larger than the capacity of any single component link.

Current signaling provides no way to specify the largest microflow that can be supported on a given link bundle in routing advertisements. Extensions which address this are discussed in Section 6.4. Absent extensions of this type, traffic containing microflows that are too large for a given Advanced Multipath may be present. There is no data plane solution for this problem that would not require reordering traffic at the Advanced Multipath egress.

Some techniques are susceptible to statistical collisions where an algorithm to distribute traffic is unable to disambiguate traffic among two or more very large microflows where their sum is in excess of the capacity of any single component. Hash based algorithms which use too small a hash space are particularly susceptible and require a change in hash seed in the event that this were to occur. A change in hash seed is highly disruptive, causing traffic reordering among all traffic flows over which the hash function is applied.

4.2.3. Traffic Ordering Constraints

Some LSPs have strict traffic ordering constraints. Most notable among these are MPLS-TP LSPs. In the absence of aggregation into hierarchical LSPs, those LSPs with strict traffic ordering constraints can be placed on individual component links if there is a means of identifying which LSPs have such a constraint. If LSPs with strict traffic ordering constraints are aggregated in hierarchical LSPs, the hierarchical LSP capacity may exceed the capacity of any single component link. In such a case the load balancing may be constrained through the use of an entropy label [RFC6790]. This and related issues are discussed further in Section 6.4.

4.2.4. Accounting for IP and LDP Traffic

Networks which carry RSVP-TE signaled MPLS traffic generally carry low volumes of native IP traffic, often only carrying control traffic as native IP. There is no architectural guarantee of this, it is just how network operators have made use of the protocols.

[I-D.ietf-rtgwg-cl-requirement] requires that native IP and native LDP be accommodated (DR#2 and DR#3). In some networks, a subset of services may be carried as native IP or carried as native LDP. Today

this may be accommodated by the network operator estimating the contribution of IP and LDP and configuring a lower set of available bandwidth figures on the RSVP-TE advertisements.

The only improvement that Advanced Multipath can offer is that of measuring the IP and LDP traffic levels and automatically reducing the available bandwidth figures on the RSVP-TE advertisements. The measurements would have to be filtered. This is similar to a feature in existing LSR, commonly known as "autobandwidth" with a key difference. In the "autobandwidth" feature, the bandwidth request of an RSVP-TE signaled LSP is adjusted in response to traffic measurements. In this case the IP or LDP traffic measurements are used to reduce the link bandwidth directly, without first encapsulating in an RSVP-TE LSP.

This may be a subtle and perhaps even a meaningless distinction if Advanced Multipath is used to form a Sub-Path Maintenance Element (SPME). A SPME is in practice essentially an unsignaled single hop LSP with PHP enabled [RFC5921]. An Advanced Multipath SPME looks very much like classic multipath, where there is no signaling, only management plane configuration creating the multipath entity (of which Ethernet Link Aggregation is a subset).

4.2.5. IP and LDP Limitations

IP does not offer traffic engineering. LDP cannot be extended to offer traffic engineering [RFC3468]. Therefore there is no traffic engineered fallback to an alternate path for IP and LDP traffic if resources are not adequate for the IP and/or LDP traffic alone on a given link in the primary path. The only option for IP and LDP would be to declare the link down. Declaring a link down due to resource exhaustion would reduce traffic to zero and eliminate the resource exhaustion. This would cause oscillations and is therefore not a viable solution.

Congestion caused by IP or LDP traffic loads is a pathologic case that can occur if IP and/or LDP are carried natively and there is a high volume of IP or LDP traffic. This situation can be avoided by carrying IP and LDP within RSVP-TE LSP.

It is also not possible to route LDP traffic differently for different FEC. LDP traffic engineering is specifically disallowed by [RFC3468]. It may be possible to support multi-topology IGP extensions to accommodate more than one set of criteria. If so, the additional IGP could be bound to the forwarding criteria, and the LDP FEC bound to a specific IGP instance, inheriting the forwarding criteria. Alternately, one IGP instance can be used and the LDP SPF can make use of the constraints, such as delay and jitter, for a

given LDP FEC.

5. Existing Mechanisms

In MPLS the one mechanism which supports explicit signaling of multiple parallel links is Link Bundling [RFC4201]. The set of techniques known as "classis multipath" support no explicit signaling, except in two cases. In Ethernet Link Aggregation the Link Aggregation Control Protocol (LACP) coordinates the addition or removal of members from an Ethernet Link Aggregation Group (LAG). The use of the "all-ones" component of a link bundle indicates use of classis multipath, however the ability to determine if a link bundle makes use of classis multipath is not yet supported.

5.1. Link Bundling

Link bundling supports advertisement of a set of homogenous links as a single route advertisement. Link bundling supports placement of an LSP on any single component link, or supports placement of an LSP on the all-ones component link. Not all link bundling implementations support the all-ones component link. There is no way for an ingress LSR to tell which potential midpoint LSR support this feature and use it by default and which do not. Based on [RFC4201] it is unclear how to advertise a link bundle for which the all-ones component link is available and used by default. Common practice is to violate the specification and set the Maximum LSP Bandwidth to the Available Bandwidth. There is no means to determine the largest microflow that could be supported by a link bundle that is using the all-ones component link.

[RFC6107] extends the procedures for hierarchical LSP but also extends link bundles. An LSP can be explicitly signaled to indicate that it is an LSP to be used as a component of a link bundle. Prior to that the common practice was to simply not advertise the component link LSP into the IGP, since only the ingress and egress of the link bundle needed to be aware of their existence, which they would be aware of due to the RSVP-TE signaling used in setting up the component LSP.

While link bundling can be the basis for Advanced Multipath, a significant number of small extension needs to be added.

1. To support link bundles of heterogeneous links, a means of advertising the capacity available within a group of homogeneous links needs to be provided.

2. Attributes need to be defined to support the following parameters for the link bundle or for a group of homogeneous links.
 - A. delay range
 - B. jitter (delay variation) range
 - C. group metric
 - D. all-ones component capable
 - E. capable of dynamically balancing load
 - F. largest supportable microflow
 - G. support for entropy label
3. For each of the prior extended attributes, the constraint based routing path selection needs to be extended to reflect new constraints based on the extended attributes.
4. For each of the prior extended attributes, LSP admission control needs to be extended to reflect new constraints based on the extended attributes.
5. Dynamic load balance must be provided for flows within a given set of links with common attributes such that Performance Objectives are not violated including frequency of load balance adjustment for any given flow.

5.2. Classic Multipath

Classic multipath is described in [I-D.ietf-rtgwg-cl-use-cases].

Classic multipath refers to the most common current practice in implementation and deployment of multipath. The most common current practice makes use of a hash on the MPLS label stack and if IPv4 or IPv6 are indicated under the label stack, makes use of the IP source and destination addresses [RFC4385] [RFC4928].

Classic multipath provides a highly scalable means of load balancing. Dynamic multipath has proven value in assuring an even loading on component link and an ability to adapt to change in offered load that occurs over periods of hundreds of milliseconds or more. Classic multipath scalability is due to the ability to effectively work with an extremely large number of flows (IP host pairs) using relatively little resources (a data structure accessed using a hash result as a key or using ranges of hash results).

Classic multipath meets a small subset of Advanced Multipath requirements. Due to scalability of the approach, classic multipath seems to be an excellent candidate for extension to meet the full set of Advanced Multipath forwarding requirements.

Additional detail can be found in [I-D.ietf-rtgwg-cl-use-cases].

6. Mechanisms Proposed in Other Documents

A number of documents which at the time of writing are works in progress address parts of the requirements of Advanced Multipath, or assist in making some of the goals achievable.

6.1. Loss and Delay Measurement

Procedures for measuring loss and delay are provided in [RFC6374]. These are OAM based measurements. This work could be the basis of delay measurements and delay variation measurement used for metrics called for in [I-D.ietf-rtgwg-cl-requirement].

Currently there are three documents that address delay and delay variation metrics.

draft-ietf-ospf-te-metric-extensions

[I-D.ietf-ospf-te-metric-extensions] provides a set of OSPF-TE extension to support delay, jitter, and loss. Stability is not adequately addressed and some minor issues remain.

I-D.previdi-isis-te-metric-extensions

[I-D.previdi-isis-te-metric-extensions] provides the set of extensions for ISIS that [I-D.ietf-ospf-te-metric-extensions] provides for OSPF. This draft mirrors [I-D.ietf-ospf-te-metric-extensions] sometimes lagging for a brief period when the OSPF version is updated.

I-D.atlas-mpls-te-express-path

[I-D.atlas-mpls-te-express-path] provides information on the use of OSPF and ISIS extensions defined in [I-D.ietf-ospf-te-metric-extensions] and [I-D.previdi-isis-te-metric-extensions] and a modified CSPF path selection to meet LSP performance criteria such as minimal delay paths or bounded delay paths.

Delay variance, loss, residual bandwidth, and available bandwidth extensions are particular prone to network instability. The question as to whether queuing delay and delay variation should be considered, and if so for which diffserv Per-Hop Service Class (PSC) is not

adequately addressed in the current versions of these drafts. These drafts are actively being discussed and updated and remaining issues are expected to be resolved.

6.2. Link Bundle Extensions

A set of extension are needed to indicate a group of component links in the ERO or RRO, where the group is given an interface identification like the bundle itself. The extensions could also be further extended to support specification of the all-ones component link in the ERO or RRO.

[I-D.ospf-cc-stlv] provides a baseline draft for extending link bundling to advertise components. A new component TLV (C-TLV) is proposed, which must reference an Advanced Multipath Link TLV. [I-D.ospf-cc-stlv] is intended for the OSPF WG and submitted for the "Experimental" track. The 00 version expired in February 2012. A replacement is expected that will be submitted for consideration on the standards track.

6.3. Pseudowire Flow and MPLS Entropy Labels

Two documents provide a means to add entropy for the purpose of improving load balance. MPLS encapsulation can bury information that is needed to identify microflows. These two documents allow a pseudowire ingress and LSP ingress respectively to add a label solely for the purpose of providing a finer granularity of microflow groups.

[RFC6391] allows pseudowires which carry a large volume of traffic, where microflows can be identified to be load balanced across multiple members of an Ethernet LAG or an MPLS link bundle. This is accomplished by adding a flow label below the pseudowire label in the MPLS label stack. For this to be effective the link bundle load balance must make use of the label stack up to and including this flow label.

[RFC6790] provides a means for a LER to put an additional label known as an entropy label on the MPLS label stack. Only the LER can add the entropy label. The LER of a PSC LSP would have to add a entropy label for contained LSPs for which it is a midpoint LSR.

Core LSR acting as LER for aggregated LSP can add entropy labels based on deep packet inspection and place an entropy label indicator (ELI) and entropy label (EL) just below the label being acted on. This would be helpful in situations where the label stack depth to which load distribution can operate is limited by implementation or is limited for other reasons such as carrying both MPLS-TP and MPLS with entropy labels within the same hierarchical LSP.

6.4. Multipath Extensions

The multipath extensions drafts address the issue of accommodating LSP which have strict packet ordering constraints in a network containing multipath. MPLS-TP has become the one important instance of LSP with strict packet ordering constraints and has driven this work.

[I-D.ietf-mpls-multipath-use] proposed to use MPLS Entropy Label [RFC6790] to allow MPLS-TP to be carried within MPLS LSP that make use of multipath. Limitations of this approach in the absence of protocol extensions is discussed.

[I-D.villamizar-mpls-multipath-extn] provides protocol extensions needed to overcome the limitations in the absence of protocol extensions is discussed in [I-D.ietf-mpls-multipath-use].

7. Required Protocol Extensions and Mechanisms

Prior sections have reviewed key characteristics, architecture tradeoffs, new challenges, existing mechanisms, and relevant mechanisms proposed in existing new documents.

This section first summarizes and groups requirements specified in [I-D.ietf-rtgwg-cl-requirement] (see Section 7.1). A set of documents coverage groupings are proposed with existing works-in-progress noted where applicable (see Section 7.2). The set of extensions are then grouped by protocol affected as a convenience to implementors (see (see Section 7.3).

7.1. Brief Review of Requirements

The following list provides a categorization of requirements specified in [I-D.ietf-rtgwg-cl-requirement] along with a short phrase indication what topic the requirement covers.

routing information aggregation

FR#1 (routing summarization), FR#20 (Advanced Multipath may be a component of another Advanced Multipath)

restoration speed

FR#2 (restoration speed meeting performance objectives), FR#12 (minimally disruptive load rebalance), DR#6 (fast convergence), DR#7 (fast worst case failure convergence)

load distribution, stability, minimal disruption

FR#3 (automatic load distribution), FR#5 (must not oscillate), FR#11 (dynamic placement of flows), FR#12 (minimally disruptive load rebalance), FR#13 (bounded rearrangement frequency), FR#18 (flow placement must satisfy performance objectives), FR#19 (flow identification finer than per top level LSP), MR#6 (operator initiated flow rebalance)

backward compatibility and migration

FR#4 (smooth incremental deployment), FR#6 (management and diagnostics must continue to function), DR#1 (extend existing protocols), DR#2 (extend LDP, no LDP TE)

delay and delay variation

FR#7 (expose lower layer measured delay), FR#8 (precision of latency reporting), FR#9 (limit latency on per LSP basis), FR#15 (minimum delay path), FR#16 (bounded delay path), FR#17 (bounded jitter path)

admission control, preemption, traffic engineering

FR#10 (admission control, preemption), FR#14 (packet ordering), FR#21 (ingress specification of path), FR#22 (path symmetry), DR#3 (IP and LDP traffic), MR#3 (management specification of path)

single vs multiple domain

DR#4 (IGP extensions allowed within single domain), DR#5 (IGP extensions disallowed in multiple domain case)

general network management

MR#1 (polling, configuration, and notification), MR#2 (activation and de-activation)

path determination, connectivity verification

MR#4 (path trace), MR#5 (connectivity verification)

The above list is not intended as a substitute for

[I-D.ietf-rtgwg-cl-requirement], but rather as a concise grouping and reminder or requirements to serve as a means of more easily determining requirements coverage of a set of protocol documents.

7.2. Proposed Document Coverage

The primary areas where additional protocol extensions and mechanisms are required include the topics described in the following subsections.

There are candidate documents for a subset of the topics below. This

grouping of topics does not require that each topic be addressed by a separate document. In some cases, a document may cover multiple topics, or a specific topic may be addressed as applicable in multiple documents.

7.2.1. Component Link Grouping

An extension to link bundling is needed to specify a group of components with common attributes. This can be a TLV defined within the link bundle that carries the same encapsulations as the link bundle. Two interface indices would be needed for each group.

- a. An index is needed that if included in an ERO would indicate the need to place the LSP on any one component within the group.
- b. A second index is needed that if included in an ERO would indicate the need to balance flows within the LSP across all components of the group. This is equivalent to the "all-ones" component for the entire bundle.

[I-D.ospf-cc-stlv] can be extended to include multipath treatment capabilities. An ISIS solution is also needed. An extension of RSVP-TE signaling is needed to indicate multipath treatment preferences.

If a component group is allowed to support all of the parameters of a link bundle, then a group TE metric would be accommodated. This can be supported with the component TLV (C-TLV) defined in [I-D.ospf-cc-stlv].

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "routing information aggregation" set of requirements. The "restoration speed", "backward compatibility and migration", and "general network management" requirements must also be considered.

7.2.2. Delay and Jitter Extensions

A extension is needed in the IGP-TE advertisement to support delay and delay variation for links, link bundles, and forwarding adjacencies. Whatever mechanism is described must take precautions that insure that route oscillations cannot occur. The following set of drafts address this.

1. [I-D.ietf-ospf-te-metric-extensions]
2. [I-D.previdi-isis-te-metric-extensions]

3. [I-D.atlas-mpls-te-express-path]

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "delay and delay variation" set of requirements. The "restoration speed", "backward compatibility and migration", and "general network management" requirements must also be considered.

7.2.3. Path Selection and Admission Control

Path selection and admission control changes must be documented in each document that proposes a protocol extension that advertises a new capability or parameter that must be supported by changes in path selection and admission control.

It would also be helpful to have an informational document which covers path selection and admission control issues in detail and briefly summarizes and references the set of documents which propose extensions. This document could be advanced in parallel with the protocol extensions.

The primary focus of this document, among the sets of requirements listed in Section 7.1 are the "load distribution, stability, minimal disruption" and "admission control, preemption, traffic engineering" sets of requirements. The "restoration speed" and "path determination, connectivity verification" requirements must also be considered. The "backward compatibility and migration", and "general network management" requirements must also be considered.

7.2.4. Dynamic Multipath Balance

FR#11 explicitly calls for dynamic placement of flows. Load balancing similar to existing dynamic multipath would satisfy this requirement. In implementations where flow identification uses a coarse granularity, the adjustments would have to be equally coarse, in the worst case moving entire LSP. The impact of flow identification granularity and potential dynamic multipath approaches may need to be documented in greater detail than provided here.

The primary focus of this document, among the sets of requirements listed in Section 7.1 are the "restoration speed" and the "load distribution, stability, minimal disruption" sets of requirements. The "path determination, connectivity verification" requirements must also be considered. The "backward compatibility and migration", and "general network management" requirements must also be considered.

7.2.5. Frequency of Load Balance

IGP-TE and RSVP-TE extensions are needed to support frequency of load balancing rearrangement called for in FR#13, and FR#15-FR#17. Constraints are not defined in RSVP-TE, but could be modeled after administrative attribute affinities in RFC3209 and elsewhere.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "load distribution, stability, minimal disruption" set of requirements. The "path determination, connectivity verification" must also be considered. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.6. Inter-Layer Communication

Lower layer to upper layer communication called for in FR#7 and FR#20. Specific parameters, specifically delay and delay variation, need to be addressed. Passing information from a lower non-MPLS layer to an MPLS layer needs to be addressed, though this may largely be generic advice encouraging a coupling of MPLS to lower layer management plane or control plane interfaces. This topic can be addressed in each document proposing a protocol extension, where applicable.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "restoration speed" set of requirements. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.7. Packet Ordering Requirements

A document is needed to define extensions supporting various packet ordering requirements, ranging from requirements to preserve microflow ordering only, to requirements to preserve full LSP ordering (as in MPLS-TP). This is covered by [I-D.ietf-mpls-multipath-use] and [I-D.villamizar-mpls-multipath-extn].

The primary focus of this document, among the sets of requirements listed in Section 7.1 are the "admission control, preemption, traffic engineering" and the "path determination, connectivity verification" sets of requirements. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.8. Minimally Disruption Load Balance

The behavior of hash methods used in classic multipath needs to be described in terms of FR#12 which calls for minimally disruptive load adjustments. For example, reseeding the hash violates FR#12. Using modulo operations is significantly disruptive if a link comes or goes down, as pointed out in [RFC2992]. In addition, backwards compatibility with older hardware needs to be accommodated.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "load distribution, stability, minimal disruption" set of requirements.

7.2.9. Path Symmetry

Protocol extensions are needed to support dynamic load balance as called for to meet FR#22 (path symmetry) and to meet FR#11 (dynamic placement of flows).

Currently path symmetry can only be supported in link bundling if the path is pinned. When a flow is moved both ingress and egress must make the move as close to simultaneously as possible to satisfy FR#22 and FR#12 (minimally disruptive load rebalance). There is currently no protocol to coordinate this move.

If a group of flows are identified using a hash, then the hash must be identical on the pair of LSR at the endpoint, using the same hash seed and with one side swapping source and destination. If the label stack is used, then either the entire label stack must be a special case flow identification, since the set of labels in either direction are not correlated, or the two LSR must conspire to use the same flow identifier. For example, using a common entropy label value, and using only the entropy label in the flow identification would satisfy the forwarding requirement. There is no protocol to indicate special treatment of a label stack within a hierarchical LSP. Adding such an extension may add significant complexity and ultimately may prove unscalable.

The primary focus of this document, among the sets of requirements listed in Section 7.1 are the "load distribution, stability, minimal disruption" and the "admission control, preemption, traffic engineering" sets of requirements. The "backward compatibility and migration" and "general network management" requirements must also be considered. Path symmetry simplifies support for the "path determination, connectivity verification" set of requirements, but with significant complexity added elsewhere.

7.2.10. Performance, Scalability, and Stability

A separate document providing analysis of performance, scalability, and stability impacts of changes may be needed. The topic of traffic adjustment oscillation must also be covered. If sufficient coverage is provided in each document covering a protocol extension, a separate document would not be needed.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "restoration speed" set of requirements. This is not a simple topic and not a topic that is well served by scattering it over multiple documents, therefore it may be best to put this in a separate document and put citations in documents called for in Section 7.2.1, Section 7.2.2, Section 7.2.3, Section 7.2.9, Section 7.2.11, Section 7.2.12, Section 7.2.13, and Section 7.2.14. Citation may also be helpful in Section 7.2.4, and Section 7.2.5.

7.2.11. IP and LDP Traffic

A document is needed to define the use of measurements of native IP and native LDP traffic levels which are then used to reduce link advertised bandwidth amounts.

The primary focus of this document, among the sets of requirements listed in Section 7.1 are the "load distribution, stability, minimal disruption" and the "admission control, preemption, traffic engineering" set of requirements. The "path determination, connectivity verification" must also be considered. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.12. LDP Extensions

Extending LDP is called for in DR#2. LDP can be extended to couple FEC admission control to local resource availability without providing LDP traffic engineering capability. Other LDP extensions such as signaling a bound on microflow size and LDP LSP requirements would provide useful information without providing LDP traffic engineering capability.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "admission control, preemption, traffic engineering" set of requirements. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.13. Pseudowire Extensions

Pseudowire (PW) extensions such as signaling a bound on microflow size and signaling requirements specific to PW would provide useful information. This information can be carried in the PW LDP signaling [RFC3985] and the the PW requirements could then be used in a containing LSP.

The primary focus of this document, among the sets of requirements listed in Section 7.1 is the "admission control, preemption, traffic engineering" set of requirements. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.2.14. Multi-Domain Advanced Multipath

DR#5 calls for Advanced Multipath to span multiple network topologies. Component LSP may already span multiple network topologies, though most often in practice these are LDP signaled. Component LSP which are RSVP-TE signaled may also span multiple network topologies using at least three existing methods (per domain [RFC5152], BRPC [RFC5441], PCE [RFC4655]). When such component links are combined in an Advanced Multipath, the Advanced Multipath spans multiple network topologies. It is not clear in which document this needs to be described or whether this description in the framework is sufficient. The authors and/or the WG may need to discuss this. DR#5 mandates that IGP-TE extension cannot be used. This would disallow the use of [RFC5316] or [RFC5392] in conjunction with [RFC5151].

The primary focus of this document, among the sets of requirements listed in Section 7.1 are "single vs multiple domain" and "admission control, preemption, traffic engineering". The "routing information aggregation" and "load distribution, stability, minimal disruption" requirements need attention due to their use of the IGP in single domain Advanced Multipath. Other requirements such as "delay and delay variation", can more easily be accommodated by carrying metrics within BGP. The "path determination, connectivity verification" requirements need attention due to requirements to restrict disclosure of topology information across domains in multi-domain deployments. The "backward compatibility and migration" and "general network management" requirements must also be considered.

7.3. Framework Requirement Coverage by Protocol

As an aid to implementors, this section summarizes requirement coverage listed in Section 7.2 by protocol or LSR functionality affected.

Some documentation may be purely informational, proposing no changes and proposing usage at most. This includes Section 7.2.3, Section 7.2.8, Section 7.2.10, and Section 7.2.14.

Section 7.2.9 may require a new protocol.

7.3.1. OSPF-TE and ISIS-TE Protocol Extensions

Many of the changes listed in Section 7.2 require IGP-TE changes, though most are small extensions to provide additional information. This set includes Section 7.2.1, Section 7.2.2, Section 7.2.5, Section 7.2.6, and Section 7.2.7. An adjustment to existing advertised parameters is suggested in Section 7.2.11.

7.3.2. PW Protocol Extensions

The only suggestion of pseudowire (PW) extensions is in Section 7.2.13.

7.3.3. LDP Protocol Extensions

Potential LDP extensions are described in Section 7.2.12.

7.3.4. RSVP-TE Protocol Extensions

RSVP-TE protocol extensions are called for in Section 7.2.1, Section 7.2.5, Section 7.2.7, and Section 7.2.9.

7.3.5. RSVP-TE Path Selection Changes

Section 7.2.3 calls for path selection to be addressed in individual documents that require change. These changes would include those proposed in Section 7.2.1, Section 7.2.2, Section 7.2.5, and Section 7.2.7.

7.3.6. RSVP-TE Admission Control and Preemption

When a change is needed to path selection, a corresponding change is needed in admission control. The same set of sections applies: Section 7.2.1, Section 7.2.2, Section 7.2.5, and Section 7.2.7. Some resource changes such as a link delay change might trigger preemption. The rules of preemption remain unchanged, still based on holding priority.

7.3.7. Flow Identification and Traffic Balance

The following describe either the state of the art in flow identification and traffic balance or propose changes: Section 7.2.4,

Section 7.2.5, Section 7.2.7, and Section 7.2.8.

8. IANA Considerations

This is a framework document and therefore does not specify protocol extensions. This memo includes no request to IANA.

9. Security Considerations

The security considerations for MPLS/GMPLS and for MPLS-TP are documented in [RFC5920] and [RFC6941].

The types protocol extensions proposed in this framework document provide additional information about links, forwarding adjacencies, and LSP requirements. The protocol semantics changes described in this framework document propose additional LSP constraints applied at path computation time and at LSP admission at midpoints LSR. The additional information and constraints provide no additional security considerations beyond the security considerations already documented in [RFC5920] and [RFC6941].

10. Acknowledgments

Authors would like to thank Adrian Farrel, Fred Jounay, Yuji Kamite for his extensive comments and suggestions regarding early versions of this document, Ron Bonica, Nabil Bitar, Eric Gray, Lou Berger, and Kireeti Kompella for their reviews of early versions and great suggestions.

Authors would like to thank Iftekhhar Hussain for review and suggestions regarding recent versions of this document.

In the interest of full disclosure of affiliation and in the interest of acknowledging sponsorship, past affiliations of authors are noted. Much of the work done by Ning So occurred while Ning was at Verizon. Much of the work done by Curtis Villamizar occurred while at Infinera. Infinera continues to sponsor this work on a consulting basis.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, September 2003.
- [RFC4201] Kompella, K., Rekhter, Y., and L. Berger, "Link Bundling in MPLS Traffic Engineering (TE)", RFC 4201, October 2005.
- [RFC4206] Kompella, K. and Y. Rekhter, "Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)", RFC 4206, October 2005.
- [RFC5036] Andersson, L., Minei, I., and B. Thomas, "LDP Specification", RFC 5036, October 2007.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.
- [RFC5712] Meyer, M. and JP. Vasseur, "MPLS Traffic Engineering Soft Preemption", RFC 5712, January 2010.
- [RFC6107] Shiomoto, K. and A. Farrel, "Procedures for Dynamically Signaled Hierarchical Label Switched Paths", RFC 6107, February 2011.
- [RFC6374] Frost, D. and S. Bryant, "Packet Loss and Delay Measurement for MPLS Networks", RFC 6374, September 2011.
- [RFC6391] Bryant, S., Filsfils, C., Drafz, U., Kompella, V., Regan, J., and S. Amante, "Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network", RFC 6391, November 2011.

11.2. Informative References

- [DBP] Bertsekas, D., "Dynamic Behavior of Shortest Path Routing Algorithms for Communication Networks", IEEE Trans. Auto. Control 1982.
- [I-D.atlas-mpls-te-express-path]

Atlas, A., Drake, J., Giacalone, S., Ward, D., Previdi, S., and C. Filsfils, "Performance-based Path Selection for Explicitly Routed LSPs", draft-atlas-mpls-te-express-path-02 (work in progress), February 2013.

[I-D.ietf-mpls-multipath-use]

Villamizar, C., "Use of Multipath with MPLS-TP and MPLS", draft-ietf-mpls-multipath-use-00 (work in progress), February 2013.

[I-D.ietf-ospf-te-metric-extensions]

Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", draft-ietf-ospf-te-metric-extensions-04 (work in progress), June 2013.

[I-D.ietf-rtgwg-cl-requirement]

Villamizar, C., McDysan, D., Ning, S., Malis, A., and L. Yong, "Requirements for Advanced Multipath in MPLS Networks", draft-ietf-rtgwg-cl-requirement-11 (work in progress), July 2013.

[I-D.ietf-rtgwg-cl-use-cases]

Ning, S., Malis, A., McDysan, D., Yong, L., and C. Villamizar, "Advanced Multipath Use Cases and Design Considerations", draft-ietf-rtgwg-cl-use-cases-04 (work in progress), July 2013.

[I-D.ospf-cc-stlv]

Osborne, E., "Component and Composite Link Membership in OSPF", draft-ospf-cc-stlv-00 (work in progress), August 2011.

[I-D.previdi-isis-te-metric-extensions]

Previdi, S., Giacalone, S., Ward, D., Drake, J., Atlas, A., and C. Filsfils, "IS-IS Traffic Engineering (TE) Metric Extensions", draft-previdi-isis-te-metric-extensions-03 (work in progress), February 2013.

[I-D.villamizar-mpls-multipath-extn]

Villamizar, C., "Multipath Extensions for MPLS Traffic Engineering", draft-villamizar-mpls-multipath-extn-00 (work in progress), November 2012.

[RFC2475]

Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated

- Services", RFC 2475, December 1998.
- [RFC2991] Thaler, D. and C. Hopps, "Multipath Issues in Unicast and Multicast Next-Hop Selection", RFC 2991, November 2000.
- [RFC2992] Hopps, C., "Analysis of an Equal-Cost Multi-Path Algorithm", RFC 2992, November 2000.
- [RFC3260] Grossman, D., "New Terminology and Clarifications for Diffserv", RFC 3260, April 2002.
- [RFC3468] Andersson, L. and G. Swallow, "The Multiprotocol Label Switching (MPLS) Working Group decision on MPLS signaling protocols", RFC 3468, February 2003.
- [RFC3945] Mannie, E., "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3945, October 2004.
- [RFC3985] Bryant, S. and P. Pate, "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, March 2005.
- [RFC4385] Bryant, S., Swallow, G., Martini, L., and D. McPherson, "Pseudowire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN", RFC 4385, February 2006.
- [RFC4448] Martini, L., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet over MPLS Networks", RFC 4448, April 2006.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.
- [RFC4928] Swallow, G., Bryant, S., and L. Andersson, "Avoiding Equal Cost Multipath Treatment in MPLS Networks", BCP 128, RFC 4928, June 2007.
- [RFC5151] Farrel, A., Ayyangar, A., and JP. Vasseur, "Inter-Domain MPLS and GMPLS Traffic Engineering -- Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 5151, February 2008.
- [RFC5152] Vasseur, JP., Ayyangar, A., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, February 2008.
- [RFC5316] Chen, M., Zhang, R., and X. Duan, "ISIS Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS

Traffic Engineering", RFC 5316, December 2008.

- [RFC5392] Chen, M., Zhang, R., and X. Duan, "OSPF Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS Traffic Engineering", RFC 5392, January 2009.
- [RFC5441] Vasseur, JP., 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, April 2009.
- [RFC5920] Fang, L., "Security Framework for MPLS and GMPLS Networks", RFC 5920, July 2010.
- [RFC5921] Bocci, M., Bryant, S., Frost, D., Levrau, L., and L. Berger, "A Framework for MPLS in Transport Networks", RFC 5921, July 2010.
- [RFC6790] Kompella, K., Drake, J., Amante, S., Henderickx, W., and L. Yong, "The Use of Entropy Labels in MPLS Forwarding", RFC 6790, November 2012.
- [RFC6941] Fang, L., Niven-Jenkins, B., Mansfield, S., and R. Graveman, "MPLS Transport Profile (MPLS-TP) Security Framework", RFC 6941, April 2013.

Authors' Addresses

So Ning
Tata Communications

Email: ning.so@tatacommunications.com

Dave McDysan
Verizon
22001 Loudoun County PKWY
Ashburn, VA 20147
USA

Email: dave.mcdysan@verizon.com

Eric Osborne
Cisco

Email: eosborne@cisco.com

Lucy Yong
Huawei USA
5340 Legacy Dr.
Plano, TX 75025
USA

Phone: +1 469-277-5837
Email: lucy.yong@huawei.com

Curtis Villamizar
Outer Cape Cod Network Consulting

Email: curtis@occnc.com

Routing Area Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 27, 2015

S. Litkowski, Ed.
B. Decraene
Orange
C. Filsfils
K. Raza
Cisco Systems
M. Horneffer
Deutsche Telekom
P. Sarkar
Juniper Networks
June 25, 2015

Operational management of Loop Free Alternates
draft-ietf-rtgwg-lfa-manageability-11

Abstract

Loop Free Alternates (LFA), as defined in RFC 5286 is an IP Fast ReRoute (IP FRR) mechanism enabling traffic protection for IP traffic (and MPLS LDP traffic by extension). Following first deployment experiences, this document provides operational feedback on LFA, highlights some limitations, and proposes a set of refinements to address those limitations. It also proposes required management specifications.

This proposal is also applicable to remote LFA solution.

Requirements Language

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 [RFC2119].

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on December 27, 2015.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Definitions	3
3. Operational issues with default LFA tie breakers	4
3.1. Case 1: PE router protecting failures within core network	4
3.2. Case 2: PE router chosen to protect core failures while P router LFA exists	5
3.3. Case 3: suboptimal P router alternate choice	6
3.4. Case 4: No-transit LFA computing node	7
4. Need for coverage monitoring	8
5. Need for LFA activation granularity	9
6. Configuration requirements	9
6.1. LFA enabling/disabling scope	10
6.2. Policy based LFA selection	10
6.2.1. Connected vs remote alternates	11
6.2.2. Mandatory criteria	12
6.2.3. Additional criteria	12
6.2.4. Criteria evaluation	12
6.2.5. Retrieving alternate path attributes	16
6.2.6. ECMP LFAs	22
7. Operational aspects	23
7.1. No-transit condition on LFA computing node	23
7.2. Manual triggering of FRR	24
7.3. Required local information	25
7.4. Coverage monitoring	25
7.5. LFA and network planning	26
8. Security Considerations	26
9. IANA Considerations	27
10. Contributors	27
11. References	27

11.1. Normative References	27
11.2. Informative References	28
Authors' Addresses	29

1. Introduction

Following the first deployments of Loop Free Alternates (LFA), this document provides feedback to the community about the management of LFA.

Section 3 provides real uses cases illustrating some limitations and suboptimal behavior.

Section 4 provides requirements for LFA simulations.

Section 5 proposes requirements for activation granularity and policy based selection of the alternate.

Section 6 express requirements for the operational management of LFA and especially a policy framework to manage alternates.

Section 7 details some operational considerations of LFA like IS-IS overload bit management or troubleshooting informations.

2. Definitions

- o Per-prefix LFA : LFA computation, and best alternate evaluation is done for each destination prefix, as opposed to "Per-next hop" simplification also proposed in [RFC5286] Section 3.8.
- o PE router : Provider Edge router. These routers are connecting customers
- o P router : Provider router. These routers are core routers, without customer connections. They provide transit between PE routers and they form the core network.
- o Core network : subset of the network composed by P routers and links between them.
- o Core link : network link part of the core network i.e. a P router to P router link.
- o Link-protecting LFA : alternate providing protection against link failure.
- o Node-protecting LFA : alternate providing protection against node failure.

- o Connected alternate : alternate adjacent (at IGP level) to the point of local repair (i.e. an IGP neighbor).
- o Remote alternate : alternate which does not share an IGP adjacency with the point of local repair.

3. Operational issues with default LFA tie breakers

[RFC5286] introduces the notion of tie breakers when selecting the LFA among multiple candidate alternate next-hops. When multiple LFA exist, RFC 5286 has favored the selection of the LFA providing the best coverage of the failure cases. While this is indeed a goal, this is one among multiple and in some deployment this lead to the selection of a suboptimal LFA. The following sections details real use cases of such limitations.

Note that the use case of LFA computation per destination (per-prefix LFA) is assumed throughout this analysis. We also assume in the network figures that all IP prefixes are advertised with zero cost.

3.1. Case 1: PE router protecting failures within core network

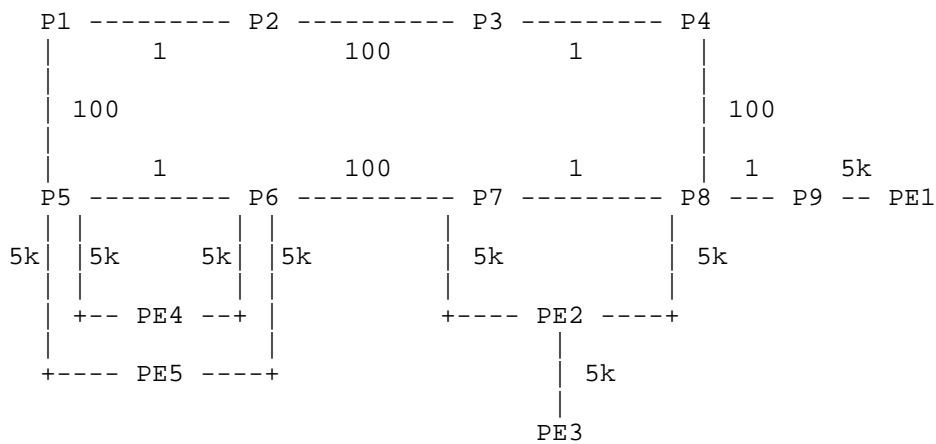


Figure 1

Px routers are P routers using n*10G links. PEs are connected using links with lower bandwidth.

In figure 1, let us consider the traffic flowing from PE1 to PE4. The nominal path is P9-P8-P7-P6-PE4. Let us consider the failure of link P7-P8. As P4 primary path to PE4 is P8-P7-P6-PE4, P4 is not an LFA for P8 (because P4 will loop back traffic to P8) and the only available LFA is PE2.

When the core link P8-P7 fails, P8 switches all traffic destined to PE4/PE5 towards the node PE2. Hence a PE node and PE links are used to protect the failure of a core link. Typically, PE links have less capacity than core links and congestion may occur on PE2 links. Note that although PE2 was not directly affected by the failure, its links become congested and its traffic will suffer from the congestion.

In summary, in case of P8-P7 link failure, the impact on customer traffic is:

- o From PE2 point of view :
 - * without LFA: no impact
 - * with LFA: traffic is partially dropped (but possibly prioritized by a QoS mechanism). It must be highlighted that in such situation, traffic not affected by the failure may be affected by the congestion.
- o From P8 point of view:
 - * without LFA: traffic is totally dropped until convergence occurs.
 - * with LFA: traffic is partially dropped (but possibly prioritized by a QoS mechanism).

Besides the congestion aspects of using an Edge router as an alternate to protect a core failure, a service provider may consider this as a bad routing design and would like to prevent it.

3.2. Case 2: PE router chosen to protect core failures while P router LFA exists

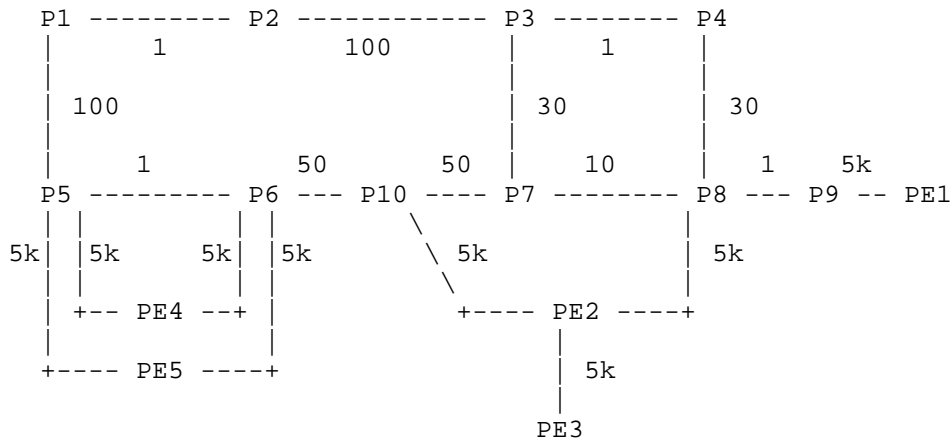


Figure 2

Px routers are P routers meshed with n*10G links. PEs are meshed using links with lower bandwidth.

In the figure 2, let us consider the traffic coming from PE1 to PE4. Nominal path is P9-P8-P7-P10-P6-PE4. Let us consider the failure of the link P7-P8. For P8, P4 is a link-protecting LFA and PE2 is a node-protecting LFA. PE2 is chosen as best LFA due to its better protection type. Just like in case 1, this may lead to congestion on PE2 links upon LFA activation.

3.3. Case 3: suboptimal P router alternate choice

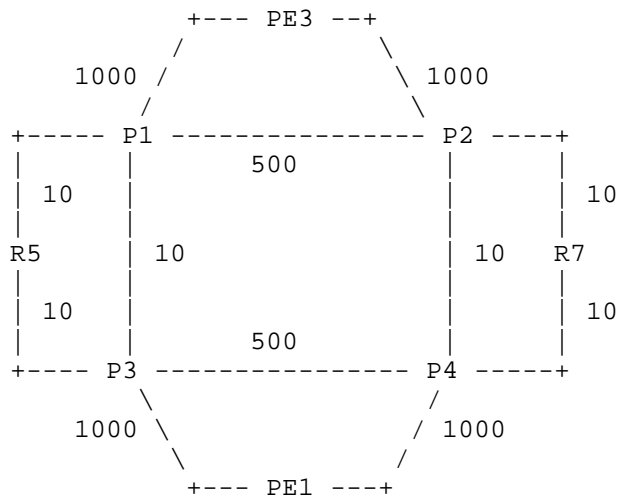


Figure 3

Px routers are P routers. P1-P2 and P3-P4 links are 1G links. All others inter Px links are 10G links.

In the figure above, let us consider the failure of link P1-P3. For destination PE3, P3 has two possible alternates:

- o P4, which is node-protecting
- o R5, which is link-protecting

P4 is chosen as best LFA due to its better protection type. However, it may not be desirable to use P4 for bandwidth capacity reason. A service provider may prefer to use high bandwidth links as preferred LFA. In this example, preferring shortest path over protection type may achieve the expected behavior, but in cases where metric are not reflecting bandwidth, it would not work and some other criteria would need to be involved when selecting the best LFA.

3.4. Case 4: No-transit LFA computing node

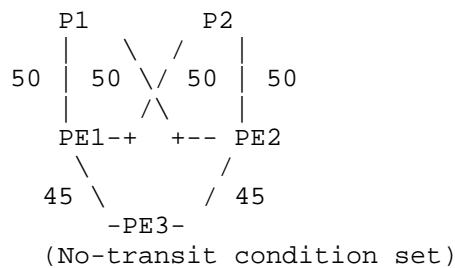


Figure 4

IS-IS and OSPF protocols define some way to prevent a router to be used as transit.

IS-IS overload bit is defined in [ISO10589] and OSPF R-bit is defined in [RFC5340]. OSPF Stub Router is also defined in [RFC6987] as a method to prevent transit on a node by advertising MaxLinkMetric on all non stub links.

In the figure above, PE3 has its no-transit condition set (permanently, for design reason) and wants to protect traffic using LFA for destination PE2.

On PE3, the loop-free condition is not satisfied : $100 \nless 45 + 45$. PE1 is thus not considered as an LFA. However thanks to the no-transit condition on PE3, we know that PE1 will not loop the traffic back to PE3. So PE1 is an LFA to reach PE2.

In case of no-transit condition set on a node, LFA behavior must be clarified.

4. Need for coverage monitoring

As per [RFC6571], LFA coverage highly depends on the used network topology. Even if remote LFA ([RFC7490]) extends significantly the coverage of the basic LFA specification, there is still some cases where protection would not be available. As network topologies are constantly evolving (network extension, capacity addings, latency optimization etc.), the protection coverage may change. Fast reroute functionality may be critical for some services supported by the network, a service provider must constantly know what protection coverage is currently available on the network. Moreover, predicting the protection coverage in case of network topology change is mandatory.

Today network simulation tool associated with whatif scenarios functionality are often used by service providers for the overall

network design (capacity, path optimization etc.). Section 7.5, Section 7.4 and Section 7.3 of this document propose to add LFA informations into such tool and within routers, so a service provider may be able :

- o to evaluate protection coverage after a topology change.
- o to adjust the topology change to cover the primary need (e.g. latency optimization or bandwidth increase) as well as LFA protection.
- o to monitor constantly the LFA coverage in the live network and being alerted.

Documentation of LFA selection algorithms by implementers (default and tuning options) is important in order to leave possibility for 3rd party modules to model these policy-LFA expressions.

5. Need for LFA activation granularity

As in all FRR mechanism, LFA installs backup paths in Forwarding Information Base (FIB). Depending on the hardware used by a service provider, FIB resource may be critical. Activating LFA, by default, on all available components (IGP topologies, interface, address families etc.) may lead to waste of FIB resource as generally in a network only few destinations should be protected (e.g. loopback addresses supporting MPLS services) compared to the number of destinations in the RIB.

Moreover a service provider may implement multiple different FRR mechanism in its networks for different usages (MRT, TE FRR). In this scenario, an implementation MAY allow to compute alternates for a specific destination even if the destination is already protected by another mechanism. This will bring redundancy and let the ability for the operator to select the best option for FRR using a policy language.

Section 6 of this document propose some implementation guidelines.

6. Configuration requirements

Controlling best alternate and LFA activation granularity is a requirement for Service Providers. This section defines configuration requirements for LFA.

6.1. LFA enabling/disabling scope

The granularity of LFA activation SHOULD be controlled (as alternate next hop consume memory in forwarding plane).

An implementation of LFA SHOULD allow its activation with the following granularities:

- o Per routing context: VRF, virtual/logical router, global routing table, etc.
- o Per interface
- o Per protocol instance, topology, area
- o Per prefixes: prefix protection SHOULD have a higher priority compared to interface protection. This means that if a specific prefix must be protected due to a configuration request, LFA MUST be computed and installed for this prefix even if the primary outgoing interface is not configured for protection.

An implementation of LFA MAY allow its activation with the following criteria:

- o Per address-family: ipv4 unicast, ipv6 unicast
- o Per MPLS control plane: for MPLS control planes that inherit routing decision from the IGP routing protocol, MPLS dataplane may be protected by LFA. The implementation may allow operator to control this inheritance of protection from the IP prefix to the MPLS label bound to this prefix. The protection inheritance will concern : IP to MPLS, MPLS to MPLS, and MPLS to IP entries. As example, LDP and segment-routing extensions for ISIS and OSPF are control plane eligible to this inheritance of protection.

6.2. Policy based LFA selection

When multiple alternates exist, LFA selection algorithm is based on tie breakers. Current tie breakers do not provide sufficient control on how the best alternate is chosen. This document proposes an enhanced tie breaker allowing service providers to manage all specific cases:

1. An implementation of LFA SHOULD support policy-based decision for determining the best LFA.
2. Policy based decision SHOULD be based on multiple criterions, with each criteria having a level of preference.

3. If the defined policy does not allow the determination of a unique best LFA, an implementation SHOULD pick only one based on its own decision. An implementation SHOULD also support election of multiple LFAs, for loadbalancing purposes.
4. Policy SHOULD be applicable to a protected interface or to a specific set of destinations. In case of application on the protected interface, all destinations primarily routed on this interface SHOULD use the interface policy.
5. It is an implementation choice to reevaluate policy dynamically or not (in case of policy change). If a dynamic approach is chosen, the implementation SHOULD recompute the best LFAs and reinstall them in FIB, without service disruption. If a non-dynamic approach is chosen, the policy would be taken into account upon the next IGP event. In this case, the implementation SHOULD support a command to manually force the recomputation/reinstallation of LFAs.

6.2.1. Connected vs remote alternates

In addition to connected LFAs, tunnels (e.g. IP, LDP, RSVP-TE or Segment Routing) to distant routers may be used to complement LFA coverage (tunnel tail used as virtual neighbor). When a router has multiple alternate candidates for a specific destination, it may have connected alternates and remote alternates (reachable via a tunnel). Connected alternates may not always provide an optimal routing path and it may be preferable to select a remote alternate over a connected alternate. Some usage of tunnels to extend LFA ([RFC5286]) coverage is described in either [RFC7490] or [I-D.francois-segment-routing-ti-lfa]. These documents present some use cases of LDP tunnels ([RFC7490]) or Segment Routing tunnels ([I-D.francois-segment-routing-ti-lfa]). This document considers any type of tunneling techniques to reach remote alternates (IP, GRE, LDP, RSVP-TE, L2TP, Segment Routing etc.) and does not restrict the remote alternates to the usage presented in the referenced document.

In figure 1, there is no P router alternate for P8 to reach PE4 or PE5, so P8 is using PE2 as alternate, which may generate congestion when FRR is activated. Instead, we could have a remote alternate for P8 to protect traffic to PE4 and PE5. For example, a tunnel from P8 to P3 (following shortest path) can be setup and P8 would be able to use P3 as remote alternate to protect traffic to PE4 and PE5. In this scenario, traffic will not use a PE link during FRR activation.

When selecting the best alternate, the selection algorithm MUST consider all available alternates (connected or tunnel). For example

with Remote LFA, computation of PQ set ([RFC7490]) SHOULD be performed before best alternate selection.

6.2.2. Mandatory criteria

An implementation of LFA MUST support the following criteria:

- o Non candidate link: A link marked as "non candidate" will never be used as LFA.
- o A primary next hop being protected by another primary next hop of the same prefix (ECMP case).
- o Type of protection provided by the alternate: link protection, node protection. In case of node protection preference, an implementation SHOULD support fall back to link protection if node protection is not available.
- o Shortest path: lowest IGP metric used to reach the destination.
- o SRLG (as defined in [RFC5286] Section 3, see also Section 6.2.4.1 for more details).

6.2.3. Additional criteria

An implementation of LFA SHOULD support the following criteria:

- o Downstreamness of an alternate : preference of a downstream path over a non downstream path SHOULD be configurable.
- o Link coloring with : include, exclude and preference based system (see Section 6.2.4.2).
- o Link Bandwidth (see Section 6.2.4.3).
- o Alternate preference/Node coloring (see Section 6.2.4.4).

6.2.4. Criteria evaluation

6.2.4.1. SRLG

[RFC5286] Section 3. proposes to reuse GMPLS IGP extensions to encode Shared Risk Link Groups ([RFC4205] and [RFC4203]). The section is also describing the algorithm to compute SRLG protection.

When SRLG protection is computed, an implementation SHOULD allow the following :

- o Exclusion alternates violating SRLG.
- o Maintenance of a preference system between alternates based on SRLG violations. How the preference system is implemented is out of scope of this document but here are few examples :
 - * Preference based on number of violations. In this case : the more violations = the less preferred.
 - * Preference based on violation cost. In this case, each SRLG violation has an associated cost. The lower violation cost sum is preferred.

When applying SRLG criteria, the SRLG violation check SHOULD be performed on source to alternate as well as alternate to destination paths based on the SRLG set of the primary path. In the case of remote LFA, PQ to destination path attributes would be retrieved from SPT rooted at PQ.

6.2.4.2. Link coloring

Link coloring is a powerful system to control the choice of alternates. Link colors are markers that will allow to encode properties of a particular link. Protecting interfaces are tagged with colors. Protected interfaces are configured to include some colors with a preference level, and exclude others.

Link color information SHOULD be signalled in the IGP and admin-groups IGP extensions ([RFC5305] and [RFC3630]) that are already standardized, implemented and widely-used, SHOULD be used for encoding and signalling link colors.

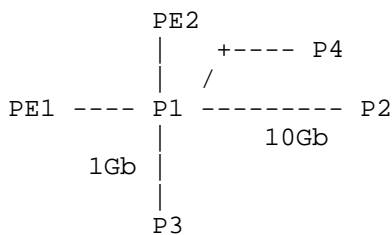


Figure 8

Example : P1 router is connected to three P routers and two PEs.

P1 is configured to protect the P1-P4 link. We assume that given the topology, all neighbors are candidate LFA. We would like to enforce a policy in the network where only a core router may protect against

the failure of a core link, and where high capacity links are preferred.

In this example, we can use the proposed link coloring by:

- o Marking PEs links with color RED
- o Marking 10Gb CORE link with color BLUE
- o Marking 1Gb CORE link with color YELLOW
- o Configured the protected interface P1->P4 with :
 - * Include BLUE, preference 200
 - * Include YELLOW, preference 100
 - * Exclude RED

Using this, PE links will never be used to protect against P1-P4 link failure and 10Gb link will be preferred.

The main advantage of this solution is that it can easily be duplicated on other interfaces and other nodes without change. A Service Provider has only to define the color system (associate color with a significance), as it is done already for TE affinities or BGP communities.

An implementation of link coloring:

- o SHOULD support multiple include and exclude colors on a single protected interface.
- o SHOULD provide a level of preference between included colors.
- o SHOULD support multiple colors configuration on a single protecting interface.

6.2.4.3. Bandwidth

As mentioned in previous sections, not taking into account bandwidth of an alternate could lead to congestion during FRR activation. We propose to base the bandwidth criteria on the link speed information for the following reason :

- o if a router S has a set of X destinations primarily forwarded to N, using per prefix LFA may lead to have a subset of X protected by a neighbor N1, another subset by N2, another subset by Nx etc.

- o S is not aware about traffic flows to each destination and is not able to evaluate how much traffic will be sent to N1,N2, etc. Nx in case of FRR activation.

Based on this, it is not useful to gather available bandwidth on alternate paths, as the router does not know how much bandwidth it requires for protection. The proposed link speed approach provides a good approximation with a small cost as information is easily available.

The bandwidth criteria of the policy framework SHOULD work in at least two ways :

- o PRUNE : exclude a LFA if link speed to reach it is lower than the link speed of the primary next hop interface.
- o PREFER : prefer a LFA based on its bandwidth to reach it compared to the link speed of the primary next hop interface.

6.2.4.4. Alternate preference/Node coloring

Rather than tagging interface on each node (using link color) to identify alternate node type (as example), it would be helpful if routers could be identified in the IGP. This would allow a grouped processing on multiple nodes. As an implementation need to exclude some specific alternates (see Section 6.2.3), an implementation :

- o SHOULD be able to give a preference to specific alternate.
- o SHOULD be able to give a preference to a group of alternate.
- o SHOULD be able to exclude a specific alternate.
- o SHOULD be able to exclude a group of alternate.

A specific alternate may be identified by its interface, IP address or router ID and group of alternates may be identified by a marker (tag) advertised in IGP. The IGP encoding and signalling for marking group of alternates SHOULD be done using [I-D.ietf-isis-node-admin-tag], [I-D.ietf-ospf-node-admin-tag]. Using a tag/marker is referred as Node coloring in comparison to link coloring option presented in Section 6.2.4.2.

Consider the following network:

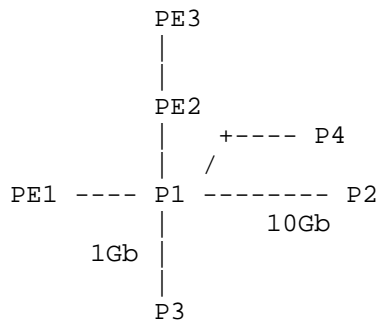


Figure 9

In the example above, each node is configured with a specific tag flooded through the IGP.

- o PE1,PE3: 200 (non candidate).
- o PE2: 100 (edge/core).
- o P1,P2,P3: 50 (core).

A simple policy could be configured on P1 to choose the best alternate for P1->P4 based on router function/role as follows :

- o criteria 1 -> alternate preference: exclude tag 100 and 200.
- o criteria 2 -> bandwidth.

6.2.5. Retrieving alternate path attributes

6.2.5.1. Alternate path

The alternate path is composed of two distinct parts : PLR to alternate and alternate to destination.

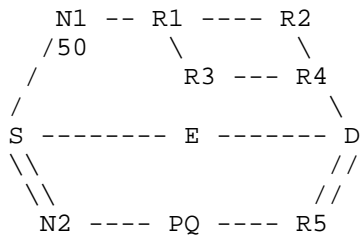


Figure 5

In the figure above, we consider a primary path from S to D, S using E as primary nexthop. All metrics are 1 except $\{S,N1\}=50$. Two alternate paths are available:

- o $\{S,N1,R1,R2|R3,R4,D\}$ where N1 is a connected alternate. This consists of two sub-paths:
 - * $\{S,N1\}$: path from PLR to the alternate.
 - * $\{N1,R1,R2|R3,R4,D\}$: path from alternate to destination.
- o $\{S,N2,PQ,R5,D\}$ where PQ is a remote alternate. Again the path consists of two sub-paths:
 - * $\{S,N2,PQ\}$: path from PLR to the alternate.
 - * $\{PQ,R5,D\}$: path from alternate to destination.

As displayed in the figure, some part of the alternate path may fanout in multipath due to ECMP.

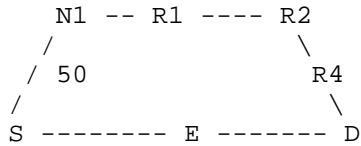
6.2.5.2. Alternate path attributes

Some criterions listed in the previous sections are requiring to retrieve some characteristic of the alternate path (SRLG, bandwidth, color, tag etc.). We call these characteristics "path attributes". A path attribute can record a list of node properties (e.g. node tag) or link properties (e.g. link color).

This document defines two types of path attributes:

- o Cumulative attribute: when a path attribute is cumulative, the implementation SHOULD record the value of the attribute on each element (link and node) along the alternate path. SRLG, link color, and node color are cumulative attributes.

- o Unitary attribute: when a path attribute is unitary, the implementation SHOULD record the value of the attribute only on the first element along the alternate path (first node, or first link). Bandwidth is a unitary attribute.



In the figure above, N1 is a connected alternate to each D from S. We consider that all links have a RED color except {R1,R2} which is BLUE. We consider all links to be 10Gbps, except {N1,R1} which is 2.5Gbps. The bandwidth attribute collected for the alternate path will be 10Gbps. As the attribute is unitary, only the link speed of the first link {S,N1} is recorded. The link color attribute collected for the alternate path will be {RED,RED,BLUE,RED,RED}. As the attribute is cumulative, the value of the attribute on each link along the path is recorded.

6.2.5.3. Connected alternate

For alternate path using a connected alternate:

- o attributes from PLR to alternate are retrieved from the interface connected to the alternate. In case the alternate is connected through multiple interfaces, the evaluation of attributes SHOULD be done once per interface (each interface is considered as a separate alternate) and once per ECMP group of interfaces (Layer 3 bundle).
- o path attributes from alternate to destination are retrieved from SPF rooted at the alternate. As the alternate is a connected alternate, the SPF has already been computed to find the alternate, so there is no need of additional computation.

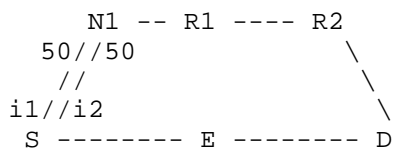


Figure 6

In the figure above, we consider a primary path from S to D, S using E as primary nexthop. All metrics are considered as 1 except {S,N1}

links which are using metric of 50. We consider the following SRLG groups on links:

- o {S,N1} using i1 : SRLG1,SRLG10
- o {S,N1} using i2 : SRLG2,SRLG20
- o {N1,R1} : SRLG3
- o {R1,R2} : SRLG4
- o {R2,D} : SRLG5
- o {S,E} : SRLG10
- o {E,D} : SRLG6

S is connected to the alternate using two interfaces i1 and i2.

If i1 and i2 are not part of an ECMP group, the evaluation of attributes is done once per interface, and each interface is considered as a separate alternate path. Two alternate paths will be available with the associated SRLG attributes :

- o Alternate path #1 : {S,N1 using if1,R1,R2,D}:
SRLG1,SRLG10,SRLG3,SRLG4,SRLG5.
- o Alternate path #2 : {S,N1 using if2,R1,R2,D}:
SRLG2,SRLG20,SRLG3,SRLG4,SRLG5.

Alternate path #1 is sharing risks with primary path and may be depreferred or pruned by user defined policy.

If i1 and i2 are part of an ECMP group, the evaluation of attributes is done once per ECMP group, and the implementation considers a single alternate path {S,N1 using if1|if2,R1,R2,D} with the following SRLG attributes: SRLG1,SRLG10,SRLG2,SRLG20,SRLG3,SRLG4,SRLG5. Alternate path is sharing risks with primary path and may be depreferred or pruned by user defined policy.

6.2.5.4. Remote alternate

For alternate path using a remote alternate (tunnel) :

- o Attributes on the path from the PLR to alternate are retrieved using the PLR's primary SPF (when using a PQ-node from P-Space) or the immediate neighbor's SPF (when using a PQ from extended P-Space). These are then combined with the attributes of the

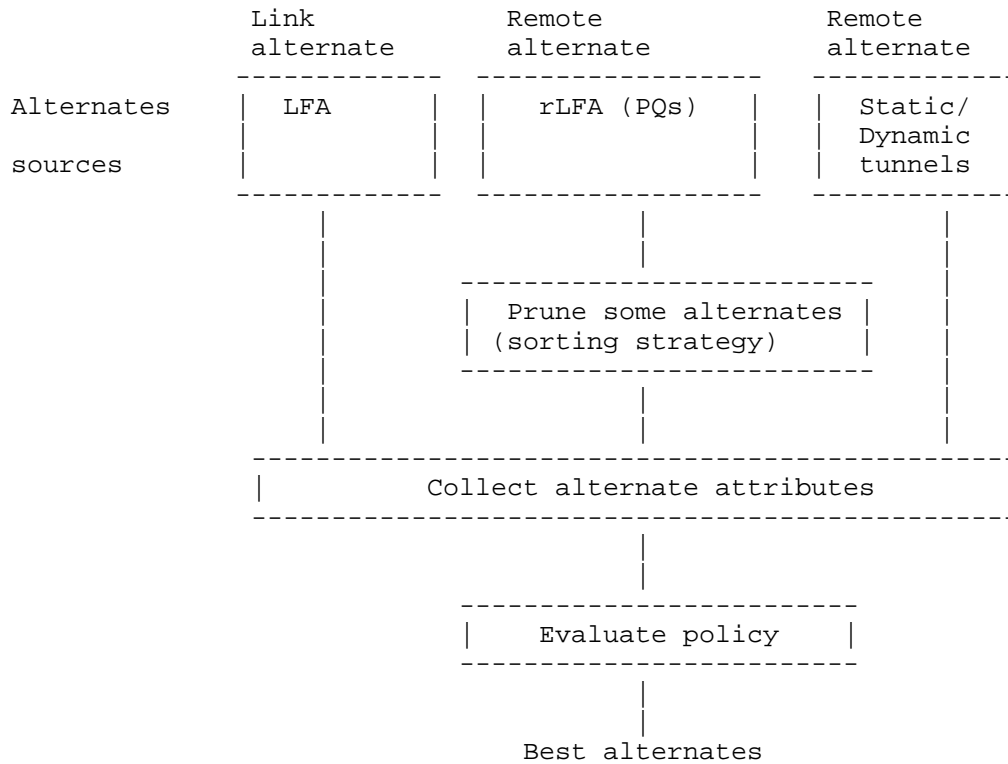
link(s) to reach the immediate neighbor. In both cases, no additional SPF is required.

- o Attributes from remote alternate to destination path may be retrieved from SPF rooted at the remote alternate. An additional forward SPF is required for each remote alternate (PQ-node) as indicated in [I-D.ietf-rtgwg-rlfa-node-protection] section 3.2 . In some remote alternate scenarios, like [I-D.francois-segment-routing-ti-lfa], alternate to destination path attributes may be obtained using a different technique.

The number of remote alternates may be very high. . In case of remote LFA, simulations of real-world network topologies have shown that order of hundreths of PQ may be possible. The computational overhead to collect all path attributes of all PQ to destination paths may grow beyond practical reason.

To handle this situation, implementations need to limit the number of remote alternates to be evaluated to a finite number before collecting alternate path attributes and running the policy evaluation. [I-D.ietf-rtgwg-rlfa-node-protection] Section 2.3.3 provides a way to reduce the number of PQ to be evaluated.

Some other remote alternate techniques using static or dynamic tunnels may not require this pruning.



6.2.5.5. Collecting attributes in case of multipath

As described in Section 6.2.5, there may be some situation where an alternate path or part of an alternate path fans out to multiple paths (e.g. ECMP). When collecting path attributes in such case, an implementation SHOULD consider the union of attributes of each sub-path.

In the figure 5 (in Section 6.2.5), S has two alternates paths to reach D. Each alternate path fans out into multipath due to ECMP. Considering the following link color attributes : all links are RED except {R1,R3} which is BLUE. The user wants to use an alternate path with only RED links. The first alternate path {S,N1,R1,R2|R3,R4,D} does not fit the constraint, as {R1,R3} is BLUE. The second alternate path {S,N2,PQ,R5,D} fits the constraint and will be preferred as it uses only RED links.

6.2.6. ECMP LFAs

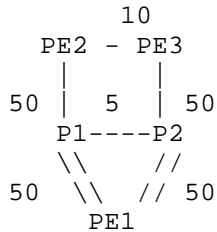


Figure 7

Links between P1 and PE1 are L1 and L2, links between P2 and PE1 are L3 and L4

In the figure above, primary path from PE1 to PE2 is through P1 using ECMP on two parallel links L1 and L2. In case of standard ECMP behavior, if L1 is failing, postconvergence next hop would become L2 and there would be no longer ECMP. If LFA is activated, as stated in [RFC5286] Section 3.4., "alternate next-hops may themselves also be primary next-hops, but need not be" and "alternate next-hops should maximize the coverage of the failure cases". In this scenario there is no alternate providing node protection, LFA will so prefer L2 as alternate to protect L1 which makes sense compared to postconvergence behavior.

Considering a different scenario using figure 7, where L1 and L2 are configured as a layer 3 bundle using a local feature, as well as L3/L4 being a second layer 3 bundle. Layer 3 bundles are configured as if a link in the bundle is failing, the traffic must be rerouted out of the bundle. Layer 3 bundles are generally introduced to increase bandwidth between nodes. In nominal situation, ECMP is still available from PE1 to PE2, but if L1 is failing, postconvergence next hop would become ECMP on L3 and L4. In this case, LFA behavior SHOULD be adapted in order to reflect the bandwidth requirement.

We would expect the following FIB entry on PE1 :

```

On PE1 : PE2 +--> ECMP -> L1
              |
              | +-----> L2
              |
              +--> LFA(ECMP) -> L3
                   |
                   +-----> L4

```

If L1 or L2 is failing, traffic must be switched on the LFA ECMP bundle rather than using the other primary next hop.

As mentioned in [RFC5286] Section 3.4., protecting a link within an ECMP by another primary next hop is not a MUST. Moreover, we already presented in this document, that maximizing the coverage of the failure case may not be the right approach and policy based choice of alternate may be preferred.

An implementation SHOULD allow to prefer to protect a primary next hop by another primary next hop. An implementation SHOULD allow to prefer to protect a primary next hop by a NON primary next hop. An implementation SHOULD allow to use an ECMP bundle as a LFA.

7. Operational aspects

7.1. No-transit condition on LFA computing node

In [RFC5286], Section 3.5, the setting of the no-transit condition (through IS-IS overload or OSPF R-bit) in LFA computation is only taken into account for the case where a neighbor has the no-transit condition set.

In addition to RFC 5286 inequality 1 Loop-Free Criterion ($\text{Distance_opt}(N, D) < \text{Distance_opt}(N, S) + \text{Distance_opt}(S, D)$), the IS-IS overload bit or OSPF R-bit of the LFA calculating neighbor (S) SHOULD be taken into account. Indeed, if it has the IS-IS overload bit set or OSPF R-bit clear, no neighbor will loop back to traffic to itself.

An OSPF router acting as a stub router [RFC 6987] SHOULD behave as if R-bit was clear regarding LFA computation.

7.2. Manual triggering of FRR

Service providers often perform manual link shutdown (using router CLI) to perform some network changes/tests. A manual link shutdown may be done at multiple level : physical interface, logical interface, IGP interface, BFD session etc. Especially testing or troubleshooting FRR requires to perform the manual shutdown on the remote end of the link as generally a local shutdown would not trigger FRR.

To enhance such situation, an implementation SHOULD support triggering/activating LFA Fast Reroute for a given link when a manual shutdown is done on a component that currently supports FRR activation.

An implementation MAY also support FRR activation for a specific interface or a specific prefix on a primary next-hop interface and revert without any action on any running component of the node (links or protocols). In this use case, the FRR activation time need to be controlled by a timer in case the operator forgot to revert traffic on primary path. When the timer expires, the traffic is automatically reverted to the primary path. This will make easier tests of fast-reroute path and then revert back to the primary path without causing a global network convergence.

For example :

- o if an implementation supports FRR activation upon BFD session down event, this implementation SHOULD support FRR activation when a manual shutdown is done on the BFD session. But if an implementation does not support FRR activation on BFD session down, there is no need for this implementation to support FRR activation on manual shutdown of BFD session.
- o if an implementation supports FRR activation on physical link down event (e.g. Rx laser Off detection, or error threshold raised etc.), this implementation SHOULD support FRR activation when a manual shutdown at physical interface is done. But if an implementation does not support FRR activation on physical link down event, there is no need for this implementation to support FRR activation on manual physical link shutdown.
- o A CLI command may allow to switch from primary path to FRR path for testing FRR path for a specific. There is no impact on controlplane, only dataplane of the local node could be changed. A similar command may allow to switch back traffic from FRR path to primary path.

7.3. Required local information

LFA introduction requires some enhancement in standard routing information provided by implementations. Moreover, due to the non 100% coverage, coverage informations is also required.

Hence an implementation :

- o MUST be able to display, for every prefix, the primary next hop as well as the alternate next hop information.
- o MUST provide coverage information per activation domain of LFA (area, level, topology, instance, virtual router, address family etc.).
- o MUST provide number of protected prefixes as well as non protected prefixes globally.
- o SHOULD provide number of protected prefixes as well as non protected prefixes per link.
- o MAY provide number of protected prefixes as well as non protected prefixes per priority if implementation supports prefix-priority insertion in RIB/FIB.
- o SHOULD provide a reason for choosing an alternate (policy and criteria) and for excluding an alternate.
- o SHOULD provide the list of non protected prefixes and the reason why they are not protected (no protection required or no alternate available).

7.4. Coverage monitoring

It is pretty easy to evaluate the coverage of a network in a nominal situation, but topology changes may change the coverage. In some situations, the network may no longer be able to provide the required level of protection. Hence, it becomes very important for service providers to get alerted about changes of coverage.

An implementation SHOULD :

- o provide an alert system if total coverage (for a node) is below a defined threshold or comes back to a normal situation.
- o provide an alert system if coverage of a specific link is below a defined threshold or comes back to a normal situation.

An implementation MAY :

- o trigger an alert if a specific destination is not protected anymore or when protection comes back up for this destination

Although the procedures for providing alerts are beyond the scope of this document, we recommend that implementations consider standard and well used mechanisms like syslog or SNMP traps.

7.5. LFA and network planning

The operator may choose to run simulations in order to ensure full coverage of a certain type for the whole network or a given subset of the network. This is particularly likely if he operates the network in the sense of the third backbone profiles described in [RFC6571], that is, he seeks to design and engineer the network topology in a way that a certain coverage is always achieved. Obviously a complete and exact simulation of the IP FRR coverage can only be achieved, if the behavior is deterministic and if the algorithm used is available to the simulation tool. Thus, an implementation SHOULD:

- o Behave deterministic in its selection LFA process. I.e. in the same topology and with the same policy configuration, the implementation MUST always choose the same alternate for a given prefix.
- o Document its behavior. The implementation SHOULD provide enough documentation of its behavior that allows an implementer of a simulation tool, to foresee the exact choice of the LFA implementation for every prefix in a given topology. This SHOULD take into account all possible policy configuration options. One possible way to document this behavior is to disclose the algorithm used to choose alternates.

8. Security Considerations

The policy mechanism introduced in this document allows to tune the selection of the alternate. This is not seen as a security threat as:

- o all candidates are already eligible as per [RFC5286] and considered useable.
- o the policy is based on information from the router's own configuration and from the IGP which are both considered trusted.

Hence this document does not introduce new security considerations compared to [RFC5286].

This document does not introduce any change in security consideration compared to [RFC5286]. The policy mechanism introduced in this document allow to tune the best alternate choice but does not change the list of alternates that are eligible. As defined in [RFC5286] Section 7., this best alternate "can be used anyway when a different topological change occurs, and hence this can't be viewed as a new security threat."

9. IANA Considerations

This document has no action for IANA.

10. Contributors

Significant contributions were made by Pierre Francois, Hannes Gredler, Chris Bowers, Jeff Tantsura, Uma Chunduri, Acee Lindem and Mustapha Aissaoui which the authors would like to acknowledge.

11. References

11.1. Normative References

[I-D.ietf-isis-node-admin-tag]

Sarkar, P., Gredler, H., Hegde, S., Litkowski, S., Decraene, B., Li, Z., Aries, E., Rodriguez, R., and H. Raghuv eer, "Advertising Per-node Admin Tags in IS-IS", draft-ietf-isis-node-admin-tag-02 (work in progress), June 2015.

[I-D.ietf-ospf-node-admin-tag]

Hegde, S., Raghuv eer, H., Gredler, H., Shakir, R., Smirnov, A., Li, Z., and B. Decraene, "Advertising per-node administrative tags in OSPF", draft-ietf-ospf-node-admin-tag-02 (work in progress), June 2015.

[ISO10589]

"Intermediate system to Intermediate system intra-domain routing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473), ISO/IEC 10589:2002, Second Edition.", Nov 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3137] Retana, A., Nguyen, L., White, R., Zinin, A., and D. McPherson, "OSPF Stub Router Advertisement", RFC 3137, June 2001.

- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, September 2003.
- [RFC4203] Kompella, K. and Y. Rekhter, "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, October 2005.
- [RFC4205] Kompella, K. and Y. Rekhter, "Intermediate System to Intermediate System (IS-IS) Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4205, October 2005.
- [RFC5286] Atlas, A. and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, September 2008.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, October 2008.
- [RFC5307] Kompella, K. and Y. Rekhter, "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, October 2008.
- [RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, July 2008.
- [RFC6571] Filsfils, C., Francois, P., Shand, M., Decraene, B., Uttaro, J., Leymann, N., and M. Horneffer, "Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks", RFC 6571, June 2012.
- [RFC6987] Retana, A., Nguyen, L., Zinin, A., White, R., and D. McPherson, "OSPF Stub Router Advertisement", RFC 6987, September 2013.
- [RFC7490] Bryant, S., Filsfils, C., Previdi, S., Shand, M., and N. So, "Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)", RFC 7490, April 2015.

11.2. Informative References

- [I-D.francois-segment-routing-ti-lfa]
Francois, P., Filsfils, C., Bashandy, A., and B. Decraene, "Topology Independent Fast Reroute using Segment Routing", draft-francois-segment-routing-ti-lfa-00 (work in progress), November 2013.

[I-D.ietf-rtgwg-rlfa-node-protection]

Sarkar, P., Gredler, H., Hegde, S., Bowers, C., Litkowski,
S., and H. Raghuv eer, "Remote-LFA Node Protection and
Manageability", draft-ietf-rtgwg-rlfa-node-protection-02
(work in progress), June 2015.

Authors' Addresses

Stephane Litkowski (editor)
Orange

Email: stephane.litkowski@orange.com

Bruno Decraene
Orange

Email: bruno.decraene@orange.com

Clarence Filsfils
Cisco Systems

Email: cfilsfil@cisco.com

Kamran Raza
Cisco Systems

Email: skraza@cisco.com

Martin Horneffer
Deutsche Telekom

Email: Martin.Horneffer@telekom.de

Pushpasis Sarkar
Juniper Networks

Email: psarkar@juniper.net

Routing Area Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 16, 2016

S. Litkowski
B. Decraene
Orange
C. Filsfils
Cisco Systems
P. Francois
IMDEA Networks
October 14, 2015

Microloop prevention by introducing a local convergence delay
draft-litkowski-rtgwg-uloop-delay-04

Abstract

This document describes a mechanism for link-state routing protocols to prevent local transient forwarding loops in case of link failure. This mechanism Proposes a two-steps convergence by introducing a delay between the convergence of the node adjacent to the topology change and the network wide convergence.

As this mechanism delays the IGP convergence it may only be used for planned maintenance or when fast reroute protects the traffic between the link failure and the IGP convergence.

Simulations using real network topologies have been performed and show that local loops are a significant portion (>50%) of the total forwarding loops.

Requirements Language

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 [RFC2119].

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on April 16, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Transient forwarding loops side effects	3
2.1. Fast reroute unefficiency	3
2.2. Network congestion	5
3. Overview of the solution	6
4. Specification	6
4.1. Definitions	7
4.2. Current IGP reactions	7
4.3. Local events	7
4.4. Local delay	8
4.4.1. Link down event	8
4.4.2. Link up event	9
5. Applicability	9
5.1. Applicable case : local loops	9
5.2. Non applicable case : remote loops	10
6. Simulations	10
7. Deployment considerations	11
8. Comparison with other solutions	12
8.1. PLSN	12
8.2. OFIB	13
9. Security Considerations	13
10. Acknowledgements	13
11. IANA Considerations	13
12. References	14
12.1. Normative References	14
12.2. Informative References	14

Authors' Addresses 15

1. Introduction

Micro-forwarding loops and some potential solutions are well described in [RFC5715]. This document describes a simple targeted mechanism that solves micro-loops local to the failure; based on network analysis, these are a significant portion of the micro-forwarding loops. A simple and easily deployable solution to these local micro-loops is critical because these local loops cause traffic loss after an advanced fast-reroute alternate has been used (see Section 2.1).

Consider the case in Figure 1 where S does not have an LFA to protect its traffic to D. That means that all non-D neighbors of S on the topology will send to S any traffic destined to D if a neighbor did not, then that neighbor would be loop-free. Regardless of the advanced fast-reroute technique used, when S converges to the new topology, it will send its traffic to a neighbor that was not loop-free and thus cause a local micro-loop. The deployment of advanced fast-reroute techniques motivates this simple router-local mechanism to solve this targeted problem. This solution can be work with the various techniques described in [RFC5715].

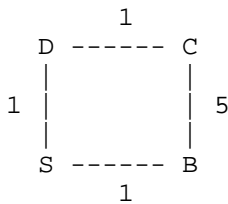


Figure 1

When S-D fails, a transient forwarding loop may appear between S and B if S updates its forwarding entry to D before B.

2. Transient forwarding loops side effects

Even if they are very limited in duration, transient forwarding loops may cause high damage for the network.

2.1. Fast reroute unefficiency

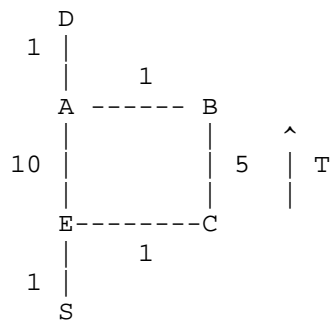
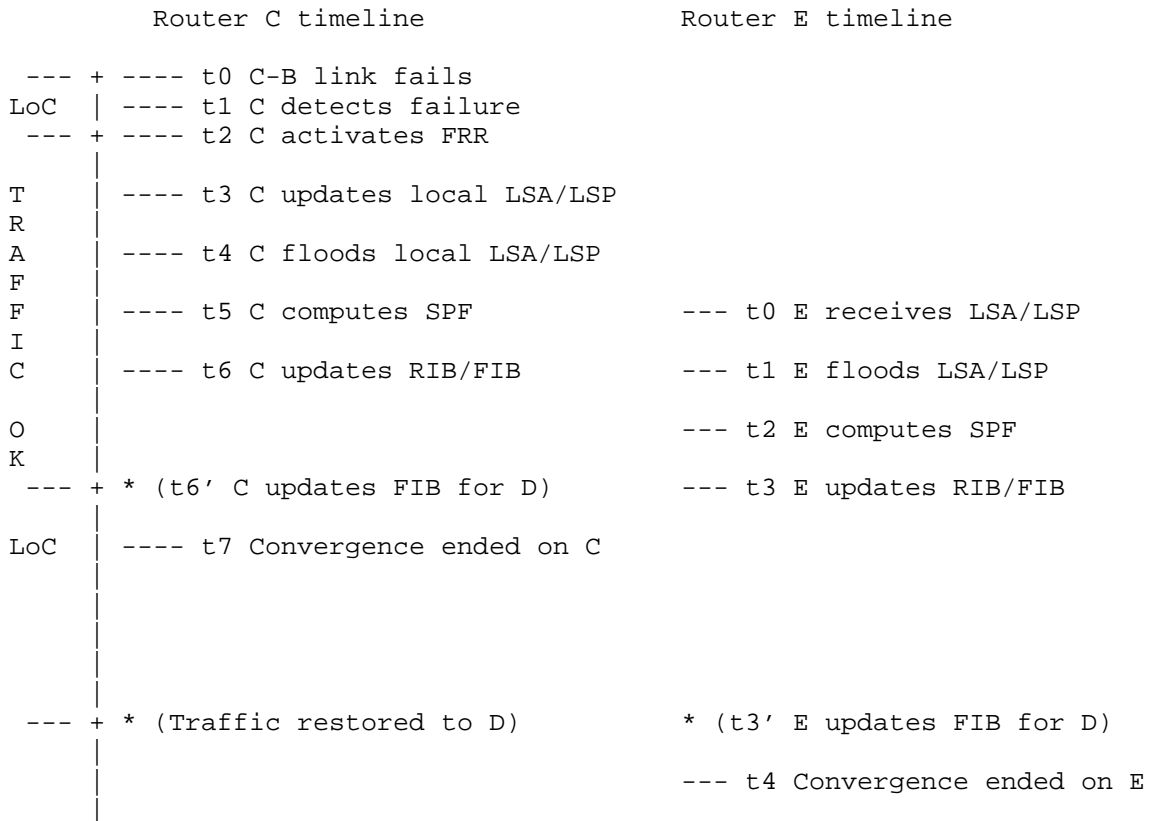


Figure 2 - RSVPTE FRR case

In figure 2, a RSVP-TE tunnel T, provisionned on C and terminating on B, is used to protect against C-B link failure (IGP shortcut activated on C). Primary path of T is C->B and FRR is activated on T providing a FRR bypass or detour using path C->E->A->B. On C, nexthop to D is tunnel T thanks to IGP shortcut. When C-B link fails :

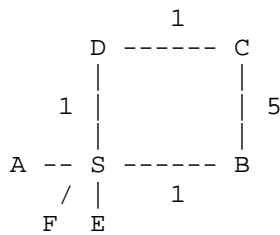
1. C detects the failure, and updates the tunnel path using preprogrammed FRR path, traffic path from S to D is :
S->E->C->E->A->B->A->D .
2. In parallel, on router C, both IGP convergence and TE tunnel convergence (tunnel path recomputation) are occurring :
 - * T path is recomputed : C->E->A->B
 - * IGP path to D is recomputed : C->E->A->D
3. On C, tail-end of the TE tunnel (router B) is no more on SPT to D, so C does not encapsulate anymore the traffic to D using the tunnel T and update forwarding entry to D using nexthop E.

If C updates its forwarding entry to D before router E, there would be a transient forwarding loop between C and E until E has converged.



The issue described here is completely independent of the fast-reroute mechanism involved (TE FRR, LFA/rLFA, MRT ...). Fast-reroute is working perfectly but ensures protection, by definition, only until the PLR has converged. When implementing FRR, a service provider wants to guarantee a very limited loss of connectivity time. The previous example shows that the benefit of FRR may be completely lost due to a transient forwarding loop appearing when PLR has converged. Delaying FIB updates after IGP convergence may permit to keep fast-reroute path until neighbor has converged and preserve customer traffic.

2.2. Network congestion



In the figure above, as presented in Section 1, when link S-D fails, a transient forwarding loop may appear between S and B for destination D. The traffic on S-B link will constantly increase due to the looping traffic to D. Depending on TTL of packets, traffic rate destined to D and bandwidth of link, the S-B link may be congested in few hundreds of milliseconds and will stay overloaded until the loop is solved.

Congestion introduced by transient forwarding loops are problematic as they are impacting traffic that is not directly concerned by the failing network component. In our example, the congestion of S-B link will impact customer traffic that is not directly concerned by the failure : e.g. A to B, F to B, E to B. Class of services may be implemented to mitigate the congestion but some traffic not directly concerned by the failure would still be dropped as a router is not able to identify looped traffic from normal traffic.

3. Overview of the solution

This document defines a two-step convergence initiated by the router detecting the failure and advertising the topological changes in the IGP. This introduces a delay between the convergence of the local router and the network wide convergence. This delay is positive in case of "down" events and negative in case of "up" events.

This ordered convergence, is similar to the ordered FIB proposed defined in [RFC6976], but limited to only one hop distance. As a consequence, it is simpler and becomes a local only feature not requiring interoperability; at the cost of only covering the transient forwarding loops involving this local router. The proposed mechanism also reuses some concept described in [I-D.ietf-rtgwg-microloop-analysis] with some limitation.

4. Specification

4.1. Definitions

This document will refer to the following existing IGP timers:

- o LSP_GEN_TIMER: to batch multiple local events in one single local LSP update. It is often associated with damping mechanism to slowdown reactions by incrementing the timer when multiple consecutive events are detected.
- o SPF_TIMER: to batch multiple events in one single computation. It is often associated with damping mechanism to slowdown reactions by incrementing the timer when the IGP is instable.
- o IGP_LDP_SYNC_TIMER: defined in [RFC5443] to give LDP some time to establish the session and learn the MPLS labels before the link is used.

This document introduces the following two new timers :

- o ULOOP_DELAY_DOWN_TIMER: slowdown the local node convergence in case of link down events.
- o ULOOP_DELAY_UP_TIMER: slowdown the network wide IGP convergence in case of link up events.

4.2. Current IGP reactions

Upon a change of status on an adjacency/link, the existing behavior of the router advertising the event is the following:

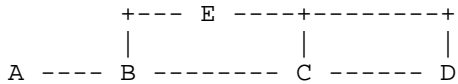
1. UP/Down event is notified to IGP.
2. IGP processes the notification and postpones the reaction in LSP_GEN_TIMER msec.
3. Upon LSP_GEN_TIMER expiration, IGP updates its LSP/LSA and floods it.
4. SPF is scheduled in SPF_TIMER msec.
5. Upon SPF_TIMER expiration, SPF is computed and RIB/FIB are updated.

4.3. Local events

The mechanisms described in this document assume that there has been a single failure as seen by the IGP area/level. If this assumption is violated (e.g. multiple links or nodes failed), then standard IP

convergence MUST be applied. There are three types of single failures: local link, local node, and remote failure.

Example :



Let B be the computing router when the link B-C fails. B updates its local LSP/LSA describing the link B->C as down, C does the same, and both start flooding their updated LSP/LSAs. During the SPF_TIMER period, B and C learn all the LSPs/LSAs to consider. B sees that C is flooding as down a link where B is the other end and that B and C are describing the same single event. Since B receives no other changes, B can determine that this is a local link failure.

[Editor s Note: Detection of a failed broadcast link involves additional complexity and will be described in a future version.]

If a router determines that the event is local link failure, then the router may use the mechanism described in this document.

Distinguishing local node failure from remote or multiple link failure requires additional logic which is future work to fully describe. To give a sense of the work necessary, if node C is failing, routers B,E and D are updating and flooding updated LSPs/LSAs. B would need to determine the changes in the LSPs/LSAs from E and D and see that they all relate to node C which is also the far-end of the locally failed link. Once this detection is accurately done, the same mechanism of delaying local convergence can be applied.

4.4. Local delay

4.4.1. Link down event

Upon an adjacency/link down event, this document introduces a change in step 5 in order to delay the local convergence compared to the network wide convergence: the node SHOULD delay the forwarding entry updates by ULOOP_DELAY_DOWN_TIMER. Such delay SHOULD only be introduced if all the LSDB modifications processed are only reporting down local events . Note that determining that all topological change are only local down events requires analyzing all modified LSP/LSA as a local link or node failure will typically be notified by multiple nodes. If a subsequent LSP/LSA is received/updated and a new SPF computation is triggered before the expiration of ULOOP_DELAY_DOWN_TIMER, then the same evaluation SHOULD be performed.

As a result of this addition, routers local to the failure will converge slower than remote routers. Hence it SHOULD only be done for non urgent convergence, such as for administrative de-activation (maintenance) or when the traffic is Fast ReRouted.

4.4.2. Link up event

Upon an adjacency/link up event, this document introduces the following change in step 3 where the node SHOULD:

- o Firstly build a LSP/LSA with the new adjacency but setting the metric to MAX_METRIC . It SHOULD flood it but not compute the SPF at this time. This step is required to ensure the two way connectivity check on all nodes when computing SPF.
- o Then build the LSP/LSA with the target metric but SHOULD delay the flooding of this LSP/LSA by SPF_TIMER + ULOOP_DELAY_UP_TIMER. MAX_METRIC is equal to MaxLinkMetric (0xFFFF) for OSPF and 2^24-2 (0xFFFFFE) for IS-IS.
- o Then continue with next steps (SPF computation) without waiting for the expiration of the above timer. In other word, only the flooding of the LSA/LSP is delayed, not the local SPF computation.

As as result of this addition, routers local to the failure will converge faster than remote routers.

If this mechanism is used in cooperation with "LDP IGP Synchronization" as defined in [RFC5443] then the mechanism defined in RFC 5443 is applied first, followed by the mechanism defined in this document. More precisely, the procedure defined in this document is applied once the LDP session is considered "fully operational" as per [RFC5443].

5. Applicability

As previously stated, the mechanism only avoids the forwarding loops on the links between the node local to the failure and its neighbor. Forwarding loops may still occur on other links.

5.1. Applicable case : local loops

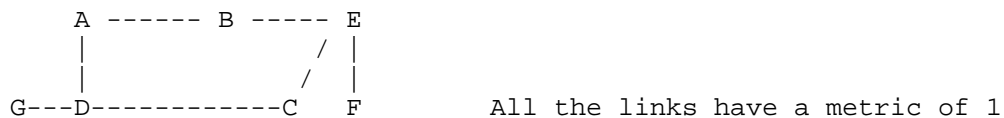
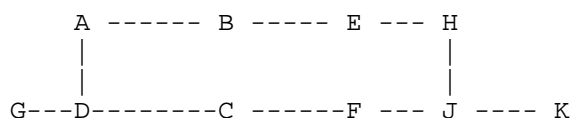


Figure 2

Let us consider the traffic from G to F. The primary path is G->D->C->E->F. When link CE fails, if C updates its forwarding entry for F before D, a transient loop occurs. This is sub-optimal as C has FRR enabled and it breaks the FRR forwarding while all upstream routers are still forwarding the traffic to itself.

By implementing the mechanism defined in this document on C, when the CE link fails, C delays the update of his forwarding entry to F, in order to let some time for D to converge. FRR keeps protecting the traffic during this period. When the timer expires on C, forwarding entry to F is updated. There is no transient forwarding loop on the link CD.

5.2. Non applicable case : remote loops



All the links have a metric of 1 except BE=15

Figure 3

Let us consider the traffic from G to K. The primary path is G->D->C->F->J->K. When the CF link fails, if C updates its forwarding entry to K before D, a transient loop occurs between C and D.

By implementing the mechanism defined in this document on C, when the link CF fails, C delays the update of his forwarding entry to K, letting time for D to converge. When the timer expires on C, forwarding entry to F is updated. There is no transient forwarding loop between C and D. However, a transient forwarding loop may still occur between D and A. In this scenario, this mechanism is not enough to address all the possible forwarding loops. However, it does not create additional traffic loss. Besides, in some cases -such as when the nodes update their FIB in the following order C, A, D, for example because the router A is quicker than D to converge- the mechanism may still avoid the forwarding loop that was occurring.

6. Simulations

Simulations have been run on multiple service provider topologies. So far, only link down event have been tested.

Topology	Gain
T1	71%
T2	81%
T3	62%
T4	50%
T5	70%
T6	70%
T7	59%
T8	77%

Table 1: Number of Repair/Dst that may loop

We evaluated the efficiency of the mechanism on eight different service provider topologies (different network size, design). The benefit is displayed in the table above. The benefit is evaluated as follows:

- o We consider a tuple (link A-B, destination D, PLR S, backup nexthop N) as a loop if upon link A-B failure, the flow from a router S upstream from A (A could be considered as PLR also) to D may loop due to convergence time difference between S and one of his neighbor N.
- o We evaluate the number of potential loop tuples in normal conditions.
- o We evaluate the number of potential loop tuples using the same topological input but taking into account that S converges after N.
- o Gain is how much loops (remote and local) we succeed to suppress.

On topology 1, 71% of the transient forwarding loops created by the failure of any link are prevented by implementing the local delay. The analysis shows that all local loops are obviously solved and only remote loops are remaining.

7. Deployment considerations

Transient forwarding loops have the following drawbacks :

- o Limit FRR efficiency : even if FRR is activated in 50msec, as soon as PLR has converged, traffic may be affected by a transient loop.

- o It may impact traffic not directly concerned by the failure (due to link congestion).

This local delay proposal is a transient forwarding loop avoidance mechanism (like OFIB). Even if it only address local transient loops, , the efficiency versus complexity comparison of the mechanism makes it a good solution. It is also incrementally deployable with incremental benefits, which makes it an attractive option for both vendors to implement and Service Providers to deploy. Delaying convergence time is not an issue if we consider that the traffic is protected during the convergence.

8. Comparison with other solutions

As stated in Section 3, our solution reuses some concepts already introduced by other IETF proposals but tries to find a tradeoff between efficiency and simplicity. This section tries to compare behaviors of the solutions.

8.1. PLSN

PLSN ([I-D.ietf-rtgwg-microloop-analysis]) describes a mechanism where each node in the network tries a avoid transient forwarding loops upon a topology change by always keeping traffic on a loop-free path for a defined duration (locked path to a safe neighbor). The locked path may be the new primary nexthop, another neighbor, or the old primary nexthop depending how the safety condition is satisfied.

PLSN does not solve all transient forwarding loops (see [I-D.ietf-rtgwg-microloop-analysis] Section 4 for more details).

Our solution reuse some concept of PLSN but in a more simple fashion :

- o PLSN has 3 different behavior : keep using old nexthop, use new primary nexthop if safe, or use another safe nexthop, while our solution only have one : keep using the current nexthop (old primary, or already activated FRR path).
- o PLSN may cause some damage while using a safe nexthop which is not the new primary nexthop in case the new safe nexthop does not enough provide enough bandwidth (see [I-D.ietf-rtgwg-lfa-manageability]). Our solution may not experience this issue as the service provider may have control on the FRR path being used preventing network congestion.

- o PLSN applies to all nodes in a network (remote or local changes), while our mechanism applies only on the nodes connected to the topology change.

8.2. OFIB

OFIB ([RFC6976]) describes a mechanism where convergence of the network upon a topology change is made ordered to prevent transient forwarding loops. Each router in the network must deduce the failure type from the LSA/LSP received and compute/apply a specific FIB update timer based on the failure type and its rank in the network considering the failure point as root.

This mechanism permit to solve all the transient forwarding loop in a network at the price of introducing complexity in the convergence process that may require strong monitoring by the service provider.

Our solution reuses the OFIB concept but limits it to the first hop that experience the topology change. As demonstrated, our proposal permits to solve all the local transient forwarding loops that represents a high percentage of all the loops. Moreover limiting the mechanism to one hop permit to keep the network-wide convergence behavior.

9. Security Considerations

This document does not introduce change in term of IGP security. The operation is internal to the router. The local delay does not increase the attack vector as an attacker could only trigger this mechanism if he already has be ability to disable or enable an IGP link. The local delay does not increase the negative consequences as if an attacker has the ability to disable or enable an IGP link, it can already harm the network by creating instability and harm the traffic by creating forwarding packet loss and forwarding loss for the traffic crossing that link.

10. Acknowledgements

We wish to thanks the authors of [RFC6976] for introducing the concept of ordered convergence: Mike Shand, Stewart Bryant, Stefano Previdi, and Olivier Bonaventure.

11. IANA Considerations

This document has no actions for IANA.

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC5443] Jork, M., Atlas, A., and L. Fang, "LDP IGP Synchronization", RFC 5443, DOI 10.17487/RFC5443, March 2009, <<http://www.rfc-editor.org/info/rfc5443>>.
- [RFC5715] Shand, M. and S. Bryant, "A Framework for Loop-Free Convergence", RFC 5715, DOI 10.17487/RFC5715, January 2010, <<http://www.rfc-editor.org/info/rfc5715>>.

12.2. Informative References

- [I-D.ietf-rtgwg-lfa-manageability] Litkowski, S., Decraene, B., Filsfils, C., Raza, K., Horneffer, M., and P. Sarkar, "Operational management of Loop Free Alternates", draft-ietf-rtgwg-lfa-manageability-11 (work in progress), June 2015.
- [I-D.ietf-rtgwg-microloop-analysis] Zinin, A., "Analysis and Minimization of Microloops in Link-state Routing Protocols", draft-ietf-rtgwg-microloop-analysis-01 (work in progress), October 2005.
- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <<http://www.rfc-editor.org/info/rfc3630>>.
- [RFC6571] Filsfils, C., Ed., Francois, P., Ed., Shand, M., Decraene, B., Uttaro, J., Leymann, N., and M. Horneffer, "Loop-Free Alternate (LFA) Applicability in Service Provider (SP) Networks", RFC 6571, DOI 10.17487/RFC6571, June 2012, <<http://www.rfc-editor.org/info/rfc6571>>.
- [RFC6976] Shand, M., Bryant, S., Previdi, S., Filsfils, C., Francois, P., and O. Bonaventure, "Framework for Loop-Free Convergence Using the Ordered Forwarding Information Base (oFIB) Approach", RFC 6976, DOI 10.17487/RFC6976, July 2013, <<http://www.rfc-editor.org/info/rfc6976>>.

[RFC7490] Bryant, S., Filsfils, C., Previdi, S., Shand, M., and N. So, "Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)", RFC 7490, DOI 10.17487/RFC7490, April 2015, <<http://www.rfc-editor.org/info/rfc7490>>.

Authors' Addresses

Stephane Litkowski
Orange

Email: stephane.litkowski@orange.com

Bruno Decraene
Orange

Email: bruno.decraene@orange.com

Clarence Filsfils
Cisco Systems

Email: cfilsfil@cisco.com

Pierre Francois
IMDEA Networks

Email: pierre.francois@imdea.org

Routing Area Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 26, 2014

P. Sarkar, Ed.
H. Gredler
S. Hegde
C. Bowers
Juniper Networks, Inc.
S. Litkowski
Orange
H. Raghuv eer

June 24, 2014

Remote-LFA Node Protection and Manageability
draft-psarkar-rtgwg-rlfa-node-protection-05

Abstract

The loop-free alternates computed following the current Remote-LFA [I-D.ietf-rtgwg-remote-lfa] specification gaurantees only link-protection. The resulting Remote-LFA nexthops (also called PQ-nodes), may not gaurantee node-protection for all destinations being protected by it.

This document describes procedures for determining if a given PQ-node provides node-protection for a specific destination or not. The document also shows how the same procedure can be utilised for collection of complete characteristics for alternate paths. Knowledge about the characteristics of all alternate path is precursory to apply operator defined policy for eliminating paths not fitting constraints.

Requirements Language

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 RFC2119 [RFC2119].

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on December 26, 2014.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 3
2. Node Protection with Remote-LFA 3
2.1. The Problem 4
2.2. Few Additional Definitions 5
2.2.1. Link-Protecting Extended P-Space 6
2.2.2. Node-Protecting Extended P-Space 6
2.2.3. Q-Space 7
2.2.4. Link-Protecting PQ Space 8
2.2.5. Candidate Node-Protecting PQ Space 8
2.3. Computing Node-protecting R-LFA Path 8
2.3.1. Computing Candidate Node-protecting PQ-Nodes for Primary nexthops 8
2.3.2. Computing node-protecting paths from PQ-nodes to destinations 10
2.3.3. Limiting extra computational overhead 12
3. Manageability of Remote-LFA Alternate Paths 13
3.1. The Problem 13
3.2. The Solution 14
4. Acknowledgements 14
5. IANA Considerations 14
6. Security Considerations 14
7. References 14
7.1. Normative References 15
7.2. Informative References 15

Authors' Addresses 15

1. Introduction

The Remote-LFA [I-D.ietf-rtgwg-remote-lfa] specification provides loop-free alternates that gaurantees only link-protection. The resulting Remote-LFA alternate nexthops (also referred to as the PQ-nodes) may not provide node-protection for all destinations covered by the same, in case of failure of the primary nexthop node. Neither does the specification provide a means to determine the same.

Also, the LFA Manageability [I-D.ietf-rtgwg-lfa-manageability] document, requires a computing router to find all possible (including all possible Remote-LFA) alternate nexthops, collect the complete set of path characteristics for each alternate path, run a alternate-selection policy (configured by the operator), and find the best alternate path. This will require the Remote-LFA implementation to gather all the required path characteristics along each link on the entire Remote-LFA alternate path.

With current LFA [RFC5286] and Remote-LFA implementations, the forward SPF (and reverse SPF) is run on the computing router and its immediate 1-hop routers as the roots. While that enables computation of path attributes (e.g. SRLG, Admin-groups) for first alternate path segment from the computing router to the PQ-node, there is no means for the computing router to gather any path attributes for the path segment from the PQ-node to destination. Consecutively any policy-based selection of alternate paths will consider only the path attributes from the computing router up until the PQ-node.

This document describes a procedure for determining node-protection with Remote-LFA. The same procedure are also extended for collection of complete set of path attributes, enabling more accurate policy-based selection for alternate paths obtained with Remote-LFA.

2. Node Protection with Remote-LFA

Node-protection is required to provide protection of traffic on a given forwarding node, against the failure of the first-hop node on the primary forwarding path. Such protection becomes more critical in the absence of mechanisms like non-stop-routing in the network. Certain operators refrains from deploying non-stop-routing in their network, due to the significant additional performance complexities it comes along with. In such cases node-protection is a must to gaurantee un-interrupted flow of traffic, even in the case of an entire forwarding node going down.

The following sections discuss the node-protection problem in the context of Remote-LFA and proposes a solution for solving the same.

2.1. The Problem

To better illustrate the problem and the solution proposed in this document the following topology diagram from the Remote-LFA [I-D.ietf-rtgwg-remote-lfa] draft is being re-used with slight modification.

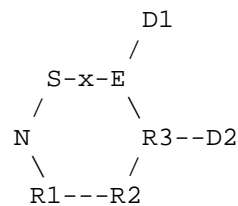


Figure 1: Topology 1

In the above topology, for all (non-ECMP) destinations reachable via the S-E link there is no standard LFA alternate. As per the Remote-LFA [I-D.ietf-rtgwg-remote-lfa] alternate specifications node R2 being the only PQ-node for the S-E link provides next-hop for all the above destinations. Table 1 below, shows all possible primary and Remote-LFA alternate paths for each destination.

Destination	Primary Path	PQ-node	Remote-LFA Backup Path
R3	S->E->R3	R2	S=>N=>R1=>R2->R3
E	S->E	R2	S=>N=>R1=>R2->R3->E
D1	S->E->D1	R2	S=>N=>R1=>R2->R3->E->D1
D2	S->E->R3->D2	R2	S=>N=>R1=>R2->R3->D2

Table 1: Remote-LFA backup paths via PQ-node R2

A closer look at Table 1 shows that, while the PQ-node R2 provides link-protection for all the destinations, it does not provide node-protection for destinations E and D1. In the event of the node-failure on primary next-hop E, the alternate path from Remote-LFA next-hop R2 to E and D1 also becomes unavailable. So for a Remote-LFA next-hop to provide node-protection for a given destination, it is mandatory that, the shortest path from the given PQ-node to the given destination MUST not traverse the primary next-hop.

In another extension of the topology in Figure 1 let us consider an additional link between N and E.

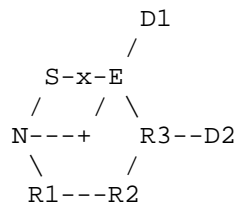


Figure 2: Topology 2

In the above topology, the S-E link is no more on any of the shortest paths from N to R3. Hence R3 is also included in both the Extended-P space and PQ space of E (w.r.t S-E link). Table 2 below, shows all possible primary and R-LFA alternate paths via PQ-node R3, for each destination reachable through the S-E link in the above topology. The R-LFA alternate paths via PQ-node R2 remains same as in Table 1.

Destination	Primary Path	PQ-node	Remote-LFA Backup Path
R3	S->E->R3	R3	S=>N=>E=>R3
E	S->E	R3	S=>N=>E=>R3->E
D1	S->E->D1	R3	S=>N=>E=>R3->E->D1
D2	S->E->R3->D2	R3	S=>N=>E=>R3->D2

Table 2: Remote-LFA backup paths via PQ-node R3

Again a closer look at Table 2 shows that, unlike Table 1, where the single PQ-node R2 provided node-protection, for destinations R3 and D1, if we choose R3 as the R-LFA nexthop, it does not provide node-protection for R3 and D1 anymore. If S chooses R3 as the R-LFA nexthop, in the event of the node-failure on primary nexthop E, the alternate path from S to R-LFA nexthop R3 also becomes unavailable. So for a Remote-LFA nexthop to provide node-protection for a given destination, it is also mandatory that, the shortest path from S to the chosen PQ-node MUST not traverse the primary nexthop node.

2.2. Few Additional Definitions

This document adds and enhances the following definitions extending the ones mentioned in Remote-LFA [I-D.ietf-rtgwg-remote-lfa] draft.

2.2.1. Link-Protecting Extended P-Space

The Remote-LFA [I-D.ietf-rtgwg-remote-lfa] draft already defines this. The link-protecting extended P-space for a link S-E being protected is the set of routers that are reachable from one or more direct neighbors of S, except primary node E, without traversing the S-E link on any of the shortest path from the direct neighbor to the router. This MUST exclude any direct neighbor for which there is at least one ECMP path from the direct neighbor traversing the link(S-E) being protected.

A node Y is in link-protecting extended P-space w.r.t to the link (S-E) being protected, if and only if, there exists atleast one direct neighbor of S, Ni, other than primary nexthop E, that satisfies the following condition.

$$D_{\text{opt}}(Ni, Y) < D_{\text{opt}}(Ni, S) + D_{\text{opt}}(S, Y)$$

Where,

- D_{opt}(A,B) : Distance on most optimum path from A to B.
- Ni : A direct neighbor of S other than primary nexthop E.
- Y : The node being evaluated for link-protecting extended P-Space.

Figure 3: Link-Protecting Ext-P-Space Condition

2.2.2. Node-Protecting Extended P-Space

The node-protecting extended P-space for a primary nexthop node E being protected, is the set of routers that are reachable from one or more direct neighbors of S, except primary node E, without traversing the node E. This MUST exclude any direct neighbors for which there is atleast one ECMP path from the direct neighbor traversing the node E being protected.

A node Y is in node-protecting extended P-space w.r.t to the node E being protected, if and only if, there exists atleast one direct neighbor of S, Ni, other than primary nexthop E, that satisfies the following condition.

$$D_{\text{opt}}(N_i, Y) < D_{\text{opt}}(N_i, E) + D_{\text{opt}}(E, Y)$$

Where,

- $D_{\text{opt}}(A, B)$: Distance on most optimum path from A to B.
- E : The primary nexthop on shortest path from S to destination.
- N_i : A direct neighbor of S other than primary nexthop E.
- Y : The node being evaluated for node-protecting extended P-Space.

Figure 4: Node-Protecting Ext-P-Space Condition

It must be noted that a node Y satisfying the condition in Figure 4 above only guarantees that the R-LFA alternate path segment from S via direct neighbor N_i to the node Y is not affected in the event of a node failure of E. It does not yet guarantee that the path segment from node Y to the destination is also unaffected by the same failure event.

2.2.3. Q-Space

The Remote-LFA [I-D.ietf-rtgwg-remote-lfa] draft already defines this. The Q-space for a link S-E being protected is the set of routers that can reach primary node E, without traversing the S-E link on any of the shortest path from the node Y to primary nexthop E. This MUST exclude any destination for which there is atleast one ECMP path from the node Y to the primary nexthop E traversing the link(S-E) being protected.

A node Y is in Q-space w.r.t to the link (S-E) being protected, if and only if, the following condition is satisfied.

$$D_{\text{opt}}(Y, E) < D_{\text{opt}}(S, E) + D_{\text{opt}}(Y, S)$$

Where,

- $D_{\text{opt}}(A, B)$: Distance on most optimum path from A to B.
- E : The primary nexthop on shortest path from S to destination.
- Y : The node being evaluated for Q-Space.

Figure 5: Q-Space Condition

2.2.4. Link-Protecting PQ Space

A node Y is in link-protecting PQ space w.r.t to the link (S-E) being protected, if and only if, Y is present in both link-protecting extended P-space and the Q-space for the link being protected.

2.2.5. Candidate Node-Protecting PQ Space

A node Y is in candidate node-protecting PQ space w.r.t to the node (E) being protected, if and only if, Y is present in both node-protecting extended P-space and the Q-space for the link being protected.

Again it must be noted that a node Y being in candidate node-protecting PQ-space does not guarantee that the R-LFA alternate path via the same, in entirety, is unaffected in the event of a node failure of primary nexthop node E. It only guarantees that the path segment from S to PQ-node Y is unaffected by the same failure event. The PQ-nodes in the candidate node-protecting PQ space may provide node protection for only a subset of destinations that are reachable through the corresponding primary link.

2.3. Computing Node-protecting R-LFA Path

The R-LFA alternate path through a given PQ-node to a given destination comprises of two path segments as follows.

1. Path segment from the computing router to the PQ-node (Remote-LFA alternate nexthop), and
2. Path segment from the PQ-node to the destination being protected.

So to ensure a R-LFA alternate path for a given destination provides node-protection we need to ensure that none of the above path segments are unaffected in the event of failure of the primary nexthop node. Sections Section 2.3.1 and Section 2.3.2 shows how this can be ensured.

2.3.1. Computing Candidate Node-protecting PQ-Nodes for Primary nexthops

To choose a node-protecting R-LFA nexthop for a destination R3, router S needs to consider a PQ-node from the candidate node-protecting PQ-space for the primary nexthop E on shortest path from S to R3. As mentioned in Section 2.2.2, to consider a PQ-node as candidate node-protecting PQ-node, there must be atleast one direct neighbor Ni of S, such that all shortest paths from Ni to the PQ-node does not traverse primary nexthop node E.

Implementations should run the inequality in Section 2.2.2 Figure 4 for all direct neighbor, other than primary nexthop node E, to determine whether a node Y is a candidate node-protecting PQ-node. All of the metrics needed by this inequality would have been already collected from the forward SPF's rooted at each of direct neighbor S, computed as part of standard LFA [RFC5286] implementation. With reference to the topology in Figure 2, Table 3 below shows how the above condition can be used to determine the candidate node-protecting PQ-space for S-E link (primary nexthop E)

Candidate PQ-node (Y)	Direct Nbr (Ni)	D_opt (Ni,Y)	D_opt (Ni,E)	D_opt (E,Y)	Condition Met
R2	N	2 (N,R2)	1 (N,E)	2 (E,R2)	Yes
R3	N	2 (N,R3)	1 (N,E)	1 (E,R3)	No

Table 3: Node-protection evaluation for R-LFA repair tunnel to PQ-node

As seen in the above Table 3 , R3 does not meet the node-protecting extended-p-space inequality And so, while R2 is in candidate node-protecting PQ space, R3 is not.

Some SPF implementations may also produce a list of links and nodes traversed on the shortest path(s) from a given root to others. In such implementations, router S may have executed a forward SPF with each of it's direct neighbors as the SPF root, executed as part of the standard LFA [RFC5286] computations. So S may re-use the list of links and nodes collected from the same SPF computations, to decide whether a node Y is a candidate node-protecting PQ-node or not. A node Y shall be considered as a node-protecting PQ-node, if and only if, there is atleast one direct neighbor of S, other than the primary nexthop E, for which, the primary nexthop node E does not exist on the list of nodes traversed on any of the shortest path(s) from the direct neighbor to the PQ-node. Table 4 below is an illustration of the mechanism with the topology in Figure 2.

Candidate PQ-node	Repair Tunnel Path (Repairing router to PQ-node)	Link-Protection	Node-Protection
R2	S->N->R1->R2	Yes	Yes
R2	S->E->R3->R2	No	No
R3	S->N->E->R3	Yes	No

Table 4: Protection of Remote-LFA tunnel to the PQ-node

As seen in the above Table 4 while R2 is candidate node-protecting Remote-LFA nexthop for R3 and D2, it is not so for E and D1, since the primary nexthop E is in the shortest path from R2 to E and F.

2.3.2. Computing node-protecting paths from PQ-nodes to destinations

Once a computing router finds all the candidate node-protecting PQ-nodes for a given directly attached primary link, it shall follow the procedure in proposed in this section, to choose one or more node-protecting R-LFA paths, for destinations reachable through the same primary link in the primary SPF graph.

To find a node-protecting R-LFA path for a given destination, the computing router needs to pick a subset of PQ-nodes from the candidate node-protecting PQ-space for the corresponding primary nexthop, such that all the path(s) from the PQ-node(s) to the given destination remain unaffected in the event of a node failure of primary nexthop node. To ensure this, the computing router will need to ensure that, the primary nexthop node should not be on any of the shortest paths from the PQ-node to the given destination.

This document proposes an additional forward SPF computation for each of the PQ-nodes, to discover all shortest paths from the PQ-nodes to the destination. The additional forward SPF computation for each PQ-node, shall help determine, if a given primary nexthop node is on the shortest paths from the PQ-node to the given destination or not. To determine if a given candidate node-protecting PQ-node provides node-protecting alternate for a given destination, the primary nexthop node should not be on any of the shortest paths from the PQ-node to the given destination. On running the forward SPF on a candidate node-protecting PQ-node the computing router shall run the inequality in Figure 6 below. PQ-nodes that does not qualify the condition for a given destination, does not guarantee node-protection for the path segment from the PQ-node to the given destination.

$$D_{\text{opt}}(Y,D) < D_{\text{opt}}(Y,E) + \text{Distance}_{\text{opt}}(E,D)$$

Where,

- $D_{\text{opt}}(A,B)$: Distance on most optimum path from A to B.
- D : The destination node.
- E : The primary nexthop on shortest path from S to destination.
- Y : The node-protecting PQ-node being evaluated

Figure 6: Node-Protecting Condition for PQ-node to Destination

All of the above metric costs except $D_{\text{opt}}(Y, D)$, can be obtained with forward and reverse SPF with E(the primary nexthop) as the root, run as part of the regular LFA and Remote-LFA implementation. The $\text{Distance}_{\text{opt}}(Y, D)$ metric can only be determined by the additional forward SPF run with PQ-node Y as the root. With reference to the topology in Figure 2, Table 5 below shows how the above condition can be used to determine node-protection with node-protecting PQ-node R2.

Destination (D)	Primary-NH (E)	D_{opt} (Y, D)	D_{opt} (Y, E)	D_{opt} (E, D)	Condition Met
R3	E	1 (R2,R3)	2 (R2,E)	1 (E,R3)	Yes
E	E	2 (R2,E)	2 (R2,E)	0 (E,E)	No
D1	E	3 (R2,D1)	2 (R2,E)	1 (E,D1)	No
D2	E	2 (R2,D2)	2 (R2,E)	1 (E,D2)	Yes

Table 5: Node-protection evaluation for R-LFA path segment between PQ-node and destination

As seen in the above example above, R2 does not meet the node-protecting inequality for destination E, and F. And so, once again, while R2 is a node-protecting Remote-LFA nexthop for R3 and G, it is not so for E and F.

In SPF implementations that also produce a list of links and nodes traversed on the shortest path(s) from a given root to others, to determine whether a PQ-node provides node-protection for a given destination or not, the list of nodes computed from forward SPF run on the PQ-node, for the given destination, should be inspected. In case the list contains the primary nexthop node, the PQ-node does not

provide node-protection. Else, the PQ-node guarantees node-protecting alternate for the given destination. Below is an illustration of the mechanism with candidate node-protecting PQ-node R2 in the topology in Figure 2.

Destination	Shortest Path (Repairing router to PQ- node)	Link-Protection	Node-Protection
R3	R2->R3	Yes	Yes
E	R2->R3->E	Yes	No
D1	R2->R3->E->D1	Yes	No
D2	R2->R3->D2	Yes	Yes

Table 6: Protection of Remote-LFA path between PQ-node and destination

As seen in the above example while R2 is candidate node-protecting R-LFA nexthop for R3 and G, it is not so for E and F, since the primary nexthop E is in the shortest path from R2 to E and F.

The procedure described in this document helps no more than to determine whether a given Remote-LFA alternate provides node-protection for a given destination or not. It does not find out any new Remote-LFA alternate nexthops, outside the ones already computed by standard Remote-LFA procedure. However, in case of availability of more than one PQ-node (Remote-LFA alternates) for a destination, and node-protection is required for the given primary nexthop, this procedure will eliminate the PQ-nodes that do not provide node-protection and choose only the ones that does.

2.3.3. Limiting extra computational overhead

In addition to the extra reverse SPF computation, one per directly connected neighbor, suggested by the Remote-LFA [I-D.ietf-rtgwg-remote-lfa] draft, this document proposes a forward SPF per PQ-node discovered in the network. Since the average number of PQ-nodes found in any network is considerably more than the number of direct neighbors of the computing router, the proposal of running one forward SPF per PQ-node may add considerably to the overall SPF computation time.

To limit the computational overhead of the approach proposed, this document proposes that implementations MUST choose a subset from the entire set of PQ-nodes computed in the network, with a finite limit

on the number of PQ-nodes in the subset. Implementations MUST choose a default value for this limit and may provide user with a configuration knob to override the default limit. Implementations MUST also evaluate some default preference criteria while considering a PQ-node in this subset. Finally, implementations MAY also allow user to override the default preference criteria, by providing a policy configuration for the same.

This document proposes that implementations SHOULD use a default preference criteria for PQ-node selection which will put a score on each PQ-node, proportional to the number of primary interfaces for which it provides coverage, its distance from the computing router, and its router-id (or system-id in case of IS-IS). PQ-nodes that cover more primary interfaces SHOULD be preferred over PQ-nodes that cover fewer primary interfaces. When two or more PQ-nodes cover the same number of primary interfaces, PQ-nodes which are closer (based on metric) to the computing router SHOULD be preferred over PQ-nodes farther away from it. For PQ-nodes that cover the same number of primary interfaces and are the same distance from the the computing router, the PQ-node with smaller router-id (or system-id in case of IS-IS) SHOULD be preferred.

Once a subset of PQ-nodes is found, computing router shall run a forward SPF on each of the PQ-nodes in the subset to continue with procedures proposed in section Section 2.3.2.

3. Manageability of Remote-LFA Alternate Paths

3.1. The Problem

With the regular Remote-LFA [I-D.ietf-rtgwg-remote-lfa] functionality the computing router may compute more than one PQ-node as usable Remote-LFA alternate nexthops. Additionally an alternate selection policy may be configured to enable the network operator to choose one of them as the most appropriate Remote-LFA alternate. For such policy-based alternate selection to run, all the relevant path characteristics for each the alternate paths (one through each of the PQ-nodes), needs to be collected. As mentioned before in section Section 2.3 the R-LFA alternate path through a given PQ-node to a given destination comprises of two path segments.

The first path segment (i.e. from the computing router to the PQ-node) can be calculated from the regular forward SPF done as part of standard and remote LFA computations. However without the mechanism proposed in section Section 2.3.2 of this document, there is no way to determine the path characteristics for the second path segment (i.e from the PQ-node to the destination). In the absence of the path characteristics for the second path segment, two Remote-LFA

alternate path may be equally preferred based on the first path segments characteristics only, although the second path segment attributes may be different.

3.2. The Solution

The additional forward SPF computation proposed in section Section 2.3.2 document shall also collect links, nodes and path characteristics along the second path segment. This shall enable collection of complete path characteristics for a given Remote-LFA alternate path to a given destination. The complete alternate path characteristics shall then facilitate more accurate alternate path selection while running the alternate selection policy.

Like specified in Section 2.3.3 to limit the computational overhead of the approach proposed, forward SPF computations MUST be run on a selected subset from the entire set of PQ-nodes computed in the network, with a finite limit on the number of PQ-nodes in the subset. The detailed suggestion on how to select this subset is specified in the same section. While this limits the number of possible alternate paths provided to the alternate-selection policy, this is needed keep the computational complexity within affordable limits. However if the alternate-selection policy is very restrictive this may leave few destinations in the entire topology without protection. Yet this limitation provides a necessary tradeoff between extensive coverage and immense computational overhead.

4. Acknowledgements

Many thanks to Bruno Decraene for providing his useful comments. We would also like to thank Uma Chunduri for reviewing this document and providing valuable feedback.

5. IANA Considerations

N/A. - No protocol changes are proposed in this document.

6. Security Considerations

This document does not introduce any change in any of the protocol specifications. It simply proposes to run an extra SPF rooted on each PQ-node discovered in the whole network.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

7.2. Informative References

- [I-D.ietf-rtgwg-lfa-manageability]
Litkowski, S., Decraene, B., Filsfils, C., and K. Raza,
"Operational management of Loop Free Alternates", draft-
ietf-rtgwg-lfa-manageability-00 (work in progress), May
2013.
- [I-D.ietf-rtgwg-remote-lfa]
Bryant, S., Filsfils, C., Previdi, S., Shand, M., and S.
Ning, "Remote LFA FRR", draft-ietf-rtgwg-remote-lfa-02
(work in progress), May 2013.
- [RFC5286] Atlas, A. and A. Zinin, "Basic Specification for IP Fast
Reroute: Loop-Free Alternates", RFC 5286, September 2008.

Authors' Addresses

Pushpasis Sarkar (editor)
Juniper Networks, Inc.
Electra, Exora Business Park
Bangalore, KA 560103
India

Email: psarkar@juniper.net

Hannes Gredler
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: hannes@juniper.net

Shraddha Hegde
Juniper Networks, Inc.
Electra, Exora Business Park
Bangalore, KA 560103
India

Email: shraddha@juniper.net

Chris Bowers
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: cbowers@juniper.net

Stephane Litkowski
Orange

Email: stephane.litkowski@orange.com

Harish Raghuv eer

Email: harish.r.prabhu@gmail.com

Routing Area Working Group
Internet-Draft
Intended status: Informational
Expires: May 23, 2019

Bin Liu
ZTE Inc., ZTE Plaza
Yantao Sun
Jing Cheng
Yichen Zhang
Beijing Jiaotong University
Bhumip Khasnabish
ZTE TX Inc.
Nov 19, 2018

Generic Fault-avoidance Routing Protocol for Data Center Networks
draft-sl-rtgwg-far-dcn-11

Abstract

This draft proposes a generic routing method and protocol for a regular data center network, named as fault-avoidance routing (FAR) protocol. FAR protocol provides a generic routing method for all types of network architectures that are proposed for large-scale cloud-based data centers over the past few years. FAR protocol is well designed to fully leverage the regularity in the topology and compute its routing table in a simplistic manner. Fat-tree is taken as an example architecture to illustrate how FAR protocol can be applied in real operational scenarios.

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). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on May 23, 2019.

Copyright Notice

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

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
1.1.	Acronyms & Definitions	4
2.	Conventions used in this document	5
3.	Problem Statement	5
3.1.	The Impact of Large-scale Networks on Route Calculation .	5
3.2.	Dilemma of conventional routing methods in a large-scale network with giant number nodes of routers	6
3.3.	Network Addressing Issues	8
3.4.	Big Routing Table Issues	8
3.5.	Adaptivity Issues for Routing Algorithms	8
3.6.	Virtual Machine Migration Issues	9
4.	The FAR Framework	9
5.	Data Format	10
5.1.	Data Tables	10
5.2.	Messages	13
6.	FAR Modules	17
6.1.	Neighbor and Link Detection Module (M1)	17
6.2.	Device Learning Module (M2)	17
6.3.	Invisible Neighbor and Link Failure Inferring Module (M3) .	18
6.4.	Link Failure Learning Module (M4)	18
6.5.	BRT Building Module (M5)	18
6.6.	NRT Building Module (M6)	19
6.7.	Routing Table Lookup (M7)	19
7.	How a FAR Router Works	19
8.	Compatible Architecture	22
9.	Application Example	22
9.1.	BRT Building Procedure	24
9.2.	NRT Building Procedure	25
9.2.1.	Single Link Failure	25
9.2.2.	A Group of Link Failures	26
9.2.3.	Node Failures	27
9.3.	Routing Procedure	27
9.4.	FAR's Performance in Large-scale Networks	29
9.4.1.	The number of control messages required by FAR	29
9.4.2.	The Calculating Time of Routing Tables	29
9.4.3.	The Size of Routing Tables	30

10. Security Considerations	30
11. Conclusions	30
12. Acknowledgments	31
13. References	31
Authors' Addresses	31

1. Introduction

In recent years, with the rapid development of cloud computing technologies, the widely deployed cloud services, such as Amazon EC2 and Google search, bring about huge challenges to data center networking (DCN). Today's cloud-based data centers (DCs) require large-scale networks with larger internal bandwidth and smaller transfer delay. However, conventional networks cannot meet such requirements due to limitations in their network architecture. In order to satisfy the requirements of cloud computing services, many new network architectures have been proposed for data centers, such as Fat-tree, MatrixDCN, and BCube. These new architectures can support non-blocking large-scale datacenter networks with more than tens of thousands of physical servers.

All of these architectures have a common feature that is with a regular topology. Here a regular topology is not a mathematical or definite conception, which means a non-arbitrary network fabric, or an inerratic network fabric. In a regular topology, each network node such as a switch or router can be addressed by its location and through a node's address, the node's connections to its neighbors in a network can be determined, and furthermore, the route to the node from other nodes in the network can be determined. So nodes can compute route entries without learning topology.

This draft proposes a generic routing method and protocol, fault-avoidance routing (FAR) protocol, for DCNs. This method leverages the regularity in the topologies of data center networks to simplify routing learning and accelerate the query of routing tables. This routing method has a better fault tolerance and can be applied to any DCN with a regular topology.

FAR is not a routing protocol to replace generic routing protocols such as OSPF and IS-IS. It cannot be used in general local networks whose topological structures are arbitrary, and whose scales are also not very large. OSPF works very well in such a network. But in a large-scale network with regular topology, FAR has a better performance. Compared with OSPF and IS-IS, FAR has shorter time of network convergence and lower PDU overhead. Furthermore, FAR requires less computing and storage resources, which let FAR routers to run at a lower cost of production than the generic routers.

In addition, for each type of network architecture, researchers designed a routing algorithm according to the features of its topology. Because these routing algorithms are different and lack compatibility with each other, it is very difficult to develop a routing protocol for network routers supporting multiple routing algorithms. FAR has better adaptability than these specified routing methods.

FAR consists of three components, i.e., link state learning unit, routing table building unit and routing table querying unit. In the link state learning unit, FAR exchanges link failures among routers to establish a consistent knowledge of the entire network. In this stage, the regularity in topology is exploited to infer failed links and routers. In the routing table building unit, FAR builds up two routing tables, i.e., a basic routing table (BRT) and a negative routing table (NRT), for each router according to the network topology and link states. In the last component, routers forward incoming packets by looking up the two routing tables. The matched entries in BRT minus the matched entries in NRT are the final route entries to be used to forward an incoming packet.

The remainder of this draft is organized as follows. The problem to be addressed by FAR is described in Section 3. The framework of FAR routing protocol is described in Section 4. Section 5 and 6 introduce FAR's data format FAR and modules in detail. Section 7 describe how FAR works by finite state machine (FSM). In Section 8, we discussed how FAR works with variable network architectures. Section 9 takes Fat-tree network as an example to illuminate how FAR works.

1.1. Acronyms & Definitions

DCN - Data Center Network
FAR - Fault-Avoidance Routing
BRT - Basic Routing Table
NRT - Negative Routing Table
NDT - Neighbor Devices Table
ADT - All Devices Table
LFT - Link Failure Table
DA - Device Announcement

LFA - Link Failure Announcement

DLR - Device and Link Request

IP - Internet Protocol

UDP - User Datagram Protocol

VM - Virtual Machine

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]. In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC-2119 significance.

3. Problem Statement

The problem to be addressed by FAR as proposed in this draft is described in this section. The expansion of Cloud data center networks has brought significant challenges to the existing routing technologies. FAR mainly solves a series of routing problems faced by large-scale data center networks.

3.1. The Impact of Large-scale Networks on Route Calculation

In a large-scale cloud data center network, there may be thousands of routers. Running OSPF and other routing protocols in such network will encounter these two challenges: a) Network convergence time would be too long, which will cause a longer time to elapse for creating and updating the routes. The response time to network failures may be excessively long; b) a large number of routing protocol packets need to be sent. The routing information consumes too much network bandwidth and CPU resources, which easily leads to packet loss and makes the problem (a) more prominent.

In order to solve these problems, a common practice is to partition a large network into some small areas, where the route calculation runs independently within different areas. However, nowadays the cloud data centers typically require very large internal bandwidth. To meet this requirement, a large number of parallel equivalent links are deployed in a network, such as a Fat-tree network. Partitioning such a network will affect the utilization of routing algorithm on equivalent multi-path and reduce internal network bandwidth requirements.

In the FAR routing calculation process, a Basic Routing Table (BRT) is built on local network topology leveraging the regularity of the network topologies. In addition to BRT, FAR also builds a Negative Routing Table (NRT). FAR gradually builds NRT in the process of learning network link failure information, which does not require learning a complete network fault information. FAR does not need to wait for the completion of the network convergence in the process of building these two tables. Therefore, it avoids the problem of excessive network convergence overheads in the route calculation process. In addition, FAR only needs to exchange a small amount of link change information between routers, and hence consumes less network bandwidth.

3.2. Dilemma of conventional routing methods in a large-scale network with giant number nodes of routers

There are many real world scenario where tens of thousands of nodes (or much more nodes) need to be deployed in a flat area, such as infiniband routing and switching system, high-performance computer network, and many IDC networks in China. The similar problems have been existed long ago. People have solved the problems through similar solutions, such as the traditional regular topology-based RFC3619 protocol, the routing protocols of infiniband routing and switching system, and high-performance computer network routing protocol.

Infiniband defines a switch-based network to interconnect processing nodes and the I/O nodes. Infiniband can support very large scale networks, use the regularity in topology to simplify its routing algorithm, which is just the same to what we do in FAR.

Why OSPF and other conventional routing methods do not work well in a large-scale network with giant number nodes of routers?

As everyone knows, the OSPF protocol uses multiple databases, more topological exchange information (as seen in the following example) and complicated algorithm. It requires routers to consume more memory and CPU processing capability. But the processing rate of CPU on the protocol message per second is very limited. When the network expands, CPU will quickly approach its processing limits, and at this time OSPF can not continue to expand the scale of the management. The SPF algorithm itself does not thoroughly solve these problems.

On the contrary, FAR does not have the convergence time delay and the additional CPU overheads, which SPF requires. Because in the initial stage, FAR already knows the regular information of the whole network topology and does not need to periodically do SPF operation.

One of the examples of "more topological exchange information":

In the OSPF protocol, LSA floods every 1800 seconds. Especially in the larger network, the occupation of CPU and band bandwidth will soon reach the router's performance bottleneck. In order to reduce these adverse effects, OSPF introduced the concept of Area, which still has not solved the problem thoroughly. By dividing the OSPF Area into several areas, the routers in the same area do not need to know the topological details outside their area. (In comparison with FAR, after OSPF introducing the concept of Area, the equivalent paths cannot be selected in the whole network scope)

OSPF can achieve the following results by Area : 1) Routers only need to maintain the same link state databases as other routers within the same Area, without the necessity of maintaining the same link state database as all routers in the whole OSPF domain. 2) The reduction of the link state databases means dealing with relatively fewer LSA, which reduces the CPU consumption of routers; 3) The large number of LSAs flood only within the same Area. But, its negative effect is that the smaller number of routers which can be managed in each OSPF area. On the contrary, because FAR does not have the above disadvantages, FAR can also manage large-scale network even without dividing Areas.

The aging time of OSPF is set in order to adapt to routing transformation and protocol message exchange happened frequently in the irregular topology. Its negative effect is: when the network does not change, the LSA needs to be refreshed every 1800 seconds to reset the aging time. In the regular topology, as the routings are fixed, it does not need the complex protocol message exchange and aging rules to reflect the routing changes, as long as LFA mechanism in the FAR is enough.

Therefore, in FAR, we can omit many unnecessary processing and the packet exchange. The benefits are fast convergence speed and much larger network scale than other dynamic routing protocol. Now there are some successful implementations of simplified routings in the regular topology in the HPC environment. Conclusion: As FAR needs few routing entries and the topology is regular, the database does not need to be updated regularly. Without the need for aging, there is no need for CPU and bandwidth overhead brought by LSA flood every 30 minutes, so the expansion of the network has no obvious effect on the performance of FAR, which is contrary to OSPF.

Comparison of convergence time: The settings of OSPF `spf_delay` and `spf_hold_time` can affect the change of convergence time. The convergence time of the network with 2480 nodes is about 15-20 seconds; while the FAR does not need to calculate the SFP, so there is no such convergence time.

These issues still exist in rapid convergence technology of OSPF and ISIS (such as I-SPF). The convergence speed and network scale constraint each other. FAR does not have the above problems, and the convergence time is almost negligible.

Can FRR solve these problems? IP FRR has some limitations. The establishment of IP FRR backup scheme will not affect the original topology and traffic forwarding which are established by protocol, however, we can not get the information of whereabouts and status when the traffic is switched to an alternate next hop.

3.3. Network Addressing Issues

Routers are typically configured with multiple network interfaces, each connected to a subnet. OSPF and other routing algorithms require that each interface of a router must be configured with an IP address. A large-scale data center network may contain thousands of routers and each router has dozens of network interfaces, thus, there are tens of thousands of IP addresses needed to be configured in a data center. It will be very complex to configure and manage a large number of network interfaces and will be difficult to troubleshoot network problems, then network maintenance will be costly and error-prone.

In FAR, the device position information is encoded in the IP address of the router. Each router only needs to be assigned a unique IP address according its location, which greatly solves complex network addressing issues in large-scale networks.

3.4. Big Routing Table Issues

There are a large number of subnets in the large-scale data center network. A router may build a routing entry for each subnet, and therefore the size of routing tables on each router may be very large. It will increase a router's cost and reduce the querying speed of the routing table.

FAR uses two measures to reduce the size of its routing tables: a) It builds a BRT on the regularity of the network topologies; b) It introduces a new routing table, i.e., a NRT. In this way FAR can reduce the size of routing tables to only a few dozen routing entries.

3.5. Adaptivity Issues for Routing Algorithms

To implement efficient routing in large-scale datacenters, besides FAR, some other routing methods are proposed for some specific network architectures, such as Fat-tree and BCube. These routing methods are different (from both design and implementation viewpoints)

and not compatible with the conventional routing methods, which brings big troubles to network equipment providers to develop new routers supporting various new routing methods.

FAR is a generic routing method. With slight modification, FAR method can be applied to most of regular datacenter networks. Furthermore, the structure of routing tables and querying a routing table in FAR are the same as conventional routing method. If FAR is adopted, the workload of developing a new type of router will be significantly decreased.

3.6. Virtual Machine Migration Issues

Supporting VM migration is very important for cloud-based datacenter networks. However, in order to support layer-3 routing, routing methods including OSPF and FAR require limiting VM migration within a subnet. For this paradox, the mainstream methods still utilize layer-3 routing on routers or switches, transmit packets encapsulated by IPinIP or MACinIP between hosts by tunnels passing through network to the destination access switch, and then extract original packet out and send it to the destination host.

By utilizing the aforementioned methods, FAR can be applied to Fat-tree, MatrixDCN or BCube networks for supporting VM migration in entire network.

4. The FAR Framework

FAR requires that a DCN has a regular topology, and network devices, including routers, switches, and servers, are assigned IP addresses according to their locations in the network. In other word, we can locate a device in the network according to its IP address.

FAR is a distributed routing method. In order to support FAR, each router needs to have a routing module that implements the FAR algorithm. FAR algorithm is composed of three parts, i.e., link-state learning, routing table building and routing table querying, as shown in Fig. 1.

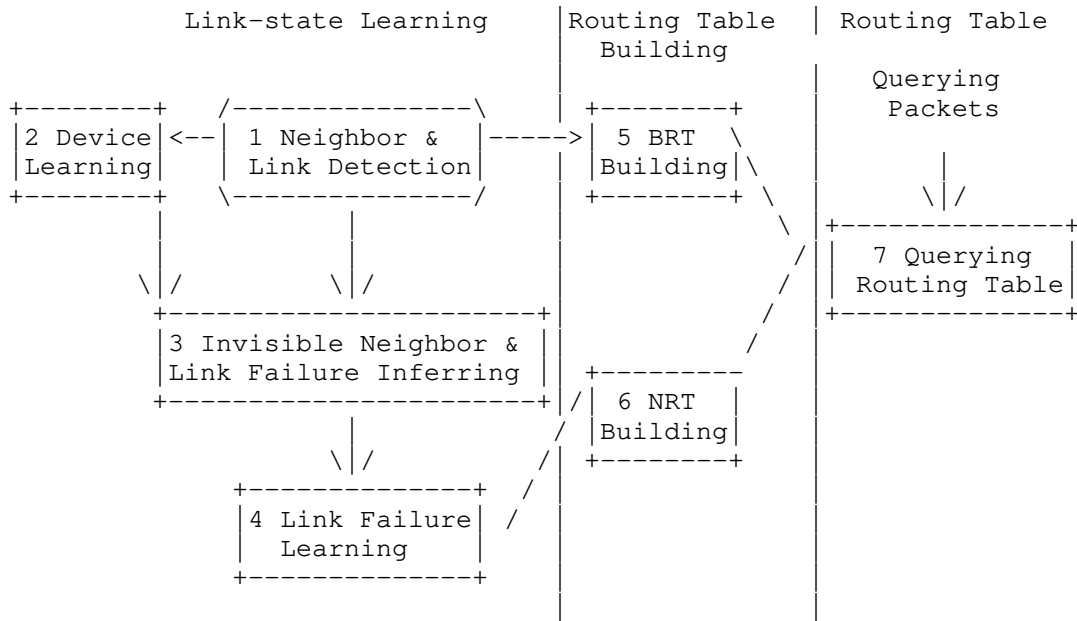


Figure 1: The FAR framework

Link-state learning is responsible for a router to detect the states of its connected links and learn the states of all the other links in the entire network. The second part builds two routing tables, a basic routing table (BRT) and an negative routing table (NRT), according to the learned link states in the first part. The third part queries the BRT and the NRT to decide a next forwarding hop for the received (ingress) packets.

5. Data Format

5.1. Data Tables

Some data tables are maintained on each router in FAR. They are:

Neighbor Device Table (NDT): To store neighbor routers and related links.

All Devices Table (ADT): To store all routers in the entire network.

Link Failures Table (LFT): To store all link failures in the entire network.

Basic Routing Table (BRT): To store the candidate routes.

Negative Routing Table (NRT): To store the avoiding routes.

The format of NRT

```
-----
Device ID | Device IP | Port ID | Link State | Update Time
-----
```

Device ID: The ID of a neighbor router.

Device IP: The IP address of a neighbor router.

Port ID: The port ID that a neighbor router is attached to.

Link State: The state of the link between a router and its neighbor router. There are two states: Up and Down.

Update Time: The time of updating the entry.

The format of ADT

```
-----
Device ID | Device IP | Type | State | Update Time
-----
```

Device ID: The ID of a neighbor router.

Device IP: The IP address of a neighbor router.

Type: The type of a neighbor router.

State: The state of a neighbor router. There are two states: Up and Down.

Update Time: The time of updating the entry.

The format of LFT

```
-----
No | Router 1 IP | Router 2 IP | Timestamp
-----
```

No: The entry number.

Router 1 IP: The IP address of one router that a failed link connects to.

Router 2 IP: The IP address of another router that a failed link connects to.

Timestamp: It identifies when the entry is created.

The format of BRT

```
-----
Destination | Mask | Next Hop | Interface | Update Time
-----
```

Destination: A destination network

Mask: The subnet mask of a destination network.

Next Hop: The IP address of a next hop for a destination.

Interface: The interface related to a next hop.

Update Time: The time of updating the entry.

The format of NRT

```
-----
Destination| Mask| Next Hop| Interface| Failed Link No| Timestamp
-----
```

Destination: A destination network.

Mask: The subnet mask of a destination network.

Next Hop: The IP address of a next hop that should be avoided for a destination.

Interface: The interface related to a next hop that should be avoided.

Failed Link No: A group of failed link numbers divided by "/", for example 1/2/3.

Timestamp: The time of updating the entry.

5.2. Messages

Some protocol messages are exchanged between routers in FAR.

Hello Message: This message is exchanged between neighbor routers to learn adjacency.

Device Announcement (DA): Synchronize the knowledge of routers between routers.

Link Failure Announcement (LFA): Synchronize link failures between routers.

Device and Link Request (DLR): When a router starts, it requests the knowledge of routers and links from its neighbors by a DLR message.

A FAR Message is directly encapsulated in an IP packet. The protocol field of IP header indicates an IP packet is an FAR message. The protocol of IP for FAR should be assigned by IANA.

The four types of FAR messages have same format of packet header, called FAR header (as shown in Figure 2).

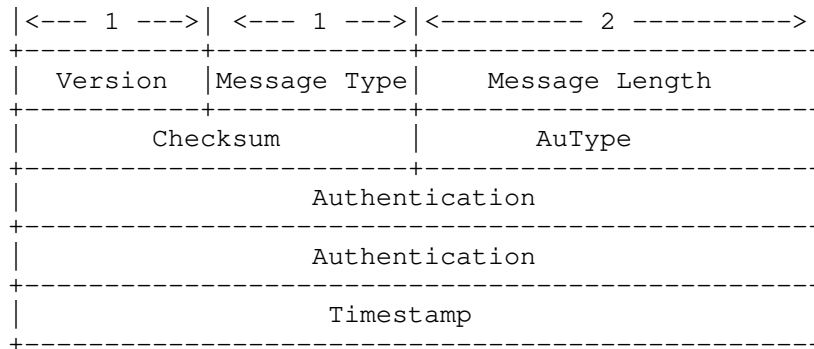


Figure 2: The format of FAR header

Version: FAR version

Message Type: The type of FAR message.

Packet Length: The packet length of the total FAR message.

Checksum: The checksum of an entire FAR message.

AuType: Authentication type. 0: no authentication, 1: Plaintext Authentication, 2: MD5 Authentication.

Authentication: Authentication information. 0: undefined, 1: Key, 2: key ID, MD5 data length and packet number. MD5 data is appended to the backend of the packet.

AuType and Authentication can refer to the definition of OSPF packet.

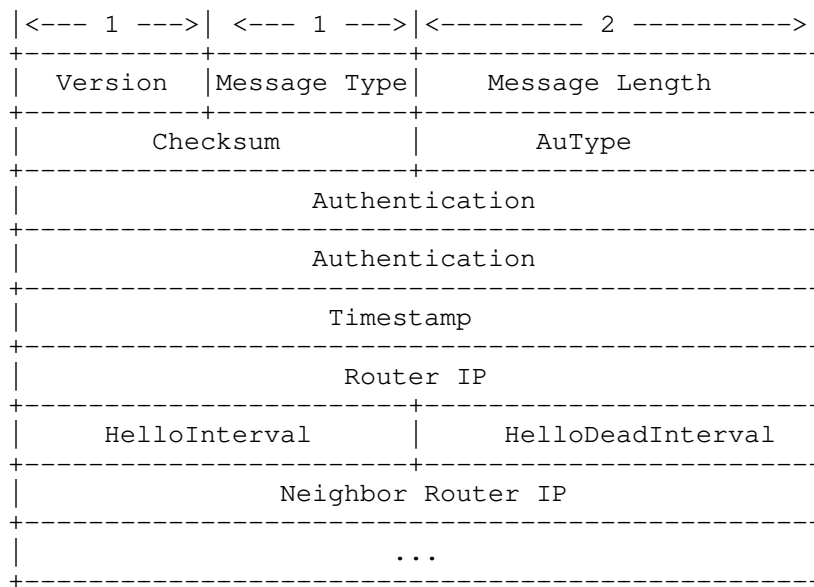


Figure 3: The Format of Hello Messages

For Hello messages, the Message Type in FAR header is set to 1. Besides FAR header, a Hello message (Fig. 3) requires the following fields:

Router IP: The router IP address.

HelloInterval: The interval of sending Hello messages to neighbor routers.

RouterDeadInterval: The interval to set a neighbor router dead (out-of-service). If in the interval time, a router doesn't receive a Hello message from its neighbor router, the neighbor router is treated as dead.

Neighbor Router IP: The IP address of a neighbor router. All the neighbor router's addresses should be included in a Hello message.

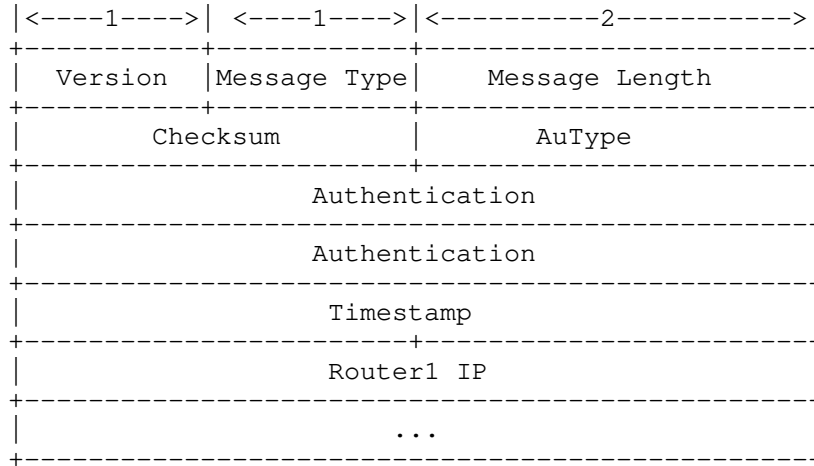


Figure 4: The Format of DA Messages

For DA messages (Fig. 4), the Message Type in FAR header is set to 2. Besides FAR header, a DA message includes IP addresses of all the announced routers.

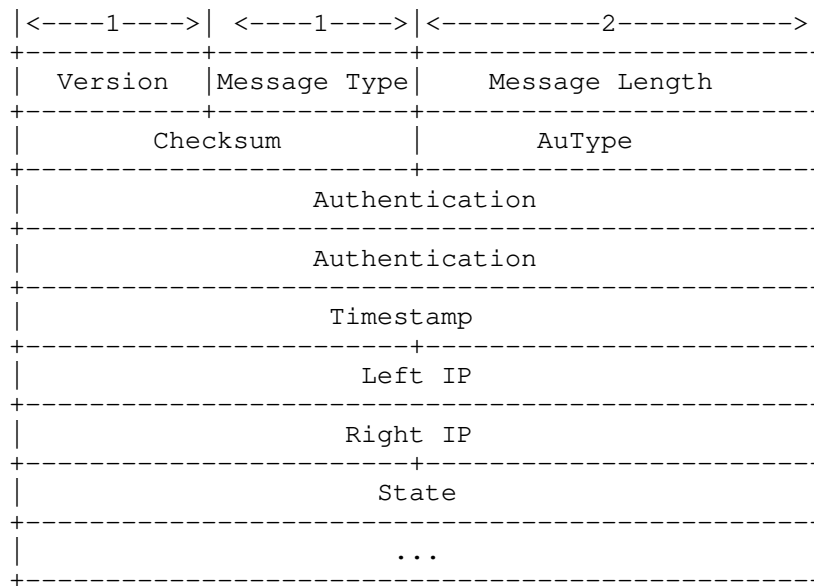


Figure 5: The Format of LFA Messages

For LFA messages (Fig. 5), the Message Type in FAR header is set to 3. Besides FAR header, a LFA message includes all the announced link failures.

Left IP: The IP address of the left endpoint router of a link.

Right IP: The IP address of the right endpoint router of a link.

State: Link state. 0: Up, 1: down

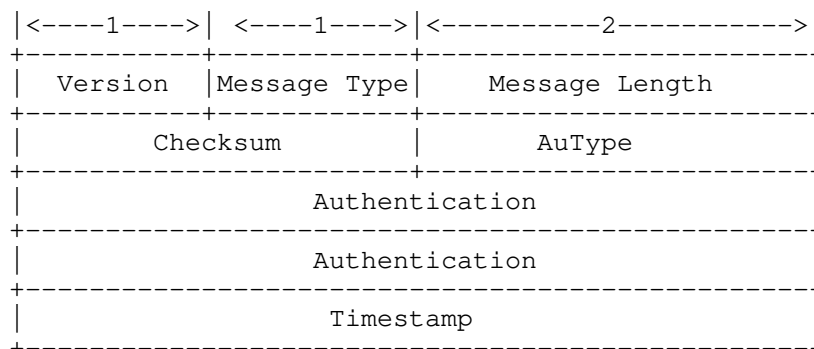


Figure 6: The Format of DLR Messages

For DLR messages (Fig. 6), the Message Type in FAR header is set to 1. Except for FAR header, DLR has no additional fields.

6. FAR Modules

6.1. Neighbor and Link Detection Module (M1)

M1 is responsible for sending and receiving Hello messages, and detecting directly-connected links and neighbor routers. Each Hello message is encapsulated in a UDP packet. M1 sends Hello messages periodically to all the active router ports and receives Hello messages from its neighbor routers. M1 detects neighbor routers and directly-connected links according to received Hello Messages and stores these neighbors and links into a Neighbor Devices Table (NDT). Additionally, M1 also stores neighbor routers into an All Devices Table (ADT).

6.2. Device Learning Module (M2)

M2 is responsible for sending, receiving, and forwarding device announcement (DA) messages, learning all the routers in the whole network, and deducing faulted routers. When a router starts, it sends a DA message announcing itself to its neighbors and a DLR message requesting the knowledge of routers and links from its neighbors. If M2 module of a router receives a DA message, it checks whether the router encapsulated in the message is in an ADT. If the router is not in the ADT, M2 puts this router into the ADT and forwards this DA message to all the active ports except for the incoming one, otherwise, M2 discards this message directly. If M2 module of a router receives a DLR message, it replies a DA message that encapsulates all of the learned routers.

6.3. Invisible Neighbor and Link Failure Inferring Module (M3)

M3 is responsible for inferring invisible neighbors of the current router by means of the ADT. If the link between a router A and its neighbor B breaks, which results in that M1 module of A cannot detect the existence of B, then B is an invisible neighbor of A. Since a device's location is coded into its IP address, it can be judged whether two routers are adjacent, according to their IP addresses. Based on this idea, M3 infers all of the invisible neighbors of the current router and the related link failures. The results are stored into a NDT. Moreover, link failures also are added into a link-failure table (LFT). LFT stores all of the failed links in the entire network.

6.4. Link Failure Learning Module (M4)

M4 is responsible for sending, receiving and forwarding link failure announcement (LFA) and learning all the link failures in the whole network. M4 broadcasts each newly inferred link failure to all the routers in the network. Each link failure is encapsulated in a LFA message and one link failure is broadcasted only once. If a router receives a DLR request from its neighbor, it will reply a LFA message that encapsulates all the learned link failures through M4 module. If M4 receives a LFA message, it checks whether the link failure encapsulated in the message is in a LFT by comparing two link ends and timestamp. If the link failure is not in the LFT or timestamp is different, M4 puts this link failure into the LFT (or update timestamp only) and forwards this LFA message to all the active ports except for the incoming one, otherwise, M4 discards this message directly.

There is a special case a router will rebroadcast a link failure. If a router receives a data packet and must forward the packet going ahead to destination through a failed link, it means some previous router should avoid this failed link according to its NRT but it doesn't. In this case, maybe the previous router missed the LFA message of the link failure due to some uncertain reasons. So the forwarding router rebroadcasts the LFA message.

6.5. BRT Building Module (M5)

M5 is responsible for building a BRT for the current router. By leveraging the regularity in topology, M5 can calculate the routing paths for any destination without the knowledge of the topology of whole network, and then build the BRT based on a NDT. Since the IP addresses of network devices are continuous, M5 only creates one route entry for a group of destination addresses that have the same network prefix by means of route aggregation technology. Usually,

the size of a BRT is very small. The detail of how to build a BRT is described in section 5.

6.6. NRT Building Module (M6)

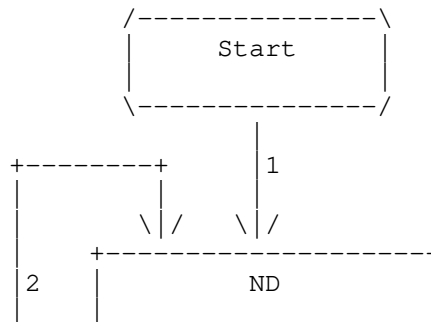
M6 is responsible for building a NRT for the current router. Because M5 builds a BRT without considering link failures in network, the routing paths calculated by the BRT cannot avoid failed links. To solve this problem, a NRT is used to exclude the routing paths that include some failed links from the paths calculated by a BRT. M6 calculate the routing paths that include failed links and stored them into the NRT. The details of how to build a NRT is described in section 5.

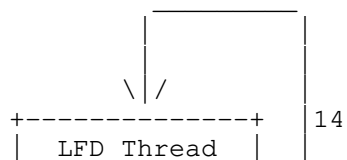
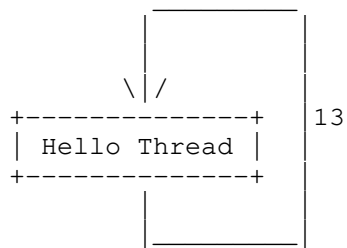
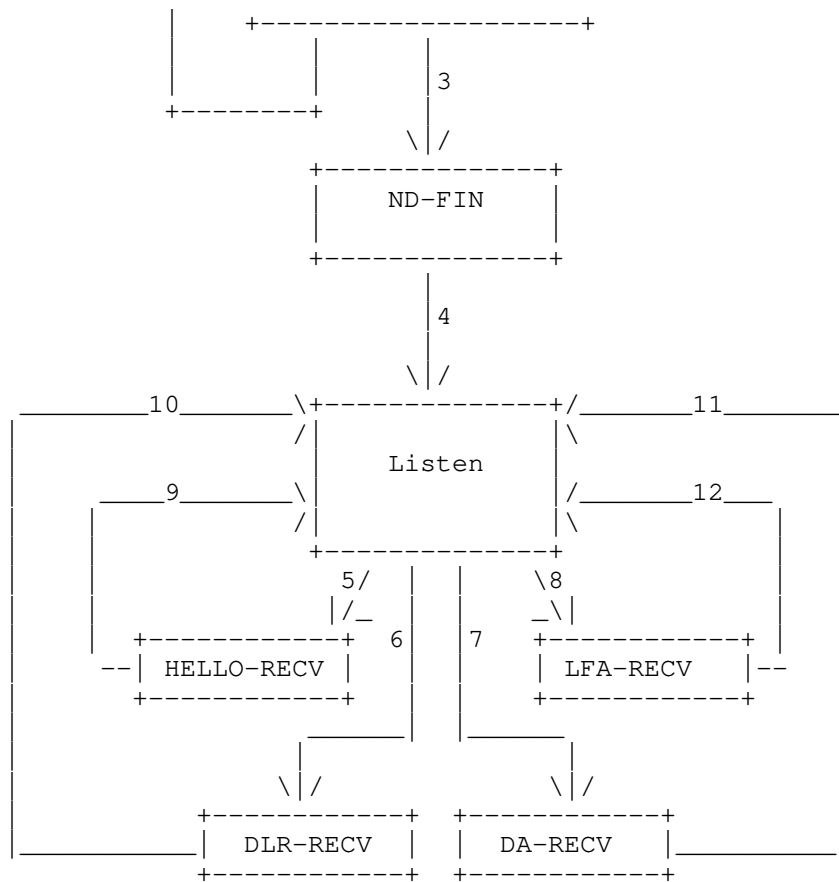
6.7. Routing Table Lookup (M7)

M7 is responsible for querying routing tables and selecting the next hop for forwarding the packets. Firstly, M7 takes the destination address of a forwarding packet as a criterion to look up route entries in a BRT based on longest prefix match. All of the matched entries are composed of a candidate hops list. Secondly, M7 look up negative route entries in a NRT taking the destination address of the forwarding packet as criteria. This lookup is not limited to the longest prefix match, any entry that matches the criteria would be selected and composed of an avoiding hops list. Thirdly, the candidate hops minus avoiding hops are composed of an applicable hops list. At last, M7 sends the forwarding packet to any one of the applicable hops. If the applicable list is empty, the forwarding packet will be dropped.

7. How a FAR Router Works

Figure 7 shows how a FAR router works by its FSM.





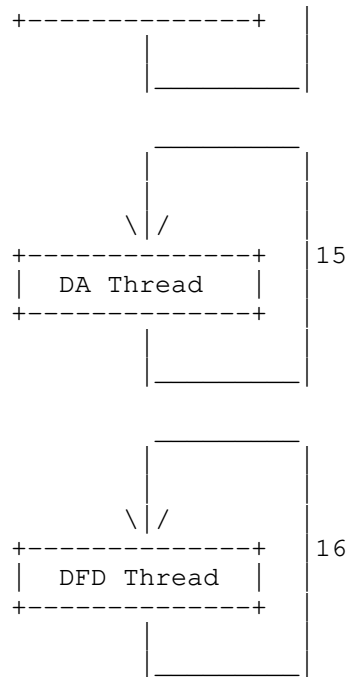


Figure 7: The Finite State Machine of FAR Router

- 1)When a router starts up, it starts a Hello thread and then starts ND (neighbor detection) timer (3 seconds). Next the router goes into ND (neighbor detection) state.
- 2)In the ND state, if a router received a Hello message, then it performs a Hello-message processing and goes back to the ND state.
- 3)When the ND timer is over, a router goes into ND-FIN (neighbor detection finished) state.
- 4)A router starts the LFD (link failure detection) thread and DFD (device failure detection) state, and sends DA message and DLR message to all of its active ports. Then the router goes into Listen state.
- 5) If a router receives a Hello message, then goes into HELLO-RECV state.
- 6) If a router receives a DLR message, then goes into DLR-RECV state.
- 7) If a router receives a DA message, then goes into DA-RECV state.
- 8) If a router receives a LFA message, then goes into LFA-RECV state.
- 9) A router performs the Hello-message processing. After that, it goes back to Listen state.
- 10) A router performs the DLR-message processing. After that, it goes back to Listen state.
- 11) A router performs the DA-message processing. After that, it goes back to Listen state.

- 12) A router performs the LFA-message processing. After that, it goes back to Listen state.
- 13) Hello thread produces and sends Hello messages to all its ports periodically.
- 14) LFD thread calls link-failure-detection processing to check link failures in all links periodically
- 15) DA thread produces and sends DA messages periodically (30 minutes).
- 16) When DFD thread starts up, it sleep a short time (30 seconds) to wait for a router learning all the active routers in the network. Then the thread calls the device-failure-detection processing to check device failures periodically (30 minutes).

8. Compatible Architecture

As a generic routing protocol, FAR can be run in various DCNs with regular topology. Up to now, we have implemented the FAR protocol for 4 types of DCN, including Fat-tree, BCube, MatrixDCN and Diamond.

For different network architectures, most processing of FAR is same besides calculation of routing tables. BRT routing tables are calculated based on Hello messages and NRT routing tables are calculated based on LFA messages in FAR. To extend FAR to support a new network architecture, only processing of Hello and LFA messages need providing to build BRT and NRT routing tables.

In this protocol, FAR can support maximally 12 network architectures and at least support 1 built-in network architecture, such as Fat-tree, BCube and MatrixDCN, etc. Each network architecture is assigned a unique number from 1 to 12. For example, if the 1 built-in architectures are assigned 1, and other customized architectures are assigned 2 to 12.

- 1: Fat-tree
- 2: BCube
- 3: MatrixDCN.
- 4: xxx.
-
- 12: xxx.

9. Application Example

In this section, we take a Fat-tree network(Fig. 7) as an example to describe how to apply FAR routing. Since M1 to M4 are very simple, we only introduce how the modules M5, M6, and M7 work in a Fat-tree network.

A Fat-tree network is composed of 4 layers. The top layer is core layer, and the other layers are aggregation layer, edge layer and

server layer. There are k pods, each one containing two layers of $k/2$ switches. Each k -port switch in the edge layer is directly connected to $k/2$ hosts. The remaining $k/2$ ports are connected to $k/2$ of the k -port switches in the aggregation layer. There are $(k/2)^2$ k -port core switches. Each core switch has one port connected to each of the k pods.

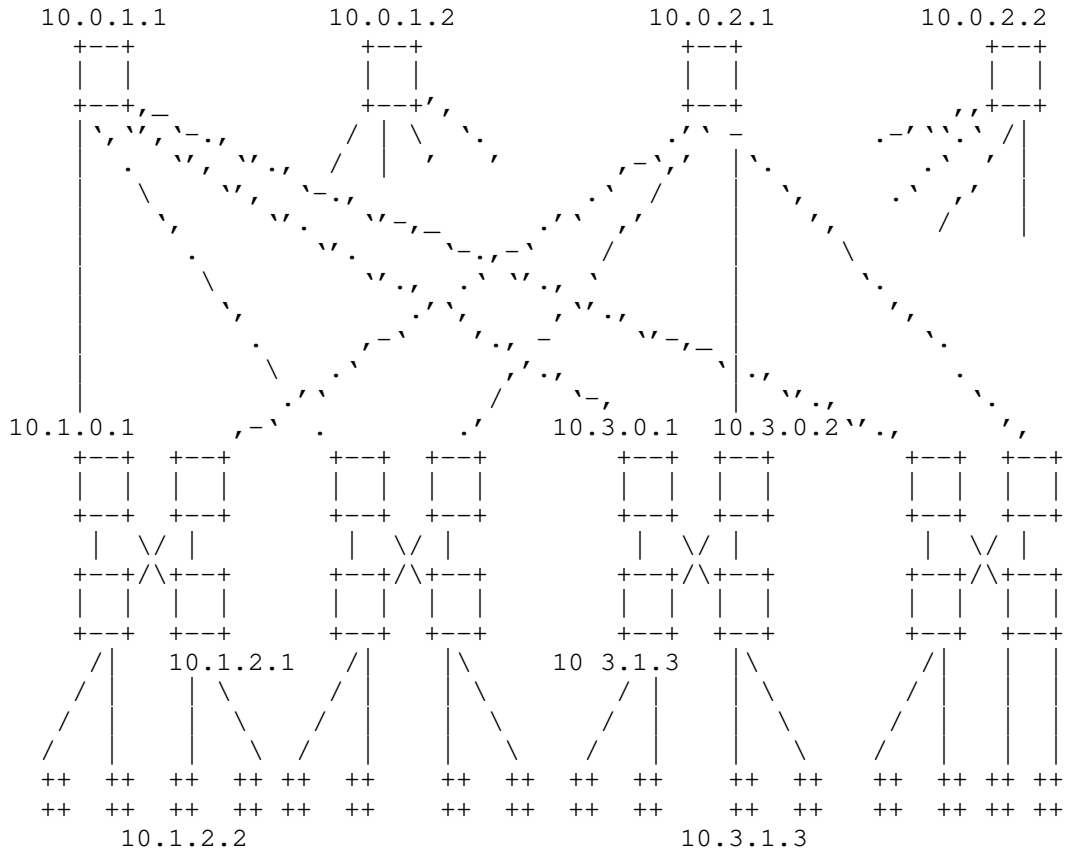


Figure 8: Fat-tree Network

Aggregation switches are given addresses of the form $10.\text{pod}.0.\text{switch}$, where pod denotes the pod number, and switch denotes the position of that switch in the upper pod (in $[1, k/2]$). Edge switches are given addresses of the form $10.\text{pod}.\text{switch}.1$, where pod denotes the pod number, and switch denotes the position of that switch in the lower pod (in $[1, k/2]$). The core switches are given addresses of the form $10.0.j.i$, where j and i denote that switch's coordinates in the

$(k/2) \times 2$ core switch grid (each in $[1, (k/2)]$, starting from top-left). The address of a host follows the pod switch to which it is connected to; hosts have addresses of the form: 10.pod.switch.ID, where ID is the host's position in that subnet (in $[2, k/2+1]$, starting from left to the right).

9.1. BRT Building Procedure

By leveraging the topology's regularity, every switch clearly knows how it forwards a packet. When a packet arrives at an edge switch, if the destination of the packet lies in the same subnet with the switch, then the switch directly forwards the packet to the destination server through layer-2 switching. Otherwise, the switch forwards the packet to any of aggregation switches in the same pod. When a packet arrives at an aggregation switch, if the destination of the packet lies in the same pod, the switch forwards the packet to the corresponding edge switch. Otherwise, the switch forwards the packet to any of core switches that it is connected to. If a core switch receives a packet, it forwards the packet to the corresponding aggregation switch that lies in the destination pod.

The forwarding policy discussed above is easily expressed through a BRT. The BRT of an edge switch, such as 10.1.1.1, is composed of the following entries:

Destination/Mask	Next hop
10.0.0.0/255.0.0.0	10.1.0.1
10.0.0.0/255.0.0.0	10.1.0.2

The BRT of an aggregation switch, such as 10.1.0.1, is composed of the following entries:

Destination/Mask	Next hop
10.1.1.0/255.255.255.0	10.1.1.1
10.1.2.0/255.255.255.0	10.1.2.1
10.0.0.0/255.0.0.0	10.0.1.1
10.0.0.0/255.0.0.0	10.0.1.2

The BRT of a core switch, such as 10.0.1.1, is composed of the following entries:

Destination/Mask	Next hop
10.1.0.0/255.255.0.0	10.1.0.1
10.2.0.0/255.255.0.0	10.2.0.1
10.3.0.0/255.255.0.0	10.3.0.1
10.4.0.0/255.255.0.0	10.4.0.1

9.2. NRT Building Procedure

The route entries in an NRT are related with link and node failures. We summarize all types of cases into three (3) catalogs.

9.2.1. Single Link Failure

In Fat-tree, Links can be classified as 3 types by their locations: 1) servers to edge switches; 2) edge to aggregation switches; 3) aggregation to core switches. Link failures between servers to edge switches only affect the communication of the corresponding servers and don't affect the routing tables of any switch, so we only discuss the second and third type of links failures.

Edge to Aggregation Switches

Suppose that the link between an edge switch, such as 10.1.2.1 (A), and an aggregation switch, such as 10.1.0.1(B), fails. This link failure may affect 3 types of communications.

- o Sources lie in the same subnet with A, and destinations do not. In this case, the link failure will only affect the routing tables of A. As this link is attached to A directly, A only needs to delete the route entries whose next hop is B in its BRT and add no entries to its NRT when A's M6 module detect the link failure.

- o Destinations lie in the same subnet with A, and sources lie in another subnet of the same pod. In this case, the link failure will affect the routing tables of all the edge switches in the same pod except for A. When an edge switch, such as 10.1.1.1, learns the link failure, it will add a route entry to its NRT:

Destination/Mask	Next hop
10.1.2.0/255.255.255.0	10.1.0.1

- o Destinations lie in the same subnet with A, sources lie in another pod. In this case, the link failure will affect the routing tables of all the edge switches in the other pods. When an edge switch in one other pod, such as 10.3.1.1, learns the link failure, because all the routings that pass through 10.3.0.1 to A will certainly pass

through the link between A and B, 10.3.1.1 need add a route entry to its NRT:

Destination/Mask	Next hop
10.1.2.0/255.255.255.0	10.3.0.1

Aggregation to Core Switches

Suppose that the link between an aggregation switch, such as 10.1.0.1 (A), and a core switch, such as 10.0.1.2(B), fails. This link failure may affect 2 types of communications.

- o Sources lie in the same pod (pod 1) with A, and destinations lie in the other pods. In this case, the link failure will only affect the routing tables of A. As this link is attached to A directly, A only need to delete the route entries whose next hop is B in its BRT and add no entries to its NRT when A's M6 module detect the link failure.

- o Destinations lie in the same pod (pod 1) with A, and sources lie in another pod. In this case, the link failure will affect the routing tables of all the aggregation switches in other pods except for pod 1. When an aggregation switch in one other pod, such as 10.3.0.1, learns the link failure, because all the routings that pass through 10.0.1.2 to the pod 1 where A lies will certainly pass through the link between A and B, 10.3.0.1 need add a route entry to its NRT:

Destination/Mask	Next hop
10.1.0.0/255.255.0.0	10.0.1.2

9.2.2. A Group of Link Failures

If all the uplinks of an aggregation switch fail, then this switch cannot forward packets, which will affect the routing of every edge switches. Suppose that all the uplinks of the node A (10.1.0.1) fail, it will affect two types of communications.

- o Sources lie in the same pod (pod 1) with A, and destinations lie in the other pods. In this case, the link failures will affect the routing of the edge switches in the Pod of A. To avoid the node A, each edge switch should remove the route entry "10.0.0.0/255.0.0.0 10.1.0.1" in which the next hop is the node A.

- o Destinations lie in the same pod (pod 1) with A, and sources lie in other pods. In this case, the link failures will affect the routing of edge switches in other pods. For example, if the edge switch 10.3.1.1 communicates with some node in the pod of A, it should avoid

the node 10.3.0.1, because any communication through 10.3.0.1 to the pod of A will pass through the node A. So a route entry should be added to 10.3.1.1:

Destination/Mask	Next hop
10.1.0.0/255.255.0.0	10.3.0.1

9.2.3. Node Failures

At last, we discuss the effect of node failures to a NRT. There are 3 types of node failures: the failure of edge, aggregation and core switches.

- o An edge switch fails. The failure doesn't affect the routing table of any switch.

- o A core switch fails. Only when all the core switches connected to the same aggregation switch fail, they will affect the routing of other switches. This case is equal to the case that all the uplinks of an aggregation switch fail, so the process of link failures can cover it.

- o An aggregation switch fails. This case is similar to the case that all the uplinks of an aggregation switch fail. It affects the routing of edge switches in other pods, but doesn't affect the routing of edge switches in pod of the failed switch. The process of this failure is same to the second case in section 6.2.2.

9.3. Routing Procedure

FAR decides a routing by looking up its BRT and NRT. We illuminate the routing procedure by an example. In this example, we suppose that the link between 10.3.1.1 and 10.3.0.2 and the link between 10.1.2.1 and 10.1.0.2 have failed. Then we look into the routing procedure of a communication from 10.3.1.3 (source) to 10.1.2.2 (destination).

Step 1: The source 10.3.1.3 sends packets to its default router 10.3.1.1

Step 2: The routing of 10.3.1.1.

1) Calculate candidate hops

10.3.1.1 looks up its BRT and gets the following matched entries:

Destination/Mask	Next hop
10.0.0.0/255.0.0.0	10.3.0.1

So the candidate hops = {10.3.0.1}

2) Calculate avoiding hops

Its NRT is empty, so the set of avoiding hop is empty too.

3) Calculate applicable hops

The applicable hops are candidate hops minus avoiding hops, so:

The applicable hops = {10.3.0.1}

4) Forward packets to 10.3.0.1

Step 3: The routing of 10.3.0.1

1) Calculate candidate hops.

10.3. 0.1 looks up its BRT and gets the following matched entries:

Destination/Mask	Next hop
10.1.0.0/255.255.0.0	10.0.1.1
10.1.0.0/255.255.0.0	10.0.1.2

So the candidate hops = {10.0.1.1, 10.0.1.2}

2) Calculate avoiding hops

Destination/Mask	Next hop
10.1.0.0/255.255.0.0	10.0.1.2

So the avoiding hops = {10.0.1.2}

3) Calculate applicable hops

The applicable hops are candidate hops minus avoiding hops, so:

The applicable hops = {10.0.1.1}

4) Forward packets to 10.0.1.1

Step 4: 10.0.1.1 forwards packets to 10.1.0.1 by looking up its routing tables.

Step 5: 10.1.0.1 forwards packets to 10.1.2.1 by looking up its routing tables.

Step 6: 10.1.2.1 forwards packets to the destination 10.1.2.2 by layer-2 switching.

9.4. FAR's Performance in Large-scale Networks

FAR has good performance to support large-scale networks. In this section, we take a Fat-tree network composed of 2,880 48-port switches and 27,648 servers as an example to show FAR's performance.

9.4.1. The number of control messages required by FAR

FAR exchanges a few messages between routers and only consumes a little network bandwidth. Tab. 1 shows the required messages in the example Fat-tree network.

Table 1: Required messages in a Fat-tree network.

Message Type	Scope	size(bytes)	Rate	Bandwidth
Hello	adjacent switches	less than 48	10 messages/sec	less than 4 kbps
DLR	adjacent switches	less than 48	(1)	48bytes
DA	entire network	less than 48	(2)	1.106M
LFA	entire network	less than 48	(3)	48 bytes

(1) Produce one when a router starts

(2) The number of switches(2,880) in a period

(3) Produce one when a link fails or recovers

9.4.2. The Calculating Time of Routing Tables

A BRT is calculated according to the states of its neighbor routers and attached links. An NRT is calculated according to device and link failures in the entire network. So FAR does not calculate network topology and has no problem of network convergence, which greatly reduces the calculating time of routing tables. The detection and spread time of link failures is very short in FAR. Detection time is up to the interval of sending Hello message. In FAR, the interval is set to 100ms, and a link failure will be detected in 200ms. The spread time between any pair of routers is

less than 200ms. If a link fails in a data center network, FAR can detect it, spread it to all the routers, and calculate routing tables in no more than 500ms.

9.4.3. The Size of Routing Tables

For the test Fat-tree network, the sizes of BRTs and NRTs are shown in Tab. 2.

Table 2: The size of routing tables in FAR

Routing Table	Core Switch	Aggregation Switch	Edge Switch
BRT	48	48	24
NRT	0	14	333

The BRT's size at a switch is determined by the number of its neighbor switches. In the example network, a core switch has 48 neighbor switches (aggregation switch), so it has 48 entries in its BRT. Only aggregation and edge switches have NRTs. The NRT size at a switch is related to the number of link failures in the network. Suppose that there are 1000 link failures in the example network, the number of failed links is 1.2% of total links, which is a very high failure ratio. We suppose that link failures are uniformly distributed in the entire network. The NRT size at an edge switch is about 333 and the NRT size of an aggregation switch is about 14 in average.

10. Security Considerations

The security considerations will be discussed in a future version of this document.

11. Conclusions

This draft introduces FAR protocol, a generic routing method and protocol, for data centers that have a regular topology. It uses two routing tables, a BRT and an NRT, to store the normal routing paths and the forbidden (to-be-avoided) routing paths, respectively. This makes the FAR protocol very simple and efficient. The sizes of these two tables are very small. Usually, a BRT has only several tens of entries and an NRT has only several or about a dozen entries.

12. Acknowledgments

This document is supported by ZTE Enterprise-University-Research Joint Project.

13. References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997.

[FAT-TREE] M. Al-Fares, A. Loukissas, and A. Vahdat."A Scalable,Commodity, Data Center Network Architecture",In ACM SIGCOMM 2008.

Authors' Addresses

Bin Liu
ZTE Inc., ZTE Plaza
No.19 East Huayuan Road,Hai Dian District
Beijing 100191
China

Phone: +86 -010-59932039
Email: 13683610386@139.com

Yantao Sun
Beijing Jiaotong University
No.3 Shang Yuan Cun, Hai Dian District
Beijing 100044
China

Email: ytsun@bjtu.edu.cn

Jing Cheng
Beijing Jiaotong University
No.3 Shang Yuan Cun, Hai Dian District
Beijing 100044
China

Email: journey.j@gmail.com

Yichen Zhang
Beijing Jiaotong University
No.3 Shang Yuan Cun, Hai Dian District
Beijing 100044
China

Email: snowfall_dan@sina.com

Bhumip Khasnabish
ZTE TX Inc.
55 Madison Avenue, Suite 160
Morristown, New Jersey 07960
USA

Phone: +001-781-752-8003
Email: vumip1@gmail.com, bhumip.khasnabish@ztetx.com
URI: <http://tinyurl.com/bhumip/>