LSR Working Group
Internet Draft
Intended status: Standards Track
Expires: April 2019

Dave Allan
Ericsson

October 2018

A Distributed Algorithm for Constrained Flooding of IGP Advertisements
draft-allan-lsr-flooding-algorithm-00

Abstract

This document describes a distributed algorithm that can be applied to the problem of constraining IGP flooding in dense mesh topologies. The flooding topology utilizes two node-diverse spanning trees in order to provide complete coverage in the presence of any single failure while constraining the number of LSAs received by any IGP speaker connected to the flooding topology.

Status of this Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

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

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

This Internet-Draft will expire in March 2019.

Copyright and License 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 (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

This memo describes an algorithm suitable for reducing the quantity of IGP flooding in dense mesh networks. The only property that the algorithm is dependent upon is that there are at least two equal and diverse shortest paths between any pair of IGP speakers in order to meet the requirements elucidated in [Li]. The algorithm uses a re-purposing of the tie breaking algorithm used in 802.1aq Shortest Path Bridging as an element of construction of the flooding topology. It is not the intention of this memo to specify a complete solution, but to offer a foundation of an eventual solution.

1.1. Authors

David Allan

1.2. 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].

2. Conventions used in this document

2.1. Terminology

Member Adjacency - An adjacency that has been determined to part of the flooding topology.

Member Node - A Participant node that is connected to the flooding topology.

Participant Adjacency - An adjacency between two participating nodes. It may be a member adjacency or a non-member adjacency

Non-Participant Adjacency - An adjacency where at least one of the two nodes is a not a Participating Node

Participating Node - An IGP speaker that has advertised the capability, and hence the intention, to participate in a flooding topology

3. Solution Overview

3.1. The Flooding Topology

A flooding topology is composed of a contiguously connected set of participating nodes.

The flooding topology constructed from two diversely rooted spanning trees. A participating node that is connected to the physical topology with a degree of two or greater and has at least two participating adjacencies will be bi-connected to the flooding topology.

The resulting flooding topology diameter will typically be two times the depth of the tree hierarchy. The compromise in this approach is that a subset of nodes in the network will not see a reduction of the replication burden from current practice when flooding LSAs as the degree of a subset of nodes in the flooding topology will correspond to the degree of the physical topology.

The protocol structure of flooded information is unmodified. A participant node may relay a received LSA onto member links of both spanning trees. Specific forwarding rules prevent undue flooding, the result being that every participant node that is bi-connected to the flooding topology will receive two copies of any flooded LSA in a fault free network. Participating nodes that due to network degradation are only singly connected will receive one copy. The forwarding rules are described in section 4.4.

3.2. Solution Applicability

This algorithm has been considered in the context of pure bipartite graphs, bipartite graphs modified with the addition of intra-tier adjacencies, and hierarchical variations of the above. Applicability to other network designs is for further study.

For all graphs the link costs are assumed to be common for all inter-tier links and common for any intra-tier links. Inter-tier and intra-tier links do not have to have the same cost.

3.3. Algorithm

The algorithm borrows from 802.1aq for the construction of the spanning trees used in this application. This is described in clause 28.5 of [802.1Q].

3.3.1. Algorithm Basics

The key component of the 802.1aq employed is the tie breaking algorithm. The original application of the algorithm was to produce a symmetrically congruent mesh of multicast trees and unicast forwarding whereby the path between any two nodes in the network was symmetric in both directions and congruent for both unicast and multicast traffic.

For this application the algorithm is used in the generation of two diversely rooted spanning trees that define the flooding topology.

As part of tree construction, the algorithm tie breaks between equal cost paths. When a tie is identified as part of a Dijkstra computation, a path-id is constructed for each equal cost path. A path-id is expressed as a lexicographically sorted list of the node-ids in the path. The set of equal cost paths is ranked, and the lowest selected. As an example:

Path-id 23-39-44-68-85 is ranked lower than

Path-id 23-44-59-63-90

When the path-ids are of unequal length, the path-ids with the fewest hops are ranked superior to the longer paths, and tie breaking is applied to select between the shorter path-ids. This is not expected to apply in the general case of the dense graphs this application is targeted at.

The node-ids used would be the loopback address of each node, therefore each path-id will be unique.

3.3.2. Generating Diverse Trees

The algorithm includes the concept of an "algorithm-mask", which is a value XOR'd with the node-ids prior to sorting into path IDs and ranking the paths. This permits the construction of diverse trees in a dense topology.

Two algorithm masks are used (zero and -1). When computing two trees from the same root, when there are at least two nodes to choose from at each distance from the root, fully diverse trees will be generated. When computing two trees from diverse roots in a tree architecture, diverse nodes will be selected in each tier in the hierarchy as the relay nodes to the next tier.

3.3.3. Desirable Properties Computation Wise

The algorithm has the property of permitting the pruning of intermediate state as a Dijkstra progresses as ties can be immediately evaluated, and the all but the selected path removed from further consideration. This is desirable when computing a Dijkstra in a dense graph as all path permutations do not need to be carried forward during computation. This permits the computation to be quite fast.

The resulting computational complexity would still be expressed as 2N(lnN).

4. Applying the Algorithm

4.1. Tree Generation

Each IGP speaker in the network has knowledge of each of the two spanning tree roots and the algorithm mask associated with each. This memo does not specify how root selection is performed and disseminated through the network, but does discuss selection requirements in section 4.5.

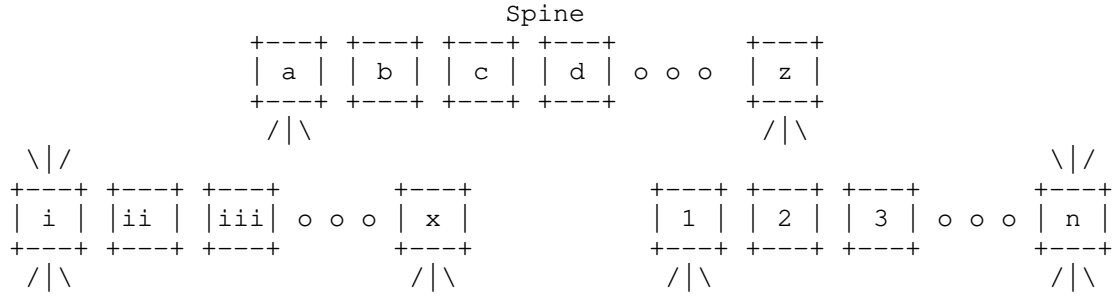
Each root has one of the two algorithm masks associated with it.

Each participating IGP speaker in the network computes a spanning tree from each the two roots (using the algorithm mask associated with each root) and from that can determine its own role in the flooding topology. The two spanning trees are designated the "low spanning tree" and the "high spanning tree".

The spanning trees are a starting point for a redundant topology. Unlike the commonly accepted operation of a spanning tree, in this application the distinction between upstream and downstream adjacencies is important and is an input to how a member node further relays any LSAs received. Upstream member adjacencies are in the direction of a root, and downstream member adjacencies are in the direction away from the root.

4.2. Illustrating the result

The following diagram illustrates the general layout of the flooding graph constructed using the algorithm as applied to a bi-partite style of tree (no intra tier links):



In the example, there are two tiers of switches. The spine (nodes a..z), and the next tier with two groups of nodes (i..x) and (1..n). The algorithm will select the node with the lowest node ID in each tier as the replicating node for the low spanning tree; 'a' and 'i' for the set of nodes connecting the spine and the next tier. The algorithm will select the nodes with highest node ID in the same set of nodes for the high spanning tree; 'z' and 'n' for the same set of nodes.

In the flooding topology:

- Node 'a' is connected to nodes i..x and 1..n for the low spanning tree.

- Node 'z' is connected to the same set of nodes for the high spanning tree.

- Node 'i' is connected to nodes 'a'..'z' for the low spanning tree, and

- Node 'n' is connected to the same nodes for the high spanning tree.

- All other nodes are bi-connected to the flooding topology

If there was a further tier added below nodes i..x, then 'i' and 'x' would be selected as the replicating nodes for the low and high spanning tree respectively. This is similarly true for nodes 1..n.

4.3. Interactions between Participating and Non-Participating Nodes

This solution proposes primarily only nodal behaviors with respect to constraining flooding to member adjacencies. To address the scenario

where the participating nodes were a subset of a larger network, it would be necessary to advertise the capability to participate in flood reduction.

This would then require that each participating node use this information to be able to identify the set of participating adjacencies and confine the spanning tree computation to the set of participating adjacencies in order to identify local set of member adjacencies. Interactions with non-participant adjacencies would conform to current practice.

4.4. Flooding of LSAs

The design of the protocol elements that are flooded is unmodified by this solution. Therefore, there is no additional information available to associate a received LSA with a given tree, nor is such information needed; the two spanning trees are not treated as unique entities in the flooding topology.

As per current practice, a node does not relay LSAs that it has already seen.

A new LSA received from an upstream member adjacency is flooded on:

- All downstream member adjacencies exclusive of the adjacency of arrival, irrespective of which tree the adjacencies are part of.

- All non-participant adjacencies

A new LSA received from a downstream member adjacency is flooded on:

- All other member adjacencies exclusive of the adjacency of arrival irrespective of which tree the adjacencies are part of.

- All non-participant adjacencies

A new LSA received from a member adjacency where upstream and downstream is ambiguous (it is an upstream member on one of the spanning trees and a downstream member on the other), is flooded on:

- All other member adjacencies exclusive of the adjacency of arrival irrespective of which adjacency the links are part of.

- All Non-Participant adjacencies

A new LSA received from a non-member adjacency is flooded on all member adjacency irrespective of which tree the adjacencies are part of (see sections 5.1 and 5.5).

4.5. Root Selection

The algorithm depends on tie breaking between sets of node IDs to produce diverse paths, therefore it does place some restrictions on root selection.

A root SHOULD be selected so that the root"s node-id when XORd with the associated algorithm mask is the lowest ranked node in the local tier in the tree hierarchy. This would be analogous to path-id ranking where the paths were all of length 1.

The root MUST NOT be selected such that the node-ID when XORd with the other root"s algorithm mask is the lowest ranked node. This would result in the root also being a transit node for the other spanning tree and produce a scenario whereby a single failure could render both spanning trees incomplete.

Roots MUST NOT be directly connected for either of the low or high spanning trees. If the topology does not permit this to be satisfied purely by root selection, then the inter-root adjacency must be pruned from the graph prior to spanning tree computation to ensure that diverse paths between the roots are used.

For a true bipartite graph, there are no other restrictions on node selection.

For a bipartite graph modified with inter-tier links, the roots MUST be placed in different tiers to ensure a pathological combination of link weights and node-ids does not result in a scenario where a single failure would render the flooding topology incomplete.

Other sources of failure may exist that may require an administrative component to root selection. This, for example, would ensure that both roots were not selected from a common shared risk group.

See also section 5.5.

4.6. Node Additions

A participating node that is added to the topology will initially not be served by the flooding topology. A participating node adjacent to that node is required to treat it as a non-participating node until such time as tree re-optimization has completed. At the end of tree

optimization, typically two adjacent participating nodes will have member adjacencies with the new node, so the ability to flood LSAs between the new node and the flooding topology will have been uninterrupted during the process.

5. Further work

5.1. Thoughts on Coexistence in the Context of a Larger Network

A node that had a combination of participating and non-participating adjacencies would be required to do the following:

- For any new LSA received on a participating adjacency, in addition to the rules for member adjacencies, it would also flood the LSA on all non-participating adjacencies.

- For any new LSA received on a non-participating adjacency, it would flood the LSA on all member adjacencies.

This is reflected in the forwarding rules described in section 4.4.

5.1.1. Multiple flooding Domains and the Severing of Flooding Domains

It is possible to envision several scenarios whereby there are sets of participating nodes that are not contiguously connected via participating adjacencies in a given IGP domain.

1. A node has been incorrectly configured as a participating node but has no participating adjacencies.

2. A participating node or set of nodes has become severed from the flooding topology but is still connected to other nodes in the network. Nodes in this set would still be able to compute a local extension of the flooding topology, but it would only be useful if the set was sufficiently large that a majority of the nodes were not connected to non-participants.

3. Procedures are designed to permit more than one flooding topology in an IGP domain. In which case participating nodes would have to be administratively configured to associate with a flooding topology instance.

5.2. Thoughts on Flooding Topology Re-Optimization

After a topology change, it is desirable that the flooding topology remain stable until the network has stabilized. However a single failure may render one of the spanning trees incomplete, such that a

further single failure could make the flooding topology incomplete. Therefore procedures should include re-optimization of the flooding topology after a topology change. In order to maintain complete coverage it would make sense not to recompute the spanning trees simultaneously.

One approach that would appear to make sense to separate in time network convergence, re-optimization of the low spanning tree and re-optimization of the high spanning tree.

The ideal would be to reoptimize an incomplete tree first, however this would require the participating nodes to maintain a complete map of all member adjacencies so that a common determination of the most degraded spanning tree and hence the order of re-optimization could be made.

5.3. Thoughts on Node and Network Initialization

A participating node at power up will be not be able to establish member links until it has synchronized with the network and the network is stable in the new topology. This suggests it simply treats power up similarly to how a topology change and network re-optimization is treated. The only difference being that it will flood all LSAs received or originated as per current practice until both spanning trees have stabilized.

5.4. Thoughts on Loop Prevention

802.1aq included additional mechanisms to prevent looping, a reverse path forwarding check, and digest exchange across adjacencies to ensure IGP synchronization.

Routing LSAs are not relayed if they are a duplicate, therefore destructive looping cannot occur and additional mitigation mechanisms are not required.

5.5. Thoughts on Pathological Failure Scenarios

While in a stable fault free network with sufficient mesh density of the types considered, the flooding topology used by this solution would ensure that no single failure rendered both spanning trees incomplete, it is also useful to consider multiple failure scenarios and if they can be mitigated.

Preliminary analysis suggests that in a tree network of sufficient mesh density, the only dual link failure that can render the flooding topology incomplete is if a participant node has failures in both

upstream member adjacencies. This can be partially mitigated if the node recognizes this scenario and reverts to flooding on all adjacencies. If the suggested procedures of 5.1.1 above are adopted, surrounding participating nodes that receive the LSA on a non-member adjacency will introduce the LSA into the flooding topology.

The pathological scenario is the simultaneous failure of both roots. This does suggest that root selection should place the roots two hops apart so there will be a constituency of participants that would observe a simultaneous failure of both upstream member adjacencies and revert to normal flooding.

6. Acknowledgements

The author would like to acknowledge Jerome Chiabaut for his original algorithm work that underpins this memo.

7. Security Considerations

For a future version of this document.

8. IANA Considerations

This memo requires no IANA allocations

9. References

9.1. Normative References

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

9.2. Informative References

[802.1Q] 802.1Q (2014) IEEE Standard for Local and Metropolitan Area Networks--Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks

[Li] Li, T., Psenak, P., "Dynamic Flooding on Dense Graphs", IETF work in progress, draft-li-dynamic-flooding-05, June 2018

10. Author's Address

Dave Allan
Ericsson
2455 Augustine Drive
Santa Clara, CA 95054
USA
Email: david.i.allan@ericsson.com

Networking Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 7, 2019

P. Psenak, Ed.
C. Filsfils
Cisco Systems
A. Bashandy
Individual
B. Decraene
Orange
Z. Hu
Huawei Technologies
March 6, 2019

IS-IS Extensions to Support Routing over IPv6 Dataplane
draft-bashandy-isis-srv6-extensions-05.txt

Abstract

Segment Routing (SR) allows for a flexible definition of end-to-end paths by encoding paths as sequences of topological sub-paths, called "segments". Segment routing architecture can be implemented over an MPLS data plane as well as an IPv6 data plane. This draft describes the IS-IS extensions required to support Segment Routing over an IPv6 data plane.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 September 7, 2019.

Copyright Notice

Copyright (c) 2019 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

With Segment Routing (SR) [I-D.ietf-spring-segment-routing], a node steers a packet through an ordered list of instructions, called segments.

Segments are identified through Segment Identifiers (SIDs).

Segment Routing can be directly instantiated on the IPv6 data plane through the use of the Segment Routing Header defined in [I-D.ietf-6man-segment-routing-header]. SRv6 refers to this SR instantiation on the IPv6 dataplane.

The network programming paradigm [I-D.filsfils-spring-srv6-network-programming] is central to SRv6. It describes how any function can be bound to a SID and how any network program can be expressed as a combination of SID's.

This document specifies IS-IS extensions that allow the IS-IS protocol to encode some of these functions.

Familiarity with the network programming paradigm [I-D.filsfils-spring-srv6-network-programming] is necessary to understand the extensions specified in this document.

This document defines one new top level IS-IS TLV and several new IS-IS sub-TLVs.

The SRv6 Capabilities sub-TLV announces the ability to support SRv6 and some Endpoint functions listed in Section 7 as well as advertising limitations when applying such Endpoint functions.

The SRv6 Locator top level TLV announces SRv6 locators - a form of summary address for the set of topology/algorithm specific SIDs associated with a node.

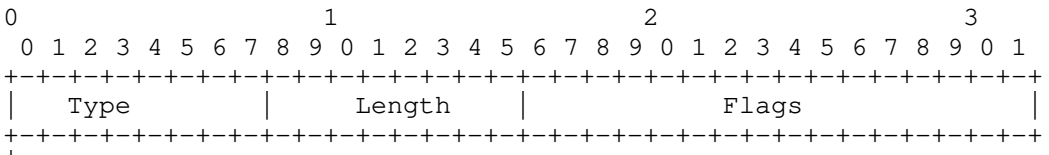
The SRv6 End SID sub-TLV, the SRv6 End.X SID sub-TLV, and the SRv6 LAN End.X SID sub-TLV are used to advertise which SIDs are instantiated at a node and what Endpoint function is bound to each instantiated SID.

2. SRv6 Capabilities sub-TLV

A node indicates that it has support for SRv6 by advertising a new SRv6- capabilities sub-TLV of the router capabilities TLV [RFC7981].

The SRv6 Capabilities sub-TLV may contain optional sub-sub-TLVs. No sub-sub-TLVs are currently defined.

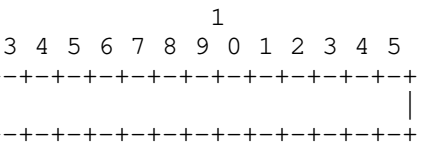
The SRv6 Capabilities sub-TLV has the following format:



Type: Suggested value 25, to be assigned by IANA

Length: 2 + length of sub-sub-TLVs

Flags: 2 octets The following flags are defined:



where:

O-flag: If set, the router supports use of the O-bit in the Segment Routing Header(SRH) as defined in [I-D.ali-spring-srv6-oam].

3. Advertising Supported Algorithms

SRv6 capable router indicates supported algorithm(s) by advertising the SR Algorithm TLV as defined in [I-D.ietf-isis-segment-routing-extensions].

4. Advertising Maximum SRv6 SID Depths

[I-D.ietf-isis-segment-routing-msd] defines the means to advertise node/link specific values for Maximum SID Depths (MSD) of various types. Node MSDs are advertised in a sub-TLV of the Router Capabilities TLV [RFC7981]. Link MSDs are advertised in a sub-TLV of TLVs 22, 23, 141, 222, and 223.

This document defines the relevant SRv6 MSDs and requests MSD type assignments in the MSD Types registry created by [I-D.ietf-isis-segment-routing-msd].

4.1. Maximum Segments Left MSD Type

The Maximum Segments Left MSD Type specifies the maximum value of the "SL" field [I-D.ietf-6man-segment-routing-header] in the SRH of a received packet before applying the Endpoint function associated with a SID.

SRH Max SL Type: 41 (Suggested value - to be assigned by IANA)

If no value is advertised the supported value is assumed to be 0.

4.2. Maximum End Pop MSD Type

The Maximum End Pop MSD Type specifies the maximum number of SIDs in the top SRH in an SRH stack to which the router can apply "PSP" or USP" as defined in [I-D.filsfils-spring-srv6-network-programming] flavors.

SRH Max End Pop Type: 42 (Suggested value - to be assigned by IANA)

If the advertised value is zero or no value is advertised then it is assumed that the router cannot apply PSP or USP flavors.

4.3. Maximum T.Insert MSD Type

The Maximum T.Insert MSD Type specifies the maximum number of SIDs that can be inserted as part of the "T.insert" behavior as defined in [I-D.filsfils-spring-srv6-network-programming].

SRH Max T.insert Type: 43 (Suggested value - to be assigned by IANA)

If the advertised value is zero or no value is advertised then the router is assumed not to support any variation of the "T.insert" behavior.

4.4. Maximum T.Encaps MSD Type

The Maximum T.Encaps MSD Type specifies the maximum number of SIDs that can be included as part of the "T.Encaps" behavior as defined in [I-D.filsfils-spring-srv6-network-programming] .

SRH Max T.encaps Type: 44 (Suggested value - to be assigned by IANA)

If the advertised value is zero then the router can apply T.Encaps only by encapsulating the incoming packet in another IPv6 header without SRH the same way IPinIP encapsulation is performed.

If the advertised value is non-zero then the router supports both IPinIP and SRH encapsulation subject to the SID limitation specified by the advertised value.

4.5. Maximum End D MSD Type

The Maximum End D MSD Type specifies the maximum number of SIDs in an SRH when performing decapsulation associated with "End.Dx" functions (e.g., "End.DX6" and "End.DT6") as defined in [I-D.filsfils-spring-srv6-network-programming].

SRH Max End D Type: 45 (Suggested value - to be assigned by IANA)

If the advertised value is zero or no value is advertised then it is assumed that the router cannot apply "End.DX6" or "End.DT6" functions if the extension header right underneath the outer IPv6 header is an SRH.

5. SRv6 SIDs and Reachability

As discussed in [I-D.filsfils-spring-srv6-network-programming], an SRv6 Segment Identifier (SID) is 128 bits and represented as

LOC:FUNCT

where LOC (the locator portion) is the L most significant bits and FUNCT is the 128-L least significant bits. L is called the locator length and is flexible. Each operator is free to use the locator length it chooses.

A node is provisioned with topology/algorithm specific locators for each of the topology/algorithm pairs supported by that node. Each locator is a covering prefix for all SIDs provisioned on that node which have the matching topology/algorithm.

Locators MUST be advertised in the SRv6 Locator TLV (see Section 6.1). Forwarding entries for the locators advertised in the SRv6 Locator TLV MUST be installed in the forwarding plane of receiving SRv6 capable routers when the associated topology/algorithm is supported by the receiving node.

Locators are routable and MAY also be advertised in Prefix Reachability TLVs (236 or 237).

Locators associated with algorithm 0 (for all supported topologies) SHOULD be advertised in a Prefix Reachability TLV (236 or 237) so that legacy routers (i.e., routers which do NOT support SRv6) will install a forwarding entry for algorithm 0 SRv6 traffic.

In cases where a locator advertisement is received in both in a Prefix Reachability TLV and an SRv6 Locator TLV, the Prefix Reachability advertisement MUST be preferred when installing entries in the forwarding plane. This is to prevent inconsistent forwarding entries on SRv6 capable/SRv6 incapable routers.

SRv6 SIDs are advertised as sub-TLVs in the SRv6 Locator TLV except for SRv6 End.X SIDs/LAN End.X SIDs which are associated with a specific Neighbor/Link and are therefore advertised as sub-TLVs in TLVs 22, 23, 222, 223, and 141.

SRv6 SIDs are not directly routable and MUST NOT be installed in the forwarding plane. Reachability to SRv6 SIDs depends upon the existence of a covering locator.

Adherence to the rules defined in this section will assure that SRv6 SIDs associated with a supported topology/algorithm pair will be forwarded correctly, while SRv6 SIDs associated with an unsupported topology/algorithm pair will be dropped. NOTE: The drop behavior depends on the absence of a default/summary route covering a given locator.

In order for forwarding to work correctly, the locator associated with SRv6 SID advertisements MUST be the longest match prefix installed in the forwarding plane for those SIDs. There are a number of ways in which this requirement could be compromised

- Another locator associated with a different topology/algorithm is the longest match
- A prefix advertisement (i.e., from TLV 236 or 237) is the longest match

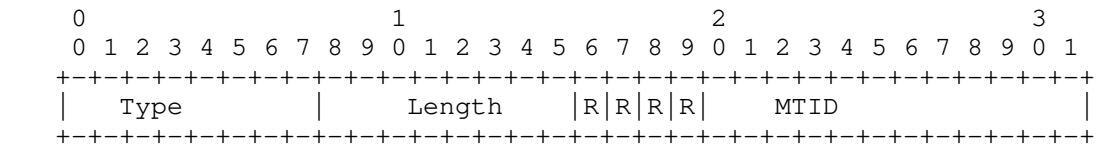
6. Advertising Locators and End SIDs

The SRv6 Locator TLV is introduced to advertise SRv6 Locators and End SIDs associated with each locator.

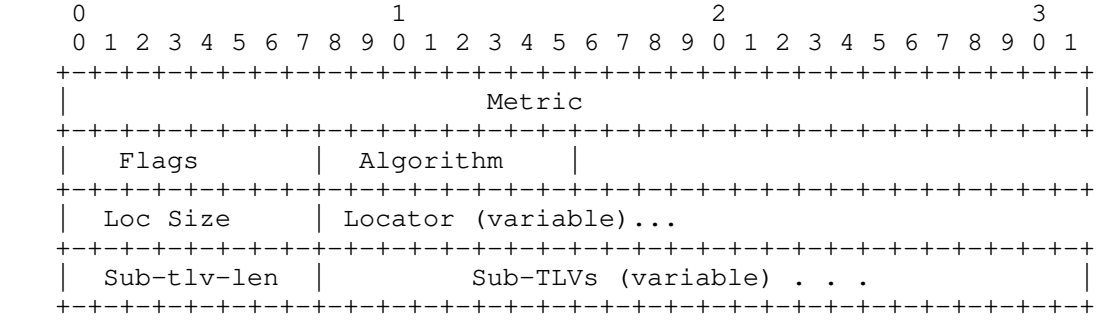
This new TLV shares the sub-TLV space defined for TLVs 135, 235, 236 and 237.

6.1. SRv6 Locator TLV Format

The SRv6 Locator TLV has the following format:



Followed by one or more locator entries of the form:



Type: 27 (Suggested value to be assigned by IANA)

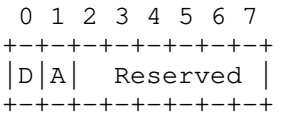
Length: variable.

MTID: Multitopology Identifier as defined in [RFC5120].
Note that the value 0 is legal.

Locator entry:

Metric: 4 octets. As described in [RFC5305].

Flags: 1 octet. The following flags are defined



where:
D bit: When the Locator is leaked from level-2 to level-1, the D bit MUST be set. Otherwise, this bit MUST be clear. Locators with the D bit set MUST NOT be leaked from level-1 to level-2.

This is to prevent looping.

A bit: When the Locator is configured as anycast, the A bit SHOULD be set. Otherwise, this bit MUST be clear.

The remaining bits are reserved for future use. They SHOULD be set to zero on transmission and MUST be ignored on receipt.

Algorithm: 1 octet. Associated algorithm. Algorithm values are defined in the IGP Algorithm Type registry.

Loc-Size: 1 octet. Number of bits in the Locator field. (1 - 128)

Locator: 1-16 octets. This field encodes the advertised SRv6 Locator. The Locator is encoded in the minimal number of octets for the given number of bits.

Sub-TLV-length: 1 octet. Number of octets used by sub-TLVs

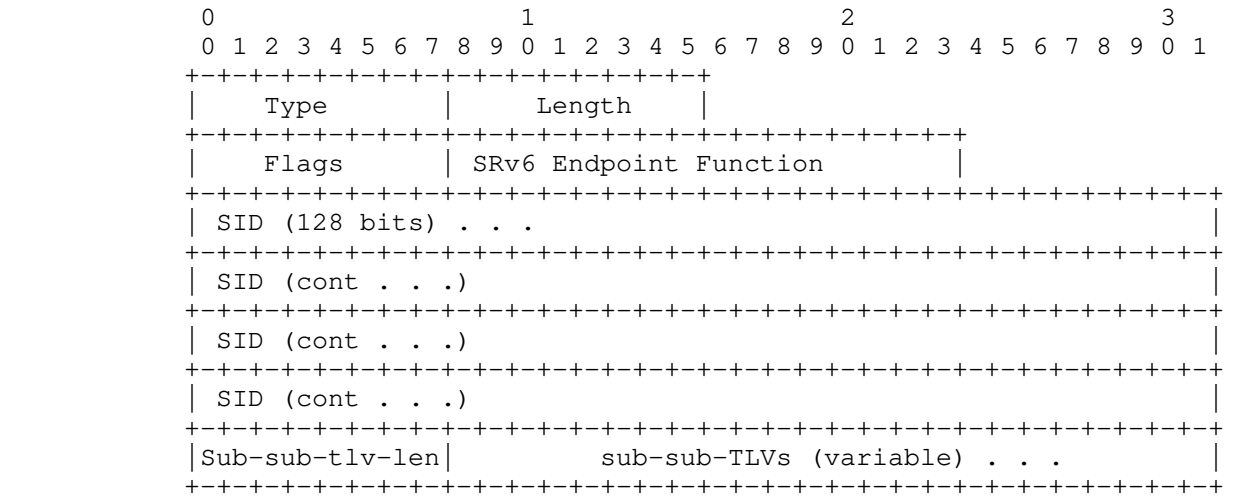
Optional sub-TLVs.

6.2. SRv6 End SID sub-TLV

The SRv6 End SID sub-TLV is introduced to advertise SRv6 Segment Identifiers (SID) with Endpoint functions which do not require a particular neighbor in order to be correctly applied [I-D.filsfils-spring-srv6-network-programming]. SRv6 SIDs associated with a neighbor are advertised using the sub-TLVs defined in Section 6.

This new sub-TLV is advertised in the SRv6 Locator TLV defined in the previous section. SRv6 End SIDs inherit the topology/algorithm from the parent locator.

The SRv6 End SID sub-TLV has the following format:



Type: 5 (Suggested value to be assigned by IANA)

Length: variable.

Flags: 1 octet. No flags are currently defined.

SRv6 Endpoint Function: 2 octets. As defined in
[I-D.filsfils-spring-srv6-network-programming]
Legal function values for this sub-TLV are defined in Section 7.

SID: 16 octets. This field encodes the advertised SRv6 SID.

Sub-sub-TLV-length: 1 octet. Number of octets used by sub-sub-TLVs

Optional sub-sub-TLVs

The SRv6 End SID MUST be a subnet of the associated Locator. SRv6 End SIDs which are NOT a subnet of the associated locator MUST be ignored.

Multiple SRv6 End SIDs MAY be associated with the same locator. In cases where the number of SRv6 End SID sub-TLVs exceeds the capacity of a single TLV, multiple Locator TLVs for the same locator MAY be advertised. For a given MTID/Locator the algorithm MUST be the same in all TLVs. If this restriction is not met all TLVs for that MTID/ Locator MUST be ignored.

7. Advertising SRv6 End.X SIDs

Certain SRv6 Endpoint functions [I-D.filsfils-spring-srv6-network-programming] must be associated with a particular neighbor, and in case of multiple layer 3 links to the same neighbor, with a particular link in order to be correctly applied.

This document defines two new sub-TLVs of TLV 22, 23, 222, 223, and 141 - namely "SRv6 End.X SID" and "SRv6 LAN End.X SID".

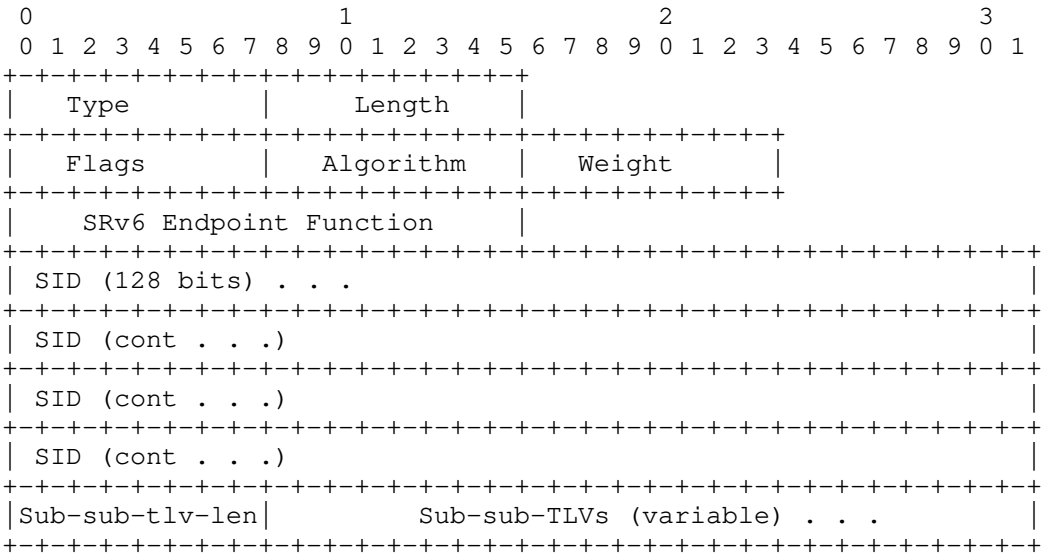
IS-IS Neighbor advertisements are topology specific - but not algorithm specific. End.X SIDs therefore inherit the topology from the associated neighbor advertisement, but the algorithm is specified in the individual SID.

All End.X SIDs MUST be a subnet of a Locator with matching topology and algorithm which is advertised by the same node in an SRv6 Locator TLV. End.X SIDs which do not meet this requirement MUST be ignored.

7.1. SRv6 End.X SID sub-TLV

This sub-TLV is used to advertise an SRv6 SID associated with a point to point adjacency. Multiple SRv6 End.X SID sub-TLVs MAY be associated with the same adjacency.

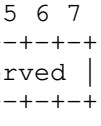
The SRv6 End.X SID sub-TLV has the following format:



Type: 43 (Suggested value to be assigned by IANA)

Length: variable.

Flags: 1 octet.



where:

B-Flag: Backup flag. If set, the End.X SID is eligible for protection (e.g., using IPFRR) as described in [RFC8355].

S-Flag. Set flag. When set, the S-Flag indicates that the End.X SID refers to a set of adjacencies (and therefore MAY be assigned to other adjacencies as well).

P-Flag. Persistent flag. When set, the P-Flag indicates that the End.X SID is persistently allocated, i.e., the End.X SID value remains consistent across router restart and/or interface flap.

Other bits: MUST be zero when originated and ignored when received.

Algorithm: 1 octet. Associated algorithm. Algorithm values are defined in the IGP Algorithm Type registry.

Weight: 1 octet. The value represents the weight of the End.X SID for the purpose of load balancing. The use of the weight is defined in [I-D.ietf-spring-segment-routing].

SRv6 Endpoint Function: 2 octets. As defined in [I-D.filsfils-spring-srv6-network-programming]
Legal function values for this sub-TLV are defined in Section 7.

SID: 16 octets. This field encodes the advertised SRv6 SID.

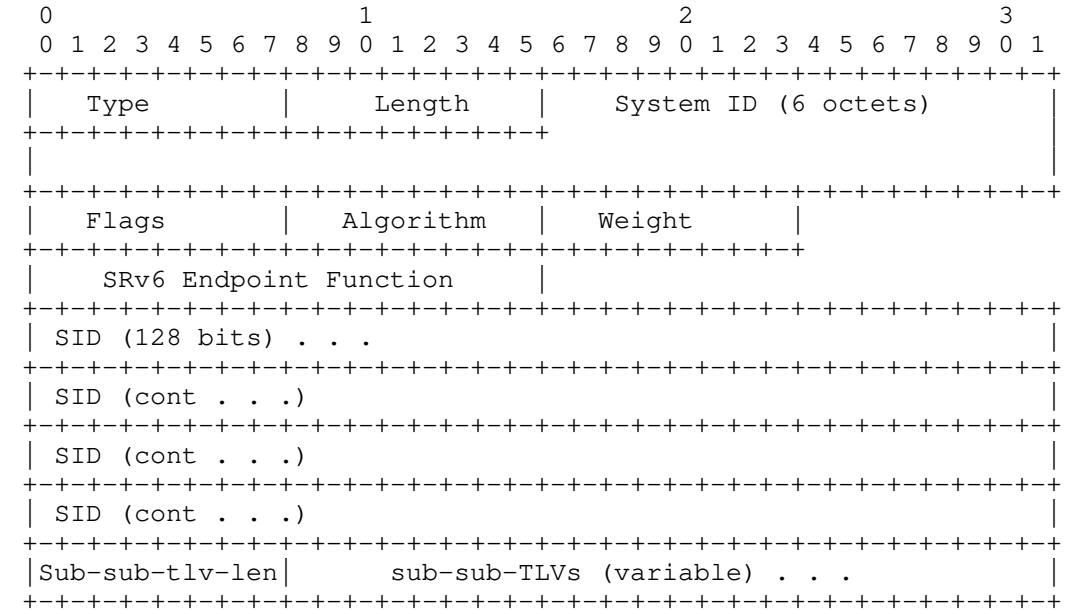
Sub-sub-TLV-length: 1 octet. Number of octets used by sub-sub-TLVs

Note that multiple TLVs for the same neighbor may be required in order to advertise all of the SRv6 End.X SIDs associated with that neighbor.

7.2. SRv6 LAN End.X SID sub-TLV

This sub-TLV is used to advertise an SRv6 SID associated with a LAN adjacency. Since the parent TLV is advertising an adjacency to the Designated Intermediate System(DIS) for the LAN, it is necessary to include the System ID of the physical neighbor on the LAN with which the SRv6 SID is associated. Given that a large number of neighbors may exist on a given LAN a large number of SRv6 LAN END.X SID sub-TLVs may be associated with the same LAN. Note that multiple TLVs for the same DIS neighbor may be required in order to advertise all of the SRv6 End.X SIDs associated with that neighbor.

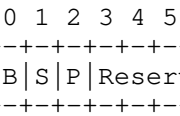
The SRv6 LAN End.X SID sub-TLV has the following format:



Type: 44 (Suggested value to be assigned by IANA)
Length: variable.

System-ID: 6 octets of IS-IS System-ID of length "ID Length" as defined in [ISO10589].

Flags: 1 octet.



where B,S, and P flags are as described in Section 6.1.
Other bits: MUST be zero when originated and ignored when received.

Algorithm: 1 octet. Associated algorithm. Algorithm values are defined in the IGP Algorithm Type registry.

Weight: 1 octet. The value represents the weight of the End.X SID for the purpose of load balancing. The use of the weight is defined in [I-D.ietf-spring-segment-routing].

SRv6 Endpoint Function: 2 octets. As defined in [I-D.filsfils-spring-srv6-network-programming]
Legal function values for this sub-TLV are defined in Section 7.

SID: 16 octets. This field encodes the advertised SRv6 SID.

Sub-sub-TLV-length: 1 octet. Number of octets used by sub-sub-TLVs.

8. Advertising Endpoint Behaviors

Endpoint behaviors are defined in [I-D.filsfils-spring-srv6-network-programming] and [I-D.ali-spring-srv6-oam]. The numerical identifiers for the Endpoint behaviors are defined in the "SRv6 Endpoint Behaviors" registry defined in [I-D.filsfils-spring-srv6-network-programming]. This section lists the Endpoint behaviors and their identifiers, which MAY be advertised by IS-IS and the SID sub-TLVs in which each type MAY appear.

Endpoint Behavior	Endpoint Behavior Identifier	End SID	End.X SID	Lan End.X SID
End (PSP, USP, USD)	1-4, 28-31	Y	N	N
End.X (PSP, USP, USD)	5-8, 32-35	N	Y	Y
End.T (PSP, USP, USD)	9-12, 36-39	Y	N	N
End.DX6	16	N	Y	Y
End.DX4	17	N	Y	Y
End.DT6	18	Y	N	N
End.DT4	19	Y	N	N
End.DT64	20	Y	N	N
End.OP	40	Y	N	N
End.OTP	41	Y	N	N

9. IANA Considerations

This document requests allocation for the following TLVs, sub- TLVs, and sub-sub-TLVs as well updating the ISIS TLV registry and defining a new registry.

9.1. SRv6 Locator TLV

This document adds one new TLV to the IS-IS TLV Codepoints registry.

Value: 27 (suggested - to be assigned by IANA)

Name: SRv6 Locator

This TLV shares sub-TLV space with existing "Sub-TLVs for TLVs 135, 235, 236 and 237 registry". The name of this registry needs to be changed to "Sub-TLVs for TLVs 27, 135, 235, 236 and 237 registry".

9.1.1. SRv6 End SID sub-TLV

This document adds the following new sub-TLV to the (renamed) "Sub-TLVs for TLVs 27, 135, 235, 236 and 237 registry".

Value: 5 (suggested - to be assigned by IANA)

Name: SRv6 End SID

This document requests the creation of a new IANA managed registry for sub-sub-TLVs of the SRv6 End SID sub-TLV. The registration procedure is "Expert Review" as defined in [RFC7370]. Suggested registry name is "sub-sub-TLVs for SRv6 End SID sub-TLV". No sub-sub-TLVs are defined by this document except for the reserved value.

0: Reserved

1-255: Unassigned

9.1.2. Revised sub-TLV table

The revised table of sub-TLVs for the (renamed) "Sub-TLVs for TLVs 27, 135, 235, 236 and 237 registry" is shown below:

Type	27	135	235	236	237
1	n	y	y	y	y
2	n	y	y	y	y
3	n	y	y	y	y
4	y	y	y	y	y
5	y	n	n	n	n
11	y	y	y	y	y
12	y	y	y	y	y

9.2. SRv6 Capabilities sub-TLV

This document adds the definition of a new sub-TLV in the "Sub- TLVs for TLV 242 registry".

Type: 25 (Suggested - to be assigned by IANA)

Description: SRv6 Capabilities

This document requests the creation of a new IANA managed registry for sub-sub-TLVs of the SRv6 Capability sub-TLV. The registration procedure is "Expert Review" as defined in [RFC7370]. Suggested registry name is "sub-sub-TLVs for SRv6 Capability sub-TLV". No sub-sub-TLVs are defined by this document except for the reserved value.

0: Reserved

1-255: Unassigned

9.3. SRv6 End.X SID and SRv6 LAN End.X SID sub-TLVs

This document adds the definition of two new sub-TLVs in the "sub-TLVs for TLV 22, 23, 25, 141, 222 and 223 registry".

Type: 43 (suggested - to be assigned by IANA)

Description: SRv6 End.X SID

Type: 44 (suggested - to be assigned by IANA)

Description: SRv6 LAN End.X SID

Type	22	23	25	141	222	223
43	y	y	y	y	y	y
44	y	y	y	y	y	y

9.4. MSD Types

This document defines the following new MSD types. These types are to be defined in the IGP MSD Types registry defined in [I-D.ietf-isis-segment-routing-msd] .

All values are suggested values to be assigned by IANA.

Type	Description
41	SRH Max SL
42	SRH Max End Pop
43	SRH Max T.insert
44	SRH Max T.encaps
45	SRH Max End D

10. Security Considerations

Security concerns for IS-IS are addressed in [ISO10589], [RFC5304], and [RFC5310].

11. Contributors

The following people gave a substantial contribution to the content of this document and should be considered as co-authors:

Stefano Previdi
Huawei Technologies
Email: stefano@previdi.net

Paul Wells
Cisco Systems
Saint Paul,
Minnesota
United States
Email: pauwells@cisco.com

Daniel Voyer
Email: daniel.voyer@bell.ca

Satoru Matsushima
Email: satoru.matsushima@g.softbank.co.jp

Bart Peirens
Email: bart.peirens@proximus.com

Hani Elmalky
Email: hani.elmalky@ericsson.com

Prem Jonnalagadda
Email: prem@barefootnetworks.com

Milad Sharif
Email: msharif@barefootnetworks.com>

Robert Hanzl
Cisco Systems
Millenium Plaza Building, V Celnici 10, Prague 1,
Prague, Czech Republic
Email rhanzl@cisco.com

Ketan Talaulikar
Cisco Systems, Inc.
Email: ketant@cisco.com

12. References

12.1. Normative References

[I-D.ali-spring-srv6-oam]
Ali, Z., Filsfils, C., Kumar, N., Pignataro, C., faiqbal@cisco.com, f., Gandhi, R., Leddy, J., Matsushima, S., Raszuk, R., daniel.voyer@bell.ca, d., Dawra, G., Peirens, B., Chen, M., and G. Naik, "Operations, Administration, and Maintenance (OAM) in Segment Routing Networks with IPv6 Data plane (SRv6)", draft-ali-spring-srv6-oam-02 (work in progress), October 2018.

[I-D.filsfils-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-filsfils-spring-srv6-network-programming-07 (work in progress), February 2019.

[I-D.ietf-6man-segment-routing-header]
Filsfils, C., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-16 (work in progress), February 2019.

[I-D.ietf-isis-segment-routing-extensions]
Previdi, S., Ginsberg, L., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-22 (work in progress), December 2018.

[I-D.ietf-isis-segment-routing-msd]
Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling MSD (Maximum SID Depth) using IS-IS", draft-ietf-isis-segment-routing-msd-19 (work in progress), October 2018.

[ISO10589]
Standardization", I. ". O. F., "Intermediate system to Intermediate system intra-domain routeing 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, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <https://www.rfc-editor.org/info/rfc5120>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC7370] Ginsberg, L., "Updates to the IS-IS TLV Codepoints Registry", RFC 7370, DOI 10.17487/RFC7370, September 2014, <https://www.rfc-editor.org/info/rfc7370>.

[RFC7981] Ginsberg, L., Previdi, S., and M. Chen, "IS-IS Extensions for Advertising Router Information", RFC 7981, DOI 10.17487/RFC7981, October 2016, <https://www.rfc-editor.org/info/rfc7981>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

12.2. Informative References

[I-D.ietf-spring-segment-routing] Filsfils, C., Previdi, S., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-15 (work in progress), January 2018.

[RFC8355] Filsfils, C., Ed., Previdi, S., Ed., Decraene, B., and R. Shakir, "Resiliency Use Cases in Source Packet Routing in Networking (SPRING) Networks", RFC 8355, DOI 10.17487/RFC8355, March 2018, <https://www.rfc-editor.org/info/rfc8355>.

Authors' Addresses

Peter Psenak (editor)
Cisco Systems
Pribinova Street 10
Bratislava 81109
Slovakia

Email: ppsenak@cisco.com

Clarence Filsfils
Cisco Systems
Brussels
Belgium

Email: cfilsfil@cisco.com

Ahmed Bashandy
Individual

Email: abashandy.ietf@gmail.com

Bruno Decraene
Orange
Issy-les-Moulineaux
France

Email: bruno.decraene@orange.com

Zhibo Hu
Huawei Technologies

Email: huzhibo@huawei.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 24, 2019

H. Chen
D. Cheng
Huawei Technologies
M. Toy
Verizon
Y. Yang
IBM

September 20, 2018

LS Flooding Reduction
draft-cc-ospf-flooding-reduction-04

Abstract

This document proposes an approach to flood link states on a topology that is a subgraph of the complete topology per underline physical network, so that the amount of flooding traffic in the network is greatly reduced, and it would reduce convergence time with a more stable and optimized routing environment. The approach can be applied to any network topology in a single area.

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 RFC 2119 [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 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 March 24, 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

For some networks such as dense Data Center (DC) networks, the existing Link State (LS) flooding mechanism is not efficient and may have some issues. The extra LS flooding consumes network bandwidth. Processing the extra LS flooding, including receiving, buffering and decoding the extra LSs, wastes memory space and processor time. This may cause scalability issues and affect the network convergence negatively.

This document proposes an approach to minimize the amount of flooding traffic in the network. Thus the workload for processing the extra LS flooding is decreased significantly. This would improve the scalability, speed up the network convergence, stable and optimize the routing environment.

This approach is also flexible. It has multiple modes for computation of flooding topology. Users can select a mode they prefer, and smoothly switch from one mode to another. The approach is applicable to any network topology in a single area. It is backward compatible.

2. Terminology

Flooding Topology:
: A sub-graph or sub-network of a given (physical) network topology that has the same reachability to every node as the given network topology, through which link states are flooded.

critical link or interface on a flooding topology:
: A only link or interface among some nodes on the flooding topology. When this link or interface goes down, the flooding topology will be split.

critical node on a flooding topology:

A only node connecting some nodes on the flooding topology. When this node goes down, the flooding topology will be split.

backup path:
A path or a sequence of links, when a critical link or node goes down, providing a connection to connect two parts of a split flooding topology. When a critical node goes down, the flooding topology may be split into more than two parts. In this case, two or more backup paths are needed to connect all the split parts into one.

Remaining Flooding Topology:
A topology from a flooding topology by removing the failed links and nodes from the flooding topology.

LSA:
A Link State Advertisement in OSPF.

LSP:
A Link State Protocol Data Unit (PDU) in IS-IS.

LS:
A Link Sate, which is an LSA or LSP.

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 [RFC2119].

4. Problem Statement

OSPF and IS-IS deploy a so-called reliable flooding mechanism, where a node must transmit a received or self-originated LS to all its interfaces (except the interface where an LS is received). While this mechanism assures each LS being distributed to every node in an area or domain, the side-effect is that the mechanism often causes redundant LS, which in turn forces nodes to process identical LS more than once. This results in the waste of link bandwidth and nodes' computing resources, and the delay of topology convergence.

This becomes more serious in networks with large number of nodes and links, and in particular, higher degree of interconnection (e.g., meshed topology, spine-leaf topology, etc.). In some environments such as in data centers, the drawback of the existing flooding mechanism has already caused operational issues, including repeated and waves of flooding storms, chock of computing resources, slow

convergence, oscillating topology changes, instability of routing environment.

One example is as shown in Figure 1, where Node 1, Node 2 and Node 3 are interconnected in a mesh. When Node 1 receives a new or updated LS on its interface I11, it by default would forward the LS to its interface Il2 and I13 towards Node 2 and Node 3, respectively, after processing. Node 2 and Node 3 upon reception of the LS and after processing, would potentially flood the same LS over their respective interface I23 and I32 toward each other, which is obviously not necessary and at the cost of link bandwidth as well as both nodes' computing resource.

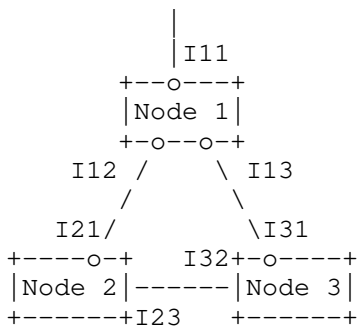


Figure 1

5. Flooding Topology

For a given network topology, a flooding topology is a sub-graph or sub-network of the given network topology that has the same reachability to every node as the given network topology. Thus all the nodes in the given network topology MUST be in the flooding topology. All the nodes MUST be inter-connected directly or indirectly. As a result, LS flooding will in most cases occur only on the flooding topology, that includes all nodes but a subset of links. Note even though the flooding topology is a sub-graph of the original topology, any single LS MUST still be disseminated in the entire network.

5.1. Construct Flooding Topology

Many different flooding topologies can be constructed for a given network topology. A chain connecting all the nodes in the given network topology is a flooding topology. A circle connecting all the nodes is another flooding topology. A tree connecting all the nodes is a flooding topology. In addition, the tree plus the connections

between some leaves of the tree and branch nodes of the tree is a flooding topology.

The following parameters need to be considered for constructing a flooding topology:

- Number of links: The number of links on the flooding topology is a key factor for reducing the amount of LS flooding. In general, the smaller the number of links, the less the amount of LS flooding.

- Diameter: The shortest distance between the two most distant nodes on the flooding topology is a key factor for reducing the network convergence time. The smaller the diameter, the less the convergence time.

- Redundancy: The redundancy of the flooding topology means a tolerance to the failures of some links and nodes on the flooding topology. If the flooding topology is split by some failures, it is not tolerant to these failures. In general, the larger the number of links on the flooding topology is, the more tolerant the flooding topology to failures.

There are many different ways to construct a flooding topology for a given network topology. A few of them are listed below:

- Central Mode: One node in the network builds a flooding topology and floods the flooding topology to all the other nodes in the network (This seems not good. Flooding the flooding topology may increase the flooding. The amount of traffic for flooding the flooding topology should be minimized.);

- Distributed Mode: Each node in the network automatically calculates a flooding topology by using the same algorithm (No flooding for flooding topology);

- Static Mode: Links on the flooding topology are configured statically.

Note that the flooding topology constructed by a node is dynamic in nature, that means when the base topology (the entire topology graph) changes, the flooding topology (the sub-graph) MUST be re-computed/ re-constructed to ensure that any node that is reachable on the base topology MUST also be reachable on the flooding topology.

For reference purpose, some algorithms that allow nodes to automatically compute flooding topology are elaborated in Appendix A.

However, this document does not attempt to standardize how a flooding topology is established.

5.2. Backup for Flooding Topology Split

It is hard to construct a flooding topology that reduces the amount of LS flooding greatly and is tolerant to multiple failures. To get around this, we can compute and use backup paths for a critical link and node on the flooding topology. Using backup paths may also speed up convergence when the link and node fail.

When a critical link on the flooding topology fails, the flooding topology without the critical link (i.e., the remaining flooding topology) is split into two parts. A backup path for the critical link connects the two parts into one. Through the backup path and the remaining flooding topology, an LS can be flooded to every node in the network. The combination of the backup path and the flooding topology is tolerant to the failure of the critical link.

When a critical node on the flooding topology goes down, the flooding topology without the critical node and the links attached to the node (i.e., the remaining flooding topology) is split into two or more parts. One or more backup paths for the critical node connects the split parts into one. Through the backup paths and the remaining flooding topology, an LS can be flooded to every live node in the network. The combination of the backup paths and the flooding topology is tolerant to the failure of the critical node.

In addition to the backup paths for a critical link and node, backup paths for every non critical link and node on the flooding topology can be computed. When the failures of multiple links and nodes on the flooding topology happen, through the remaining flooding topology and the backup paths for these links and nodes, an LS can be flooded to every live node in the network. The combination of the backup paths and the flooding topology is tolerant to the failures of these links and nodes. If there are other failures that break the backup paths, an LS can be flooded to every live node by the traditional flooding procedure.

In a centralized mode, the leader computes the backup paths and floods them to all the other nodes. In a distributed mode, every node computes the backup paths.

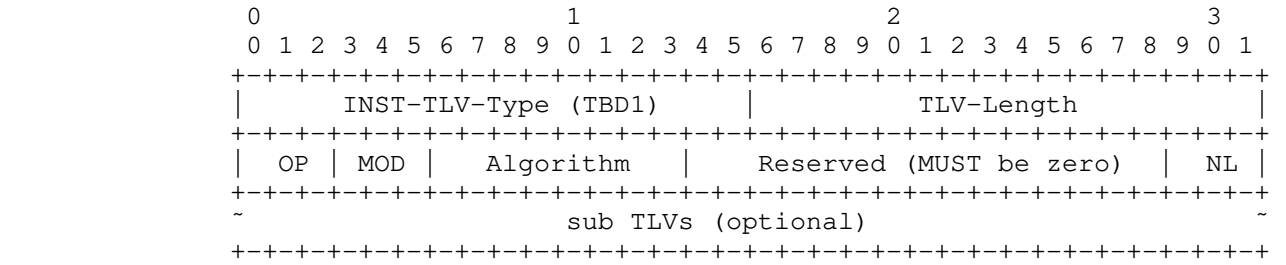
6. Extensions to OSPF

The extensions to OSPF comprises two parts: one part is for operations on flooding reduction, the other is specially for centralized mode flooding reduction.

6.1. Extensions for Operations

A new TLV is defined in OSPF RI LSA [RFC7770]. It contains instructions about flooding reduction, which is called Flooding Reduction Instruction TLV or Instruction TLV for short. This TLV is originated from only one node at any time.

The format of a Flooding Reduction Instruction TLV is as follows.



Flooding Reduction Instruction TLV

A OP field of three bits is defined in the TLV. It may have a value of the followings.

- 0x001 (R): Perform flooding Reduction, which instructs the nodes in a network to perform flooding reduction.

- 0x010 (N): Roll back to Normal flooding, which instructs the nodes in a network to roll back to perform normal flooding.

When any of the other values is received, it is ignored.

A MOD field of three bits is defined in the TLV and may have a value of the followings.

- 0x001 (C): Central Mode, which instructs 1) the nodes in a network to select leaders (primary/designated leader, secondary/backup leader, and so on); 2) the leaders in a network to compute a flooding topology and the primary leader to flood the flooding topology to all the other nodes in the network; 3) every node in the network to receive and use the flooding topology originated by the primary leader.

- 0x010 (D): Distributed Mode, which instructs every node in a network to compute and use its own flooding topology.

- 0x011 (S): Static Mode, which instructs every node in a network to use the flooding topology statically configured on the node.

When any of the other values is received, it is ignored.

An Algorithm field of eight bits is defined in the TLV to instruct the leader node in central mode or every node in distributed mode to use the algorithm indicated in this field for computing a flooding topology.

A NL field of three bits is defined in the TLV, which indicates the number of leaders to be selected when Central Mode is used. NL set to 2 means two leaders (a designated/primary leader and a backup/ secondary leader) to be selected for an area, and NL set to 3 means three leaders to be selected. When Central Mode is not used, The NL field is not valid.

Some optional sub TLVs may be defined in the future, but none is defined now.

6.2. Extensions for Centralized Mode

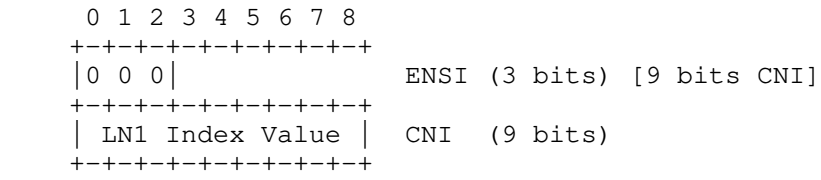
6.2.1. Message for Flooding Topology

A flooding topology can be represented by the links in the flooding topology. For the links between a local node and a number of its adjacent (or remote) nodes, we can encode the local node in a way, and encode its adjacent nodes in the same way or another way. After all the links in the flooding topology are encoded, the encoded links can be flooded to every node in the network. After receiving the encoded links, every node decodes the links and creates and/or updates the flooding topology.

For every node in an area, we may use an index to represent it. Every node in an area may order the nodes in a rule, which generates the same sequence of the nodes on every node in the area. The sequence of nodes have the index 0, 1, 2, and so on respectively. For example, every node orders the nodes by their router IDs in ascending order.

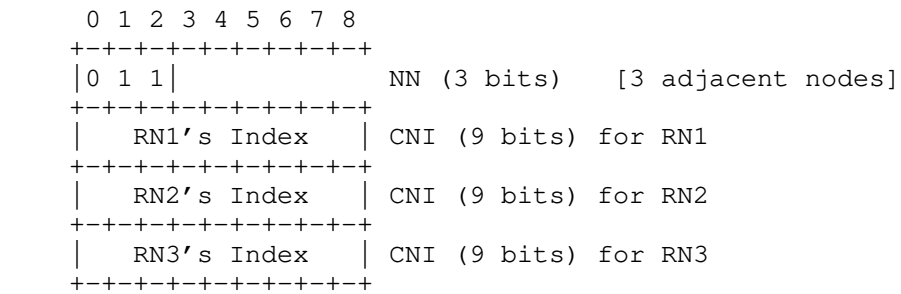
6.2.1.1. Links Encoding

A local node can be encoded in two parts: encoded node index size indication (ENSI) and compact node index (CNI). ENSI value plus a number (e.g., 9) gives the size of compact node index. For example, ENSI = 0 indicates that the size of CNIs is 9 bits. In the figure below, Local node LN1 is encoded as ENSI=0 using 3 bits and CNI=LN1's Index using 9 bits. LN1 is encoded in 12 bits in total.



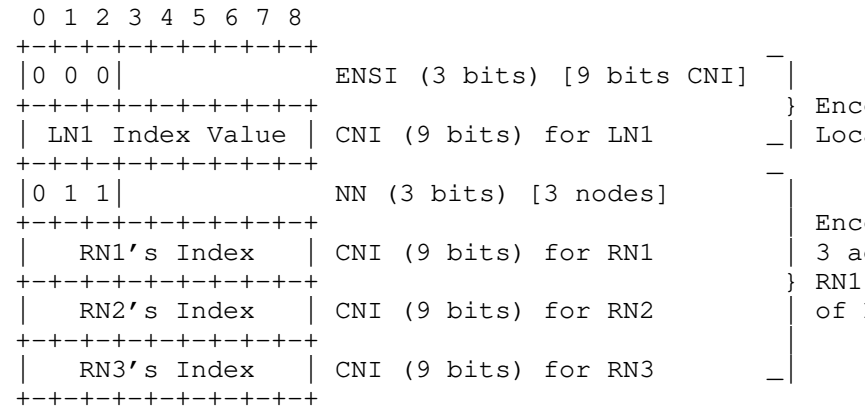
An Example of Local Node Encoding

The adjacent nodes can be encoded in two parts: Number of Nodes (NN) and compact node indexes (CNIs). The size of CNIs is the same as the local node. For example, three adjacent nodes RN1, RN2 and RN3 are encoded below in 30 bits (i.e., 3.75 bytes).



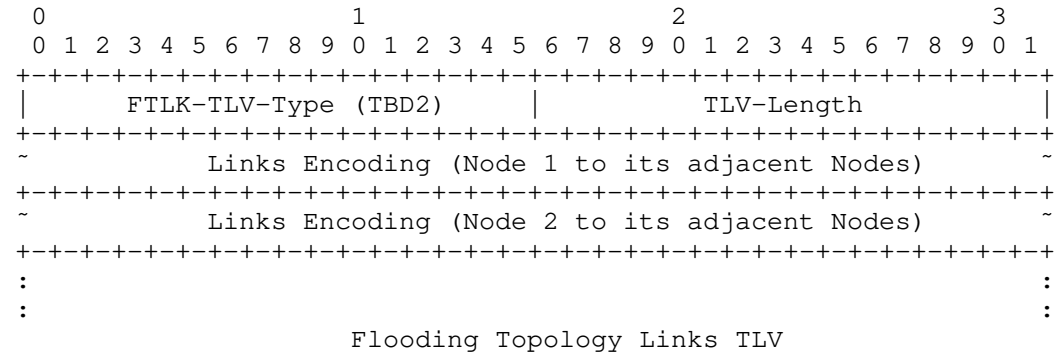
An Example of Adjacent Nodes Encoding

The links between a local node and a number of its adjacent (or remote) nodes can be encoded as the local node followed by the adjacent nodes. For example, three links between local node LN1 and its three adjacent nodes RN1, RN2 and RN3 are encoded below in 42 bits (i.e., 5.25 bytes).



An Example of Links Encoding

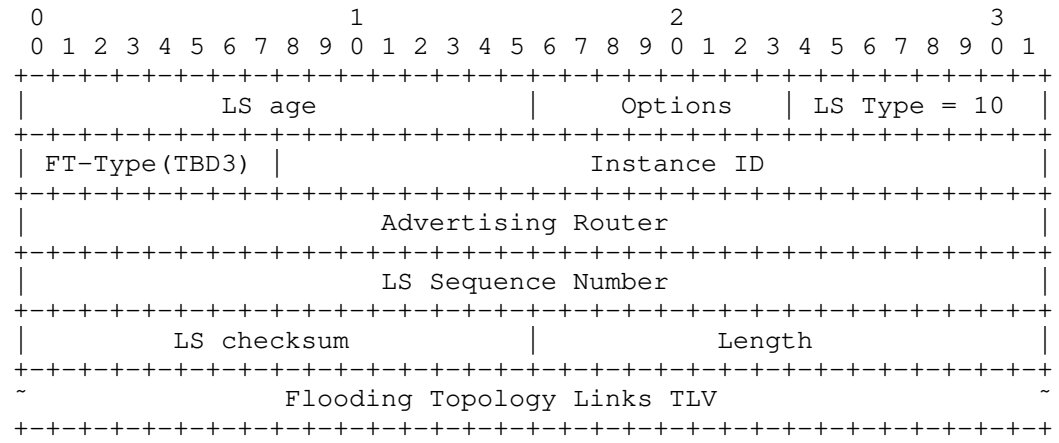
For a flooding topology computed by a leader of an area, it may be represented by all the links on the flooding topology. A Type-Length-Value (TLV) of the following format for the links encodings can be included in an LSA to represent the flooding topology (FT) and flood the FT to every node in the area.



Flooding Topology Links TLV

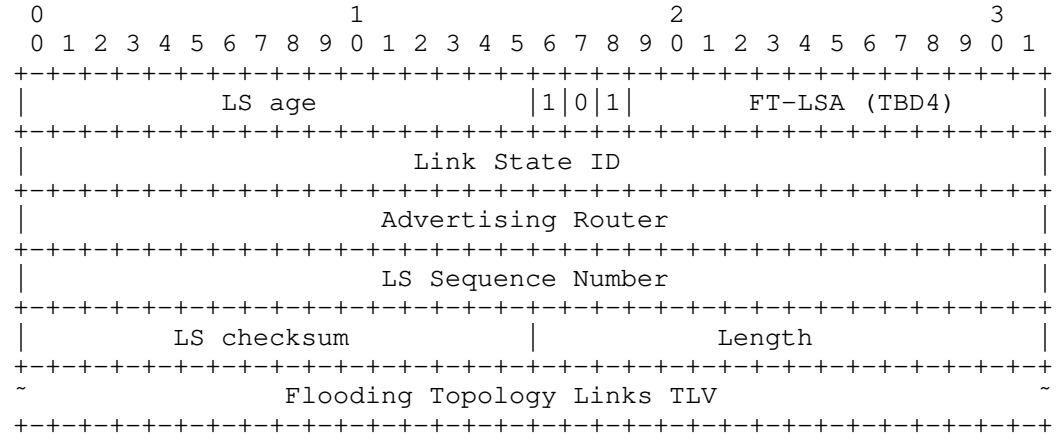
Note that a link between a local node LN and its adjacent node RN can be encoded once and as a bi-directional link. That is that if it is encoded in a Links Encoding from LN to RN, then the link from RN to LN is implied or assumed.

For OSPFv2, an Opaque LSA of a new opaque type (TBD3) containing a Flooding Topology Links TLV is used to flood the flooding topology from the leader of an area to all the other nodes in the area.



OSPFv2: Flooding Topology Opaque LSA

For OSPFv3, an area scope LSA of a new LSA function code (TBD4) containing a Flooding Topology Links TLV is used to flood the flooding topology from the leader of an area to all the other nodes in the area.



OSPFv3: Flooding Topology LSA

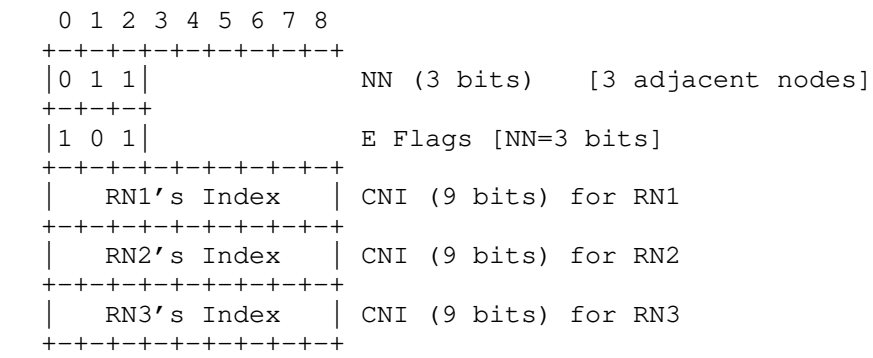
The U-bit is set to 1, and the scope is set to 01 for area-scoping.

6.2.1.2. Block Encoding

Block encoding uses a single structure to encode a block (or part) of topology, which can be a block of links in a flooding topology. It can also be all the links in the flooding topology. It starts with a local node LN and its adjacent (or remote) nodes RNi (i = 1, 2, ..., n), and can be considered as an extension to the links encoding.

The encoding of links between a local node and its adjacent nodes described in Section 6.2.1.1 is extended to include the links attached to the adjacent nodes.

The encoding for the adjacent nodes is extended to include Extending Flags (E Flags for short) between the NN (Number of Nodes) field and the CNIs (Compact Node Indexes) for the adjacent nodes. The length of the E Flags field is NN bits. The following is an example encoding of the adjacent nodes with E Flags of 3 bits, which is the value of the NN (the number of adjacent nodes).



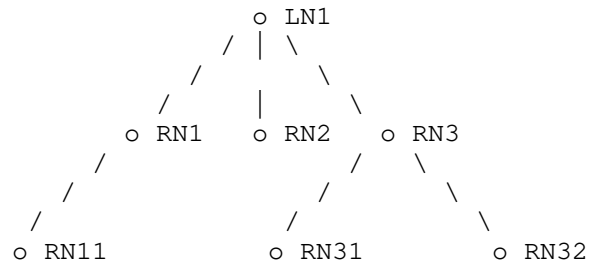
An Example of Adjacent Nodes with E Flags Encoding

There is a bit flag (called E flag) in the E Flags field for each adjacent node. The first bit (i.e., the most significant bit) in the E Flags field is for the first adjacent node (e.g., RN1), the second bit is for the second adjacent node (e.g., RN2), and so on. The E flag for an adjacent node RNi set to one indicates that the links attached to the adjacent node RNi are included below. The E flag for an adjacent node RNi set to zero means that no links attached to the adjacent node RNi are included below.

The links attached to the adjacent node RNi are represented by the RNi as a local node and the adjacent nodes of RNi. The encoding for the adjacent nodes of RNi is the same as that for the adjacent nodes

of a local node. It consists of an NN field of 3 bits, E Flags field of NN bits, and CNIs for the adjacent nodes of RNi.

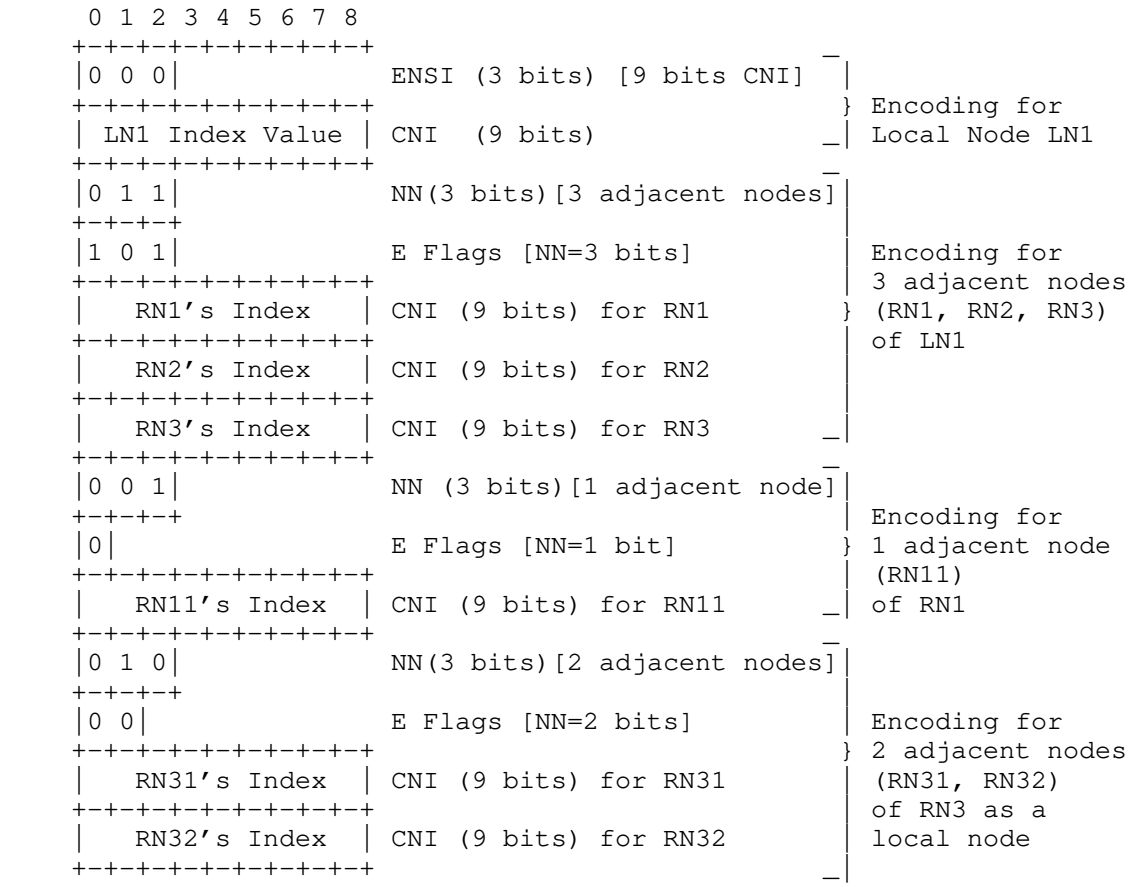
The following is an example of a block encoding for a block (or part) of flooding topology below.



An Example Block of Flooding Topology

It represents 6 links: 3 links between local node LN1 and its 3 adjacent nodes RN1, RN2 and RN3; 1 link between RN1 as a local node and its 1 adjacent node RN11; and 2 links between RN3 as a local node and its 2 adjacent nodes RN31 and RN32.

It starts with the encoding of the links between local node LN1 and 3 adjacent nodes RN1, RN2 and RN3 of the local node LN1. The encoding for the local node LN1 is the same as that for a local node described in Section 6.2.1.1. The encoding for 3 adjacent nodes RN1, RN2 and RN3 of local node LN1 comprises an NN field of 3 bits with value of 3, E Flags field of NN = 3 bits, and the indexes of adjacent nodes RN1, RN2 and RN3.



An Example of Block Encoding

The first E flag in the encoding for adjacent nodes RN1, RN2 and RN3 is set to one, which indicates that the links between the first adjacent node RN1 as a local node and its adjacent nodes are included below. In this example, 1 link between RN1 and its adjacent node RN11 is represented by the encoding for the adjacent node RN11 of RN1 as a local node. The encoding for 1 adjacent node RN11 consists of an NN field of 3 bits with value of 1, E Flags field of NN = 1 bits, and the index of adjacent node RN11. The size of the index of RN11 is the same as that of local node LN1 indicated by the ENSI in the encoding for local node LN1.

The second E flag in the encoding for adjacent nodes RN1, RN2 and RN3 is set to zero, which indicates that no links between the second

adjacent node RN2 as a local node and its adjacent nodes are included below.

The third E flag in the encoding for adjacent nodes RN1, RN2 and RN3 is set to one, which indicates that the links between the third adjacent node RN3 as a local node and its adjacent nodes are included below. In this example, 2 links between RN3 and its 2 adjacent nodes RN31 and RN32 are represented by the encoding for the adjacent nodes RN31 and RN32 of RN3 as a local node. The encoding for 2 adjacent nodes RN31 and RN32 consists of an NN field of 3 bits with value of 2, E Flags field of NN = 2 bits, and the indexes of adjacent nodes RN31 and RN32. The size of the index of RN31 and RN32 is the same as that of local node LN1 indicated by the ENSI in the encoding for local node LN1.

The block encoding may be used in the place of the links encoding in Section 6.2.1.1 for more efficiency. That is that it may be used in a Flooding Topology Links TLV. Alternatively, a new TLV, which is similar to the Flooding Topology Links TLV, may be defined to contain a number of block encodings.

6.2.2. Encodings for Backup Paths

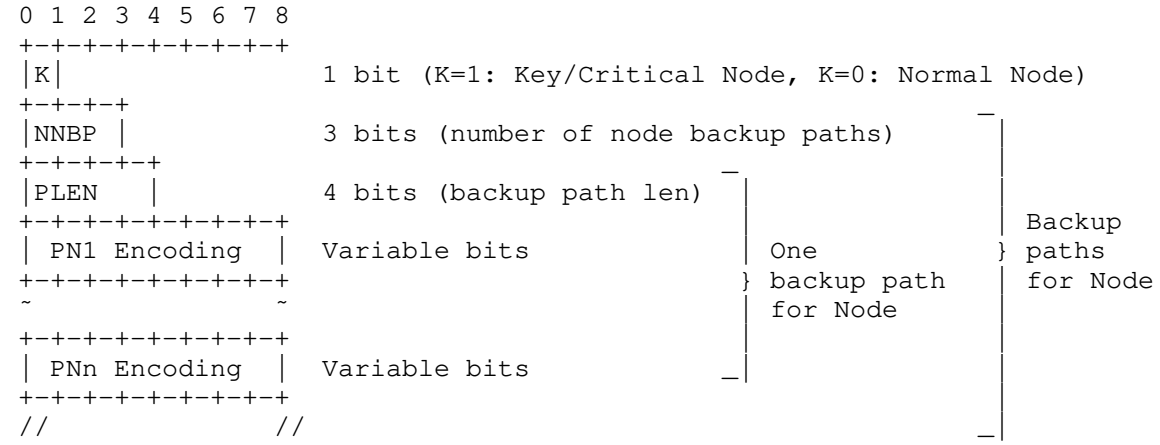
When the leader of an area computes a flooding topology, it may compute a backup path or multiple backup paths for a critical link on the flooding topology. When the critical link fails, a link state can be distributed to every node in the area through one backup path and other links on the flooding topology. In addition, it may compute a backup path or multiple backup paths for a node. When the node fails, a link state can be distributed to the other nodes in the area through the backup paths and the links on the flooding topology.

This section describes two encodings for backup paths: separated encoding and integrated one. In the former, backup paths are encoded in a new message, where the message for the flooding topology described in the previous section is required; In the latter, backup paths are integrated into the flooding topology links encoding, where one message contains the flooding topology and the backup paths.

6.2.2.1. Message for Backup Paths

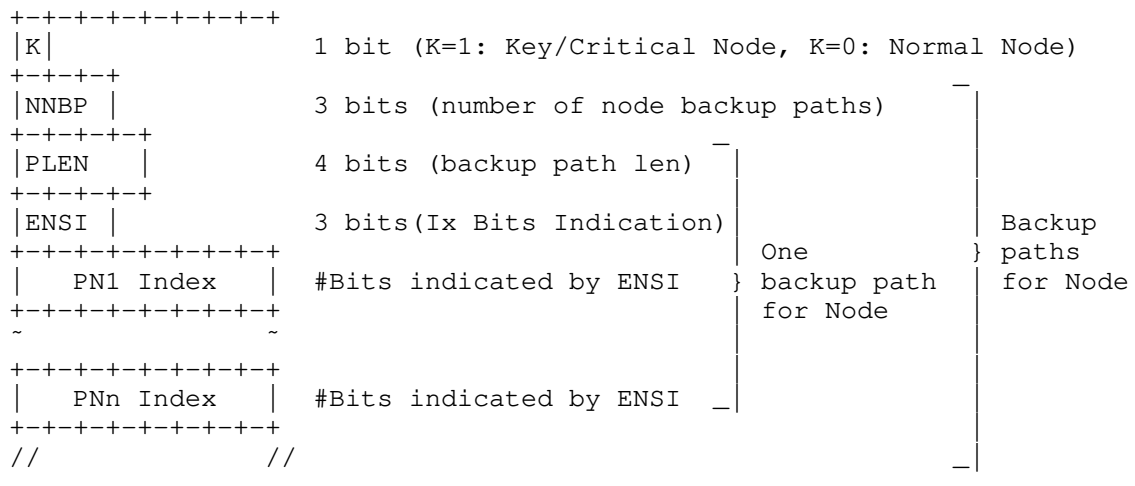
Backup paths for a node (such as Node1) may be represented by the node index encoding and node backup paths encoding. The former is similar to local node index encoding. The latter has the following format. It comprises a K flag (Key/Critical node flag) of 1 bit, a 3 bits NNBP field (number of node backup paths), and each of the backup paths encoding, which consists of the path length PLEN of 4 bits indicating the length of the path (i.e., the number of nodes), and

the encoding of the sequence of nodes along the path such as encodings for nodes PN1, ..., PNn. The encoding of every node may use the encoding of a local node, which comprises encoded node index size indication (ENSI) and compact node index (CNI).



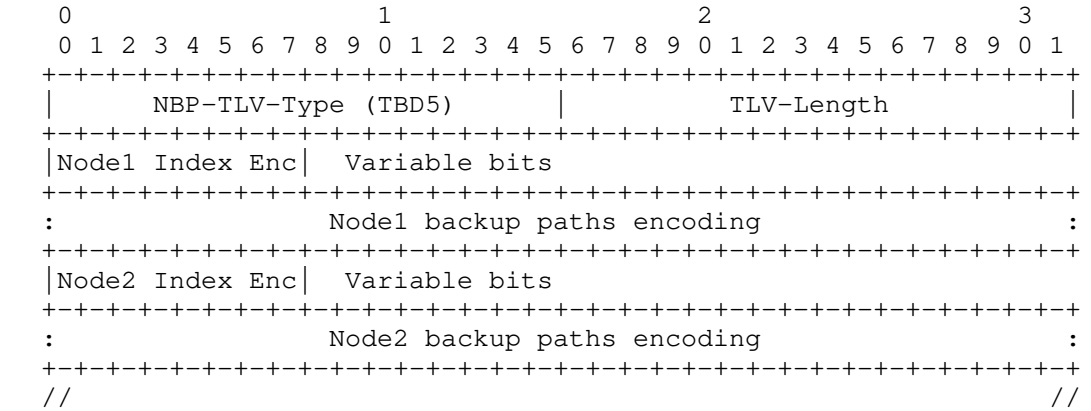
An Example of Node Backup Paths Encoding

Another encoding of the sequence of nodes along the path uses one encoded node index size indication (ENSI) for all the nodes in the path. Thus we have the following Node Backup Paths Encoding.



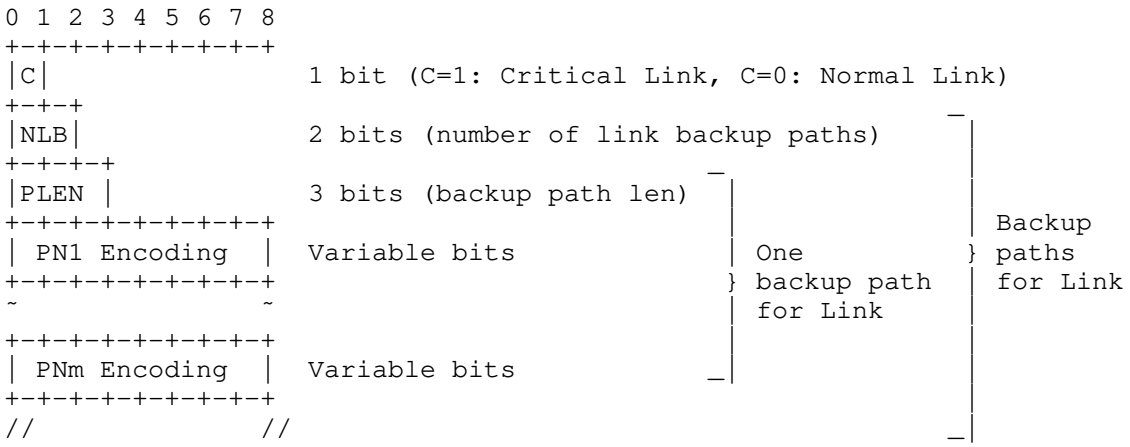
Another Example of Node Backup Paths Encoding

A new TLV called Node Backup Paths TLV is defined below. It may include multiple nodes and their backup paths. Each node is represented by its index encoding, which is followed by its node backup paths encoding.



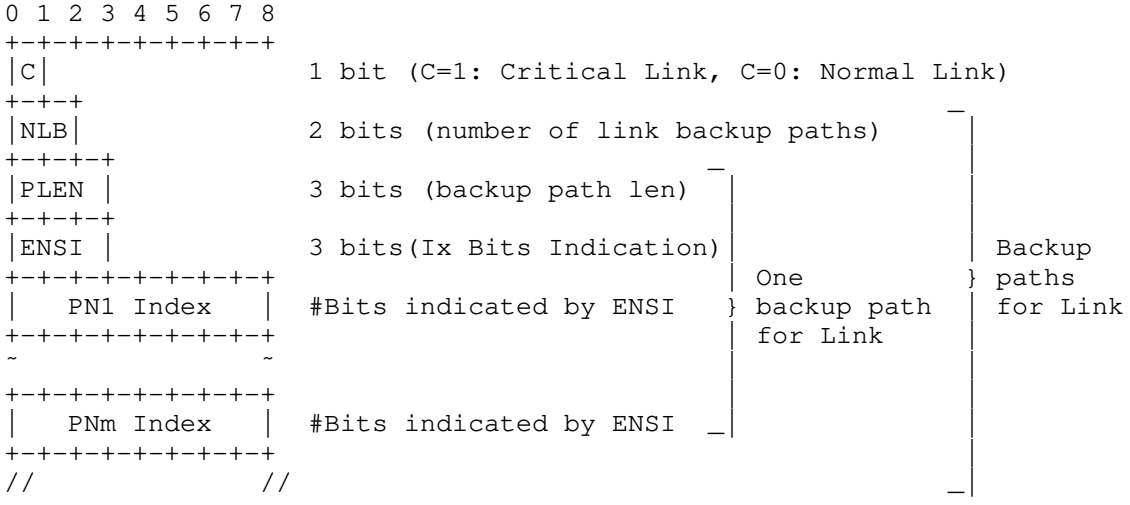
Node Backup Paths TLV

The encoding for backup paths for a link (such as Link1) on the flooding topology consists of the link encoding such as Link1 Index Encoding and the link backup paths encoding. The former is similar to local node encoding. It contains encoded link index size indication (ELSI) and compact link index (CLI). The latter has the following format. It comprises a C flag (Critical link flag) of 1 bit, a 2 bits NLB field (number of link backup paths), and each of the backup paths encoding, which consists of the path length PLEN of 3 bits indicating the length of the path (i.e., the number of nodes), and the encoding of the sequence of nodes along the path such as encodings for nodes PN1, ..., PNm. Note that two ends of a link (i.e., the local node and the adjacent/remote node of the link) are not needed in the path. The encoding of every node may use the encoding of a local node, which comprises encoded node index size indication (ENSI) and compact node index (CNI).



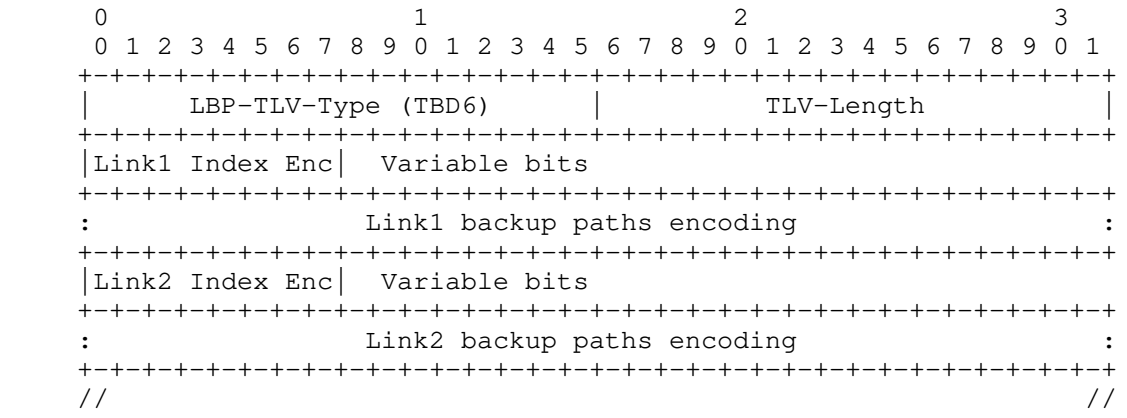
An Example of Link Backup Paths Encoding

Another encoding of the sequence of nodes along the path uses one encoded node index size indication (ENSI) for all the nodes in the path. Thus we have the following Link Backup Paths Encoding.



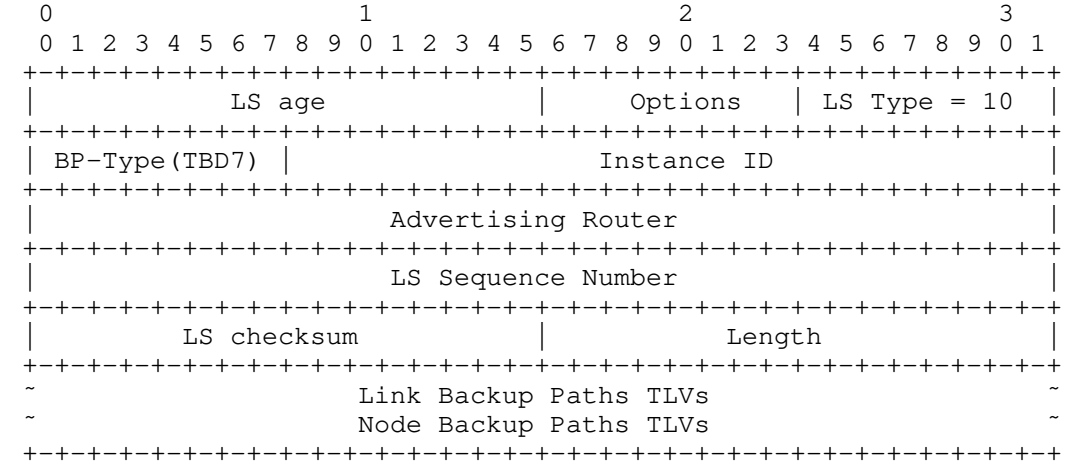
Another Example of Link Backup Paths Encoding

A new TLV called Link Backup Paths TLV is defined below. It may include multiple links and their backup paths. Each link is represented by its index encoding, which is followed by its link backup paths encoding.



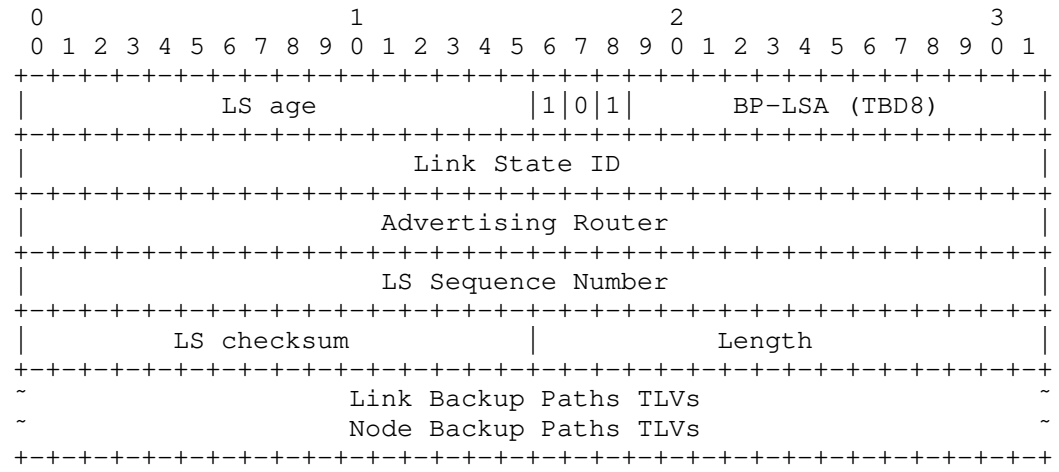
Link Backup Paths TLV

For OSPFv2, an Opaque LSA of a new opaque type (TBD7), containing node backup paths TLVs and link backup paths TLVs, is used to flood the backup paths from the leader of an area to all the other nodes in the area.



OSPFv2: Backup Paths Opaque LSA

For OSPFv3, an area scope LSA of a new LSA function code (TBD8), containing node backup paths TLVs and link backup paths TLVs, is used to flood the backup paths from the leader of an area to all the other nodes in the area.

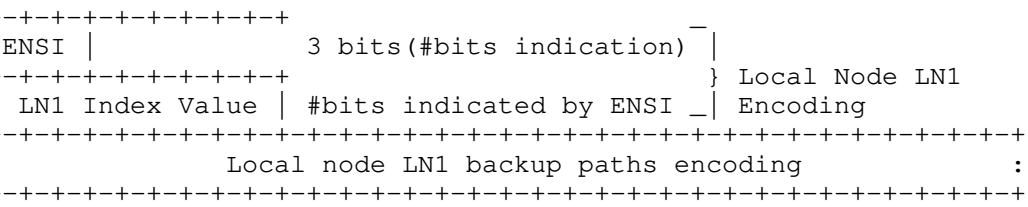


OSPFv3: Backup Paths LSA

The U-bit is set to 1, and the scope is set to 01 for area-scoping.

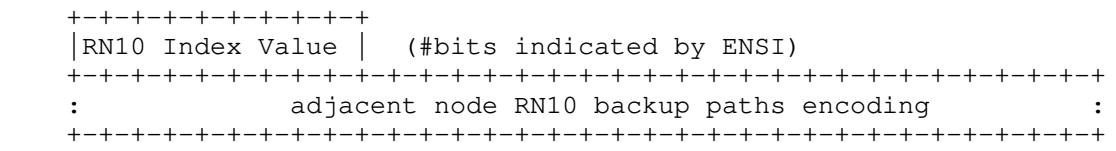
6.2.2.2. Backup Paths in Links TLV

A local node and its backup paths can be encoded in the following format. It is the local node (such as local node LN1) encoding followed by the local node backup paths encoding, which is the same as the node backup paths encoding described in Section 6.2.2.1.



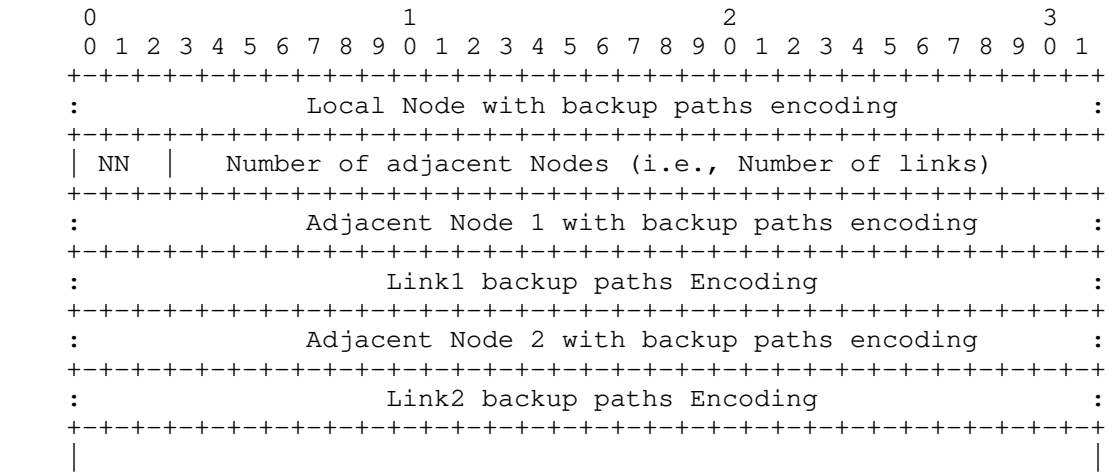
Local Node with Backup Paths Encoding

A adjacent node and its backup paths can be encoded in the following format. It is the adjacent node (such as adjacent node RN10) index value followed by the adjacent node backup paths encoding, which is the same as the node backup paths encoding described in Section 6.2.2.1.



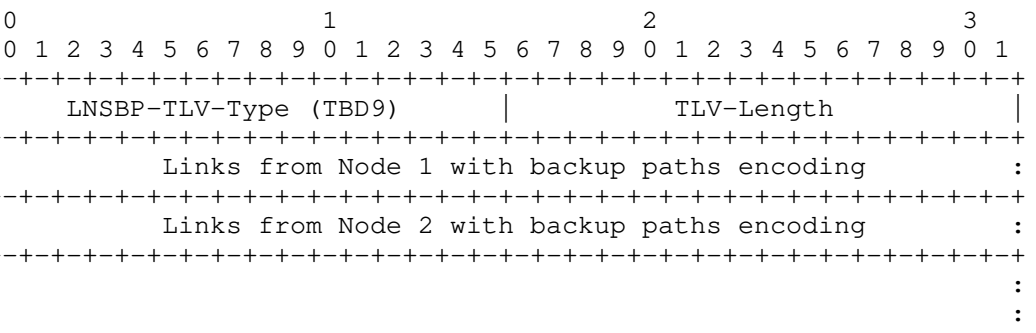
Adjacent Node with Backup Paths Encoding

The links between a local node and a number of its adjacent nodes, the backup paths for each of the nodes, and the backup paths for each of the links can be encoded in the following format. It is called Links from Node with Backup Paths Encoding.



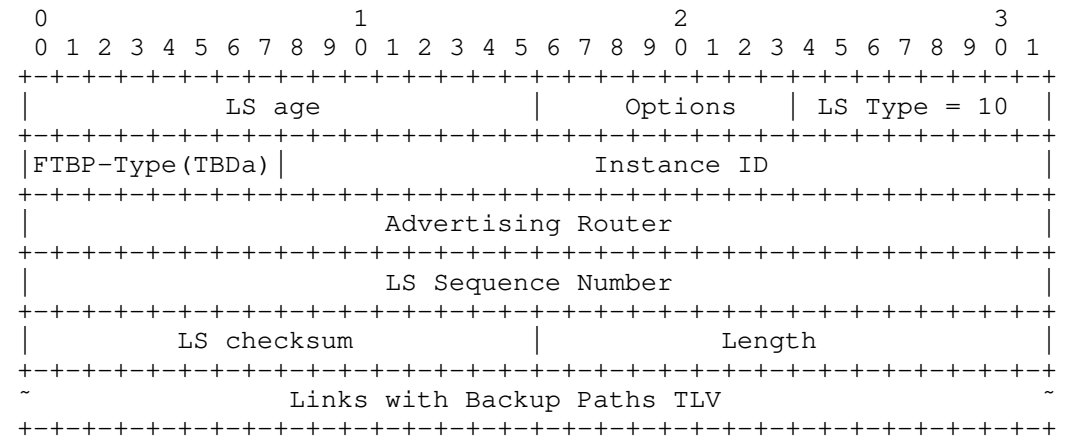
Links from Node with Backup Paths Encoding

A new TLV called Links with Backup Paths TLV is defined below. It includes a number of Links from Node with Backup Paths Encodings described above. This TLV contains both the flooding topology and the backup paths for the links and nodes on the flooding topology.



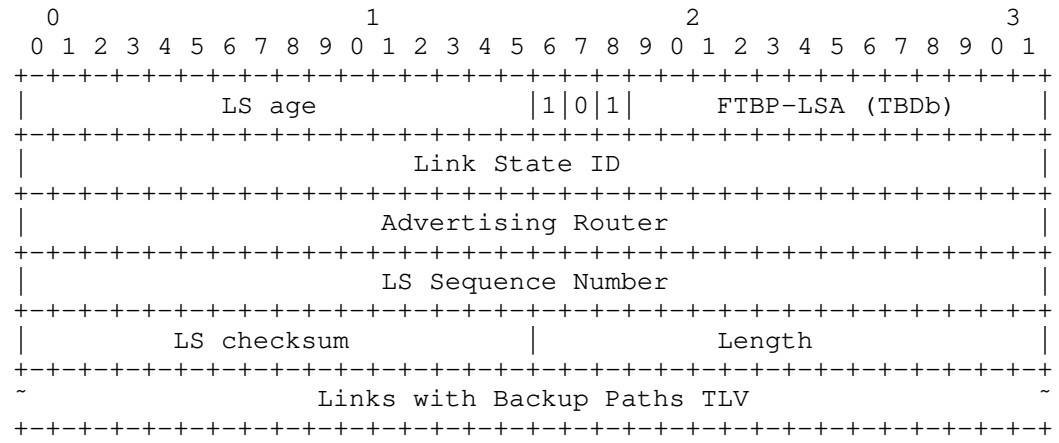
Links with Backup Paths TLV

For OSPFv2, an Opaque LSA of a new opaque type (TBDa), called Flooding Topology with Backup Paths (FTBP) Opaque LSA, containing a Links with Backup Paths TLV, is used to flood the flooding topology with backup paths from the leader of an area to all the other nodes in the area.



OSPFv2: Flooding Topology with Backup Paths (FTBP) Opaque LSA

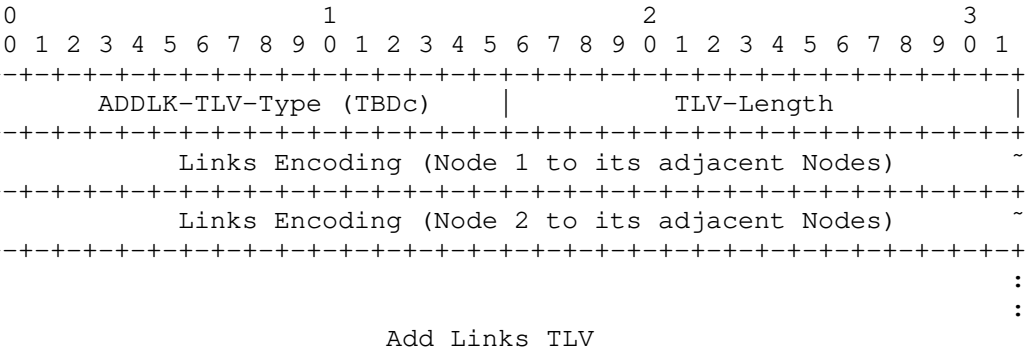
For OSPFv3, an area scope LSA of a new LSA function code (TBDb), containing a Links with Backup Paths TLV, is used to flood the flooding topology with backup paths from the leader of an area to all the other nodes in the area.



OSPFv3: Flooding Topology with Backup Paths (FTBP) LSA

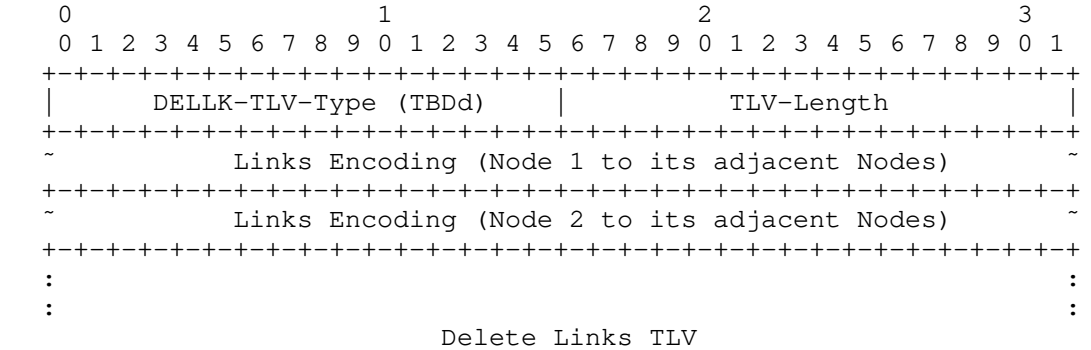
6.2.3. Message for Incremental Changes

For adding some links to the flooding topology, we define a new TLV called Add Links TLVs of the following format. When some new links are added to the flooding topology, the leader may not flood the whole flooding topology with the new links to all the other nodes. It may just flood these new links. After receiving these new links, each of the other nodes adds these new links into the existing flooding topology. When the leader floods the whole flooding topology with the new links to all the other nodes, it removes the LSA for the new links. When removing the LSA for these new links, each of the other nodes does not update the flooding topology (i.e., does not remove these links from the flooding topology).



Add Links TLV

For deleting some links from the flooding topology, we define a new TLV called Delete Links TLVs of the following format. When some old links are removed from the flooding topology, the leader may not flood the whole flooding topology without the old links to all the other nodes. It may just flood these old links. After receiving these old links, each of the other nodes deletes these old links from the existing flooding topology. When the leader floods the whole flooding topology without the old links to all the other nodes, it removes the LSA for the old links. When removing the LSA for these old links, each of the other nodes does not update the flooding topology (i.e., does not add these links into the flooding topology).



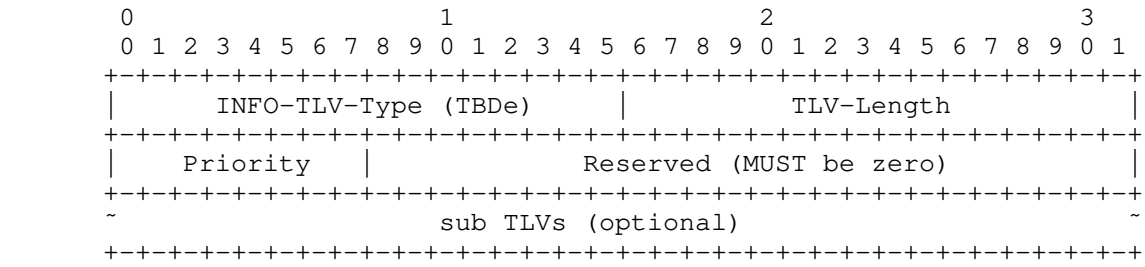
Delete Links TLV

The Add Links TLVs and Delete Links TLVs should be in a separate LSA instance. The LSA can be a Flooding Topology LSA defined above. Alternatively, we may define a new LSA for these TLVs.

6.2.4. Leaders Selection

The leader or Designated Router (DR) selection for a broadcast link is about selecting two leaders: a DR and Backup DR. This is generalized to select two or more leaders for an area: the primary/ first leader (or leader for short), the secondary leader, the third leader and so on.

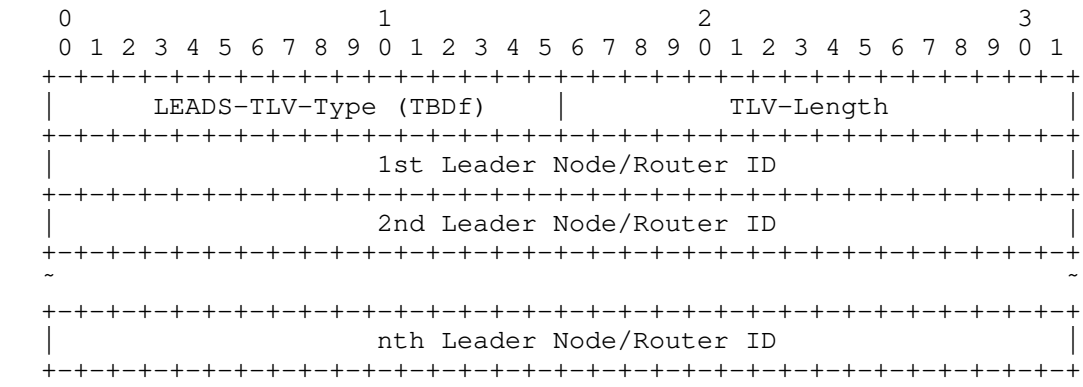
A new TLV is defined to include the information on flooding reduction of a node, which is called Flooding Reduction Information TLV or Information TLV for short. This TLV is generated by every node that supports flooding reduction in general. Every node originates a RI LSA with a Flooding Reduction Information TLV containing its priority to become a leader. The format of the TLV is as follows.



Flooding Reduction Information TLV

A Priority field of eight bits is defined in the TLV to indicate the priority of the node originating the TLV to become the leader node in central mode.

A sub-TLV called leaders sub-TLV is defined. It has the following format.



Leaders sub-TLV

When a node selects itself as a leader, it originates a RI LSA containing the leader in a leaders sub-TLV.

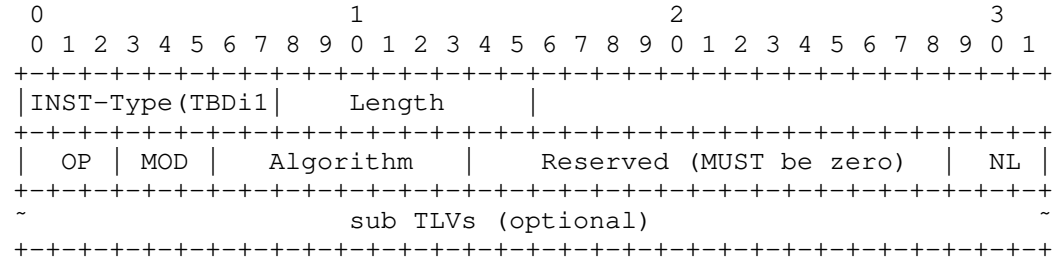
After the first leader node is down, the other leaders will be promoted. The secondary leader becomes the first leader, the third leader becomes the secondary leader, and so on. When a node selects itself as the n-th leader, it originates a RI LSA with a Leaders sub-TLV containing n leaders.

7. Extensions to IS-IS

The extensions to IS-IS is similar to OSPF.

7.1. Extensions for Operations

A new TLV for operations is defined in IS-IS LSP. It has the following format and contains the same contents as the Flooding Reduction Instruction TLV defined in OSPF RI LSA.

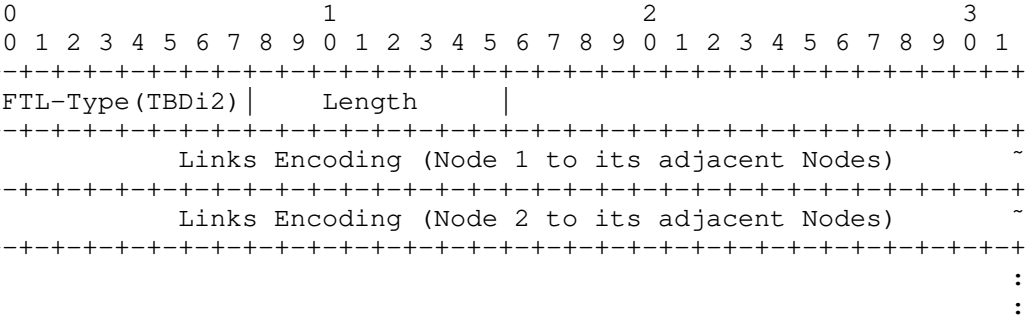


IS-IS Flooding Reduction Instruction TLV

7.2. Extensions for Centralized Mode

7.2.1. TLV for Flooding Topology

A new TLV for the encodings of the links in the flooding topology is defined. It has the following format and contains the same contents as the Flooding Topology Links TLV defined in OSPF Flooding Topology Opaque LSA.

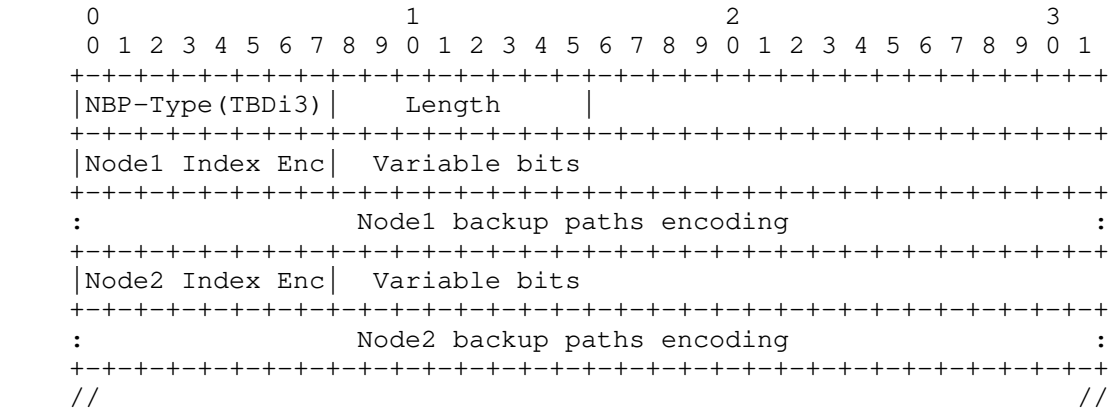


IS-IS Flooding Topology Links TLV

7.2.2. Encodings for Backup Paths

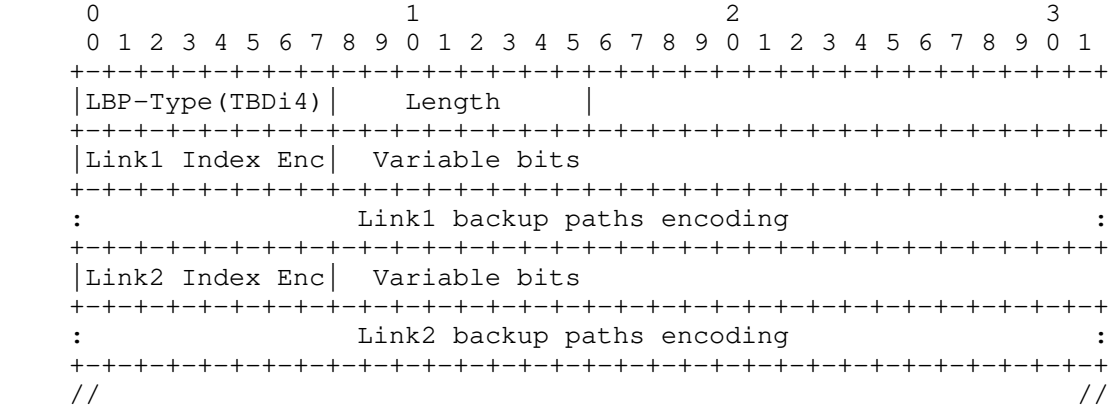
7.2.2.1. TLVs for Backup Paths

For flooding backup paths separately, we define two TLVs: IS-IS Node Backup Paths TLV and IS-IS Link Backup Path TLV. The former has the following format and contains the same contents as Node Backup Paths TLV in OSPF.



IS-IS Node Backup Paths TLV

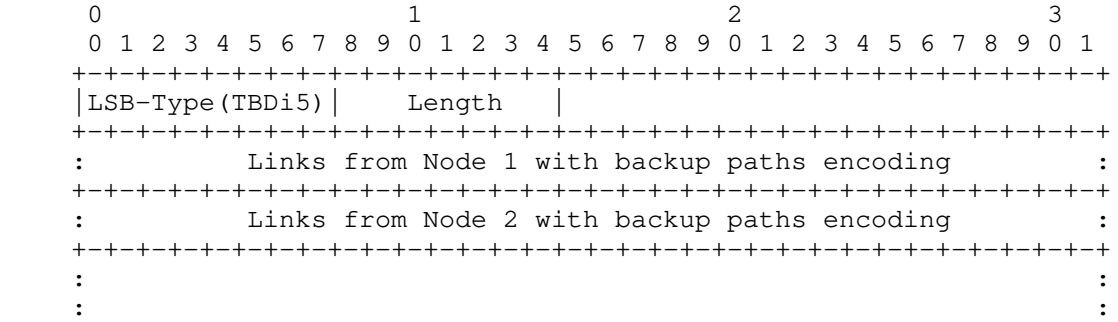
The latter has the following format and contains the same contents as Link Backup Paths TLV in OSPF.



IS-IS Link Backup Paths TLV

7.2.2.2. Backup Paths in Links TLV

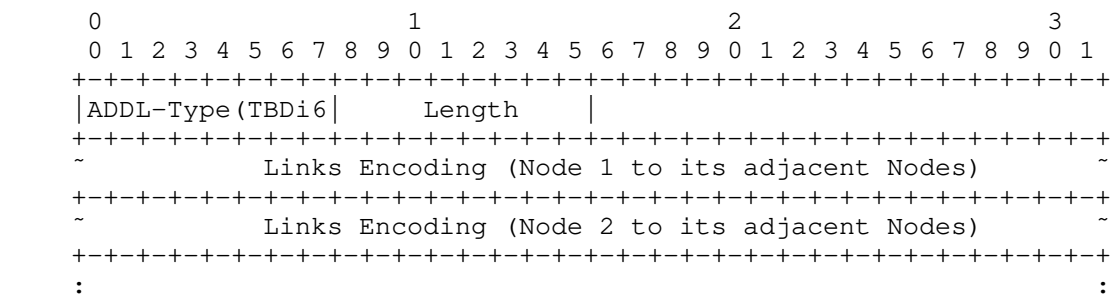
A new TLV is defined to integrate the backup paths with the links on the flooding topology. It has the following format and contains the same contents as the Links with Backup Paths TLV in OSPF.



IS-IS Links with Backup Paths TLV

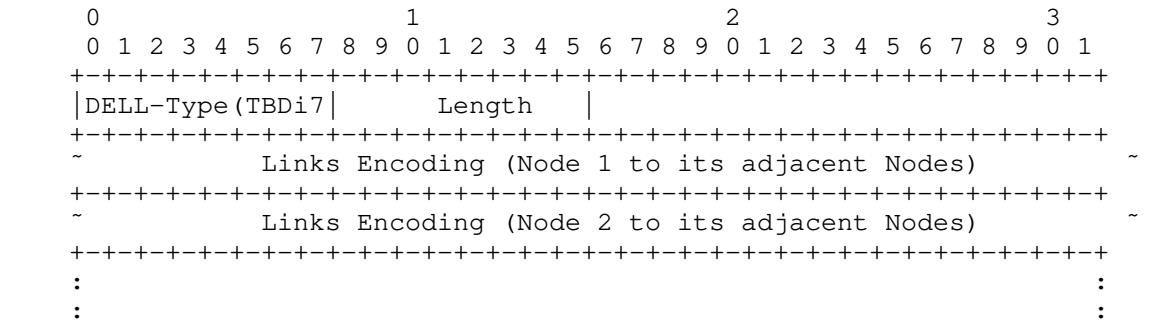
7.2.3. TLVs for Incremental Changes

Similar to Add Links TLV in OSPF, a new TLV called IS-IS Add Links TLV is defined. It has the following format and contains the same contents as Add Links TLV in OSPF.



IS-IS Add Links TLV

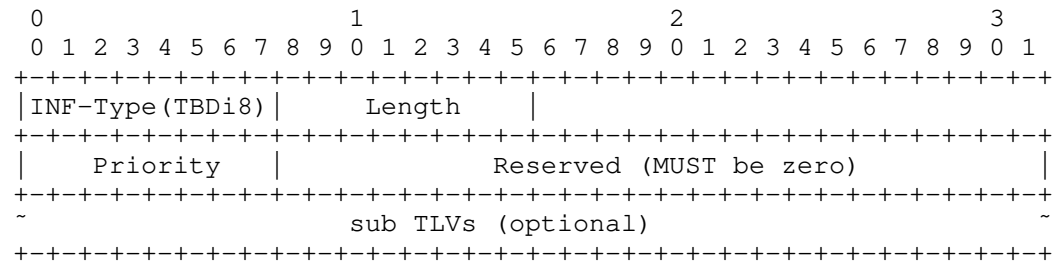
Similar to Delete Links TLV in OSPF, a new TLV called IS-IS Delete Links TLV is defined. It has the following format and contains the same contents as Delete Links TLV in OSPF.



IS-IS Delete Links TLV

7.2.4. Leaders Selection

Similar to Flooding Reduction Information TLV in OSPF, a new TLV called IS-IS Flooding Reduction Information TLV is defined. It has the following format and contains the same contents as Flooding Reduction Information TLV in OSPF.



IS-IS Flooding Reduction Information TLV

8. Flooding Behavior

This section describes the revised flooding behavior for a node having at least one link on the flooding topology. The revised flooding procedure MUST flood an LS to every node in the network in any case, as the standard flooding procedure does.

8.1. Nodes Perform Flooding Reduction without Failure

8.1.1. Receiving an LS

When a node receives a newer LS that is not originated by itself from one of its interfaces, it floods the LS only to all the other interfaces that are on the flooding topology.

When the LS is received from an interface on the flooding topology, it is flooded only to all the other interfaces that are on the flooding topology. When the LS is received on an interface that is not on the flooding topology, it is also flooded only to all the other interfaces that are on the flooding topology.

In any case, the LS must not be transmitted back to the receiving interface.

Note before forwarding a received LS, the node would do the normal processing as usual.

8.1.2. Originating an LS

When a node originates an LS, it floods the LS to its interfaces on the flooding topology if the LS is a refresh LS (i.e., there is no significant change in the LS comparing to the previous LS); otherwise (i.e., there are significant changes in the LS), it floods the LS to all its interfaces. Choosing flooding the LS with significant changes to all the interfaces instead of limiting to the interfaces on the flooding topology would speed up the distribution of the significant link state changes.

8.1.3. Establishing Adjacencies

Adjacencies being established can be classified into two categories: adjacencies to new nodes and adjacencies to existing nodes.

8.1.3.1. Adjacency to New Node

An adjacency to a new node is an adjacency between a node (say node A) on the flooding topology and the new node (say node Y) which is not on the flooding topology. There is not any adjacency between node Y and a node in the network area.

When new node Y is up and connected to node A, node A assumes that node Y and the link between node Y and node A are on the flooding topology until a new flooding topology is computed and built. Node A may determine whether node Y is a new node through checking if node Y is reachable or on the flooding topology.

The procedure for establishing the adjacency between node A and node Y is the existing normal procedure unchanged. After the status of the adjacency reaches to Exchange or Full, node A sends node Y every new or updated LS that node A receives or originates.

8.1.3.2. Adjacency to Existing Node

An adjacency to an existing node is an adjacency between a node (say node A) on the flooding topology and the existing node (say node X) which exists on the flooding topology. There are some adjacencies between node X and some nodes in the network area.

When existing node X is connected to node A after a link between node X and node A is up, node A assumes that the link connecting node A and node X is not on the flooding topology until a new flooding topology is computed and built. Node A may determine whether node X is an existing node through checking if node X is reachable or on the flooding topology.

The procedure for establishing the adjacency between node A and node X is the existing normal procedure unchanged. Node A does not send node X any new or updated LS that node A receives or originates even after the status of the adjacency reaches to Exchange or Full.

8.2. An Exception Case

During an LS flooding, one or multiple link and node failures may happen. Some failures do not split the flooding topology, thus do not affect the flooding behavior. For example, multiple failures of the links not on the flooding topology do not split the flooding topology and do not affect the flooding behavior. The sections below focus on the failures that may split the flooding topology.

8.2.1. A Critical Failure

For a link failure, if the link is a critical link on the flooding topology, then the LS is flooded through a backup path for the link and the remaining flooding topology until a new flooding topology is computed and built; otherwise, the flooding behavior in Section 8.1 follows.

Similarly, for a node failure, if the node is a critical node on the flooding topology, then the LS is flooded through backup paths for the node and the remaining flooding topology until a new flooding topology is computed and built; otherwise, the flooding behavior in Section 8.1 follows.

8.2.2. Multiple Failures

For multiple link failures, if the number of the failed links on the flooding topology is greater than or equal to two, then the LS is flooded through a backup path for each of the failed links on the flooding topology and the remaining flooding topology until a new

flooding topology is computed and built; otherwise, the flooding behavior in Section 8.1 follows.

If all the backup paths for some of the failed links are broken by some failures, the LS is flooded to all interfaces (except where it is received from) until a new flooding topology is computed and built.

For multiple node failures, the LS is flooded through the backup paths for each of the failed nodes and the remaining flooding topology until a new flooding topology is computed and built; otherwise, the flooding behavior in Section 8.1 follows.

If the backup paths for some of the failed nodes are broken by some failures, the LS is flooded to all interfaces (except where it is received from) until a new flooding topology is computed and built.

Note that if it can be quickly determined that the flooding topology is not split by the failures, the flooding behavior in Section 8.1 may follow.

9. Security Considerations

This document does not introduce any security issue.

10. IANA Considerations

10.1. OSPFv2

Under Registry Name: OSPF Router Information (RI) TLVs [RFC7770], IANA is requested to assign two new TLV values for OSPF flooding reduction as follows:

TLV Value	TLV Name	reference
11	Instruction TLV	This document
12	Information TLV	This document

Under the registry name "Opaque Link-State Advertisements (LSA) Option Types" [RFC5250], IANA is requested to assign new Opaque Type registry values for FT LSA, BP LSA, FTBP LSA as follows:

Registry Value	Opaque Type	reference
10	FT LSA	This document
11	BP LSA	This document
12	FTBP LSA	This document

IANA is requested to create and maintain new registries:

o OSPFv2 FT LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv2 FT LSA TLV Name	Definition
0	Reserved	
1	FT Links TLV	see Section 6.2.1
2-32767	Unassigned	
32768-65535	Reserved	

o OSPFv2 BP LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv2 TBPLSA TLV Name	Definition
0	Reserved	
1	Node Backup Paths TLV	see Section 6.2.2
2	Link Backup Paths TLV	see Section 6.2.2
3-32767	Unassigned	
32768-65535	Reserved	

o OSPFv2 FTBP LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv2 FTBP LSA TLV Name	Definition
0	Reserved	
1	Links with Backup Paths TLV	see Section 6.2.2
2-32767	Unassigned	
32768-65535	Reserved	

10.2. OSPFv3

Under the registry name "OSPFv3 LSA Function Codes", IANA is requested to assign new registry values for FT LSA, BP LSA, FTBP LSA as follows:

Value	LSA Function Code Name	reference
16	FT LSA	This document
17	BP LSA	This document
18	FTBP LSA	This document

IANA is requested to create and maintain new registries:

- OSPFv3 FT LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv3 FT LSA TLV Name	Definition
0	Reserved	
1	FT Links TLV	see Section 6.2.1
2-32767	Unassigned	
32768-65535	Reserved	

- OSPFv3 BP LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv3 TBPLSA TLV Name	Definition
0	Reserved	
1	Node Backup Paths TLV	see Section 6.2.2
2	Link Backup Paths TLV	see Section 6.2.2
3-32767	Unassigned	
32768-65535	Reserved	

- OSPFv3 FTBP LSA TLVs

Initial values for the registry are given below. The future assignments are to be made through IETF Review [RFC5226].

Value	OSPFv3 FTBP LSA TLV Name	Definition
0	Reserved	
1	Links with Backup Paths TLV	see Section 6.2.2
2-32767	Unassigned	
32768-65535	Reserved	

10.3. IS-IS

Under Registry Name: IS-IS TLV Codepoints, IANA is requested to assign new TLV values for IS-IS flooding reduction as follows:

Value	TLV Name	Definition
151	FT Links TLV	see Section 7.2.1
152	Node Backup Paths TLV	see Section 7.2.2
153	Link Backup Paths TLV	see Section 7.2.2
154	Links with Backup Paths TLV	see Section 7.2.2
155	Add Links TLV	see Section 7.2.3
156	Delete Links TLV	see Section 7.2.3
157	Instruction TLV	see Section 7.1
158	Information TLV	see Section 7.2.4

11. Acknowledgements

The authors would like to thank Acee Lindem, Zhibo Hu, Robin Li, Stephane Litkowski and Alvaro Retana for their valuable suggestions and comments on this draft.

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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC5250] Berger, L., Bryskin, I., Zinin, A., and R. Coltun, "The OSPF Opaque LSA Option", RFC 5250, DOI 10.17487/RFC5250, July 2008, <https://www.rfc-editor.org/info/rfc5250>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 7770, DOI 10.17487/RFC7770, February 2016, <https://www.rfc-editor.org/info/rfc7770>.

12.2. Informative References

[I-D.li-dynamic-flooding]
Li, T. and P. Psenak, "Dynamic Flooding on Dense Graphs", draft-li-dynamic-flooding-05 (work in progress), June 2018.

[I-D.shen-isis-spine-leaf-ext]
Shen, N., Ginsberg, L., and S. Thyamagundalu, "IS-IS Routing for Spine-Leaf Topology", draft-shen-isis-spine-leaf-ext-06 (work in progress), June 2018.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

Appendix A. Algorithms to Build Flooding Topology

There are many algorithms to build a flooding topology. A simple and efficient one is briefed below.

- Select a node R according to a rule such as the node with the biggest/smallest node ID;
- Build a tree using R as root of the tree (details below); and then
- Connect k (k>=0) leaves to the tree to have a flooding topology (details follow).

A.1. Algorithms to Build Tree without Considering Others

An algorithm for building a tree from node R as root starts with a candidate queue Cq containing R and an empty flooding topology Ft:

1. Remove the first node A from Cq and add A into Ft
2. If Cq is empty, then return with Ft

3. Suppose that node Xi (i = 1, 2, ..., n) is connected to node A and not in Ft and X1, X2, ..., Xn are in a special order. For example, X1, X2, ..., Xn are ordered by the cost of the link between A and Xi. The cost of the link between A and Xi is less than the cost of the link between A and Xj (j = i + 1). If two costs are the same, Xi's ID is less than Xj's ID. In another example, X1, X2, ..., Xn are ordered by their IDs. If they are not ordered, then make them in the order.

4. Add Xi (i = 1, 2, ..., n) into the end of Cq, goto step 1.

Another algorithm for building a tree from node R as root starts with a candidate queue Cq containing R and an empty flooding topology Ft:

1. Remove the first node A from Cq and add A into Ft

2. If Cq is empty, then return with Ft

3. Suppose that node Xi (i = 1, 2, ..., n) is connected to node A and not in Ft and X1, X2, ..., Xn are in a special order. For example, X1, X2, ..., Xn are ordered by the cost of the link between A and Xi. The cost of the link between A and Xi is less than the cost of the link between A and Xj (j = i + 1). If two costs are the same, Xi's ID is less than Xj's ID. In another example, X1, X2, ..., Xn are ordered by their IDs. If they are not ordered, then make them in the order.

4. Add Xi (i = 1, 2, ..., n) into the front of Cq and goto step 1.

A third algorithm for building a tree from node R as root starts with a candidate list Cq containing R associated with cost 0 and an empty flooding topology Ft:

1. Remove the first node A from Cq and add A into Ft

2. If all the nodes are on Ft, then return with Ft

3. Suppose that node A is associated with a cost Ca which is the cost from root R to node A, node Xi (i = 1, 2, ..., n) is connected to node A and not in Ft and the cost of the link between A and Xi is LCi (i=1, 2, ..., n). Compute Ci = Ca + LCi, check if Xi is in Cq and if Cxi (cost from R to Xi) < Ci. If Xi is not in Cq, then add Xi with cost Ci into Cq; If Xi is in Cq, then If Cxi > Ci then replace Xi with cost Cxi by Xi with Ci in Cq; If Cxi == Ci then add Xi with cost Ci into Cq.

4. Make sure Cq is in a special order. Suppose that Ai (i=1, 2, ..., m) are the nodes in Cq, Cai is the cost associated with Ai,

and IDi is the ID of Ai. One order is that for any k = 1, 2, ..., m-1, Cak < Caj (j = k+1) or Cak = Caj and IDk < IDj. Goto step 1.

A.2. Algorithms to Build Tree Considering Others

An algorithm for building a tree from node R as root with consideration of others's support for flooding reduction starts with a candidate queue Cq containing R associated with previous hop PH=0 and an empty flooding topology Ft:

1. Remove the first node A that supports flooding reduction from the candidate queue Cq if there is such a node A; otherwise (i.e., if there is not such node A in Cq), then remove the first node A from Cq. Add A into the flooding topology Ft.

2. If Cq is empty or all nodes are on Ft, then return with Ft

3. Suppose that node Xi (i = 1, 2, ..., n) is connected to node A and not in the flooding topology Ft and X1, X2, ..., Xn are in a special order considering whether some of them that support flooding reduction (. For example, X1, X2, ..., Xn are ordered by the cost of the link between A and Xi. The cost of the link between A and Xi is less than that of the link between A and Xj (j = i + 1). If two costs are the same, Xi's ID is less than Xj's ID. The cost of a link is redefined such that 1) the cost of a link between A and Xi both support flooding reduction is much less than the cost of any link between A and Xk where Xk with F=0; 2) the real metric of a link between A and Xi and the real metric of a link between A and Xk are used as their costs for determining the order of Xi and Xk if they all (i.e., A, Xi and Xk) support flooding reduction or none of Xi and Xk support flooding reduction.

4. Add Xi (i = 1, 2, ..., n) associated with previous hop PH=A into the end of the candidate queue Cq, and goto step 1.

Another algorithm for building a tree from node R as root with consideration of others' support for flooding reduction starts with a candidate queue Cq containing R associated with previous hop PH=0 and an empty flooding topology Ft:

1. Remove the first node A that supports flooding reduction from the candidate queue Cq if there is such a node A; otherwise (i.e., if there is not such node A in Cq), then remove the first node A from Cq. Add A into the flooding topology Ft.

2. If Cq is empty or all nodes are on Ft, then return with Ft.

3. Suppose that node Xi (i = 1, 2, ..., n) is connected to node A and not in the flooding topology Ft and X1, X2, ..., Xn are in a special order considering whether some of them support flooding reduction. For example, X1, X2, ..., Xn are ordered by the cost of the link between A and Xi. The cost of the link between A and Xi is less than the cost of the link between A and Xj (j = i + 1). If two costs are the same, Xi's ID is less than Xj's ID. The cost of a link is redefined such that 1) the cost of a link between A and Xi both support flooding reduction is much less than the cost of any link between A and Xk where Xk does not support flooding reduction; 2) the real metric of a link between A and Xi and the real metric of a link between A and Xk are used as their costs for determining the order of Xi and Xk if they all (i.e., A, Xi and Xk) support flooding reduction or none of Xi and Xk supports flooding reduction.

4. Add Xi (i = 1, 2, ..., n) associated with previous hop PH=A into the front of the candidate queue Cq, and goto step 1.

A third algorithm for building a tree from node R as root with consideration of others' support for flooding reduction (using flag F = 1 for support, and F = 0 for not support in the following) starts with a candidate list Cq containing R associated with low order cost Lc=0, high order cost Hc=0 and previous hop ID PH=0, and an empty flooding topology Ft:

1. Remove the first node A from Cq and add A into Ft.

2. If all the nodes are on Ft, then return with Ft

3. Suppose that node A is associated with a cost Ca which is the cost from root R to node A, node Xi (i = 1, 2, ..., n) is connected to node A and not in Ft and the cost of the link between A and Xi is LCi (i=1, 2, ..., n). Compute Ci = Ca + LCi, check if Xi is in Cq and if Cxi (cost from R to Xi) < Ci. If Xi is not in Cq, then add Xi with cost Ci into Cq; If Xi is in Cq, then If Cxi > Ci then replace Xi with cost Cxi by Xi with Ci in Cq; If Cxi == Ci then add Xi with cost Ci into Cq.

4. Suppose that node A is associated with a low order cost LCa which is the low order cost from root R to node A and a high order cost HCa which is the high order cost from R to A, node Xi (i = 1, 2, ..., n) is connected to node A and not in the flooding topology Ft and the real cost of the link between A and Xi is Ci (i=1, 2, ..., n). Compute LCxi and HCxi: LCxi = LCa + Ci if both A and Xi have flag F set to one, otherwise LCxi = LCa HCxi = HCa + Ci if A or Xi does not have flag F set to one, otherwise HCxi = HCa If Xi is not in Cq, then add Xi associated with LCxi, HCxi and PH = A

into Cq; If Xi associated with LCxi' and HCxi' and PHxi' is in Cq, then If HCxi' > HCxi then replace Xi with HCxi', LCxi' and PHxi' by Xi with HCxi, LCxi and PH=A in Cq; otherwise (i.e., HCxi' == HCxi) if LCxi' > LCxi , then replace Xi with HCxi', LCxi' and PHxi' by Xi with HCxi, LCxi and PH=A in Cq; otherwise (i.e., HCxi' == HCxi and LCxi' == LCxi) if PHxi' > PH, then replace Xi with HCxi', LCxi' and PHxi' by Xi with HCxi, LCxi and PH=A in Cq.

5. Make sure Cq is in a special order. Suppose that Ai (i=1, 2, ..., m) are the nodes in Cq, HCai and LCai are low order cost and high order cost associated with Ai, and IDi is the ID of Ai. One order is that for any k = 1, 2, ..., m-1, HCak < HCaj (j = k+1) or HCak = HCaj and LCak < LCaj or HCak = HCaj and LCak = LCaj and IDk < IDj. Goto step 1.

A.3. Connecting Leaves

Suppose that we have a flooding topology Ft built by one of the algorithms described above. Ft is like a tree. We may connect k (k >=0) leaves to the tree to have a enhanced flooding topology with more connectivity.

Suppose that there are m (0 < m) leaves directly connected to a node X on the flooding topology Ft. Select k (k <= m) leaves through using a deterministic algorithm or rule. One algorithm or rule is to select k leaves that have smaller or larger IDs (i.e., the IDs of these k leaves are smaller/bigger than the IDs of the other leaves directly connected to node X). Since every node has a unique ID, selecting k leaves with smaller or larger IDs is deterministic.

If k = 1, the leaf selected has the smallest/largest node ID among the IDs of all the leaves directly connected to node X.

For a selected leaf L directly connected to a node N in the flooding topology Ft, select a connection/adjacency to another node from node L in Ft through using a deterministic algorithm or rule.

Suppose that leaf node L is directly connected to nodes Ni (i = 1,2,...,s) in the flooding topology Ft via adjacencies and node Ni is not node N, IDi is the ID of node Ni, and Hi (i = 1,2,...,s) is the number of hops from node L to node Ni in the flooding topology Ft.

One Algorithm or rule is to select the connection to node Nj (1 <= j <= s) such that Hj is the largest among H1, H2, ..., Hs. If there is another node Na (1 <= a <= s) and Hj = Ha, then select the one with smaller (or larger) node ID. That is that if Hj == Ha and IDj < IDa then select the connection to Nj for selecting the one with smaller

node ID (or if Hj == Ha and IDj < IDa then select the connection to Na for selecting the one with larger node ID).

Suppose that the number of connections in total between leaves selected and the nodes in the flooding topology Ft to be added is NLc. We may have a limit to NLc.

Authors' Addresses

Huaimo Chen
Huawei Technologies

Email: huaimo.chen@huawei.com

Dean Cheng
Huawei Technologies

Email: dean.cheng@huawei.com

Mehmet Toy
Verizon
USA

Email: mehmet.toy@verizon.com

Yi Yang
IBM
Cary, NC
United States of America

Email: yyietf@gmail.com

LSR Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 1, 2021

U. Chunduri
T. Eckert
Futurewei

September 28, 2020

Preferred Path Route Graph Structure
draft-ce-lsr-ppr-graph-04

Abstract

This document defines a graph structure for the Preferred Path Route (PPR) for IS-IS, OSPFv2 and OSPFv3 protocols. This structure helps further scale of the PPR and reduce domain level global entries needed in some data planes.

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], RFC8174 [RFC8174] when, and only when they appear in all capitals, as shown here.

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 April 1, 2021.

Copyright Notice

Copyright (c) 2020 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

Preferred Path Routing (PPR) is a routing protocol mechanism concerned with the creation of a routing path as specified in the PPR-Path objects. These can be signaled via appropriate IGPs (IS-IS, OSPFv2, OSPFv3) and indicate the path for a data plane identifier (PPR-ID). With this, all PPR capable nodes along that path establish forwarding state for the PPR-ID and any packet destined to the PPR-ID would use that path instead of the IGP computed shortest path to the destination.

PPR-Paths and relevant IGP extensions are defined in [I-D.chunduri-lsr-isis-preferred-path-routing] and [I-D.chunduri-lsr-ospf-preferred-path-routing]. In these IGP

extensions, PPR-Paths are described as a path structure, which is an ordered linear list of Path Description Elements (PDEs) starting with a sender PDE followed by zero or more transit PDE and finishing with the destination PED. PDEs can indicate the node, a link to the node and services on a node.

A separate PPR-ID is required for every possible PPR-Path, even if one is just a subset of another path with the same destination. To provide PPR-Paths from N possible source nodes to one destination node, N PPR-IDs are therefore necessary. To create full-mesh connectivity via PPR-Paths between N nodes, N^2 PPR-Paths and N^2 PPR-IDs would be needed. Even if PPR-Paths would only be used for a subset of connections, such as for high-value traffic in larger networks, this scale behavior is less than ideal.

To allow scalability, in-terms of number of PPR-IDs needed on the destination nodes, number of forwarding entries needed on the nodes in the paths (for overlapping paths), and to minimize the amount of PPR information needed in the control plane, this document introduces a PPR-Tree structure in Section 2.

The terminology in this document uses the more generic term of PPR Graphs instead of PPR Trees because it is extensible.

1.1. Acronyms

MPLS - Multi Protocol Label Switching

MSD - Maximum SID Depth

PDE - Path Description Element

PPG - Preferred Path Graph

PPR - Preferred Path Routing/Route

PPR-ID - Preferred Path Route Identifier, a data plane identifier

SID - Segment Identifier

SPF - Shortest Path First

SR-MPLS - Segment Routing with MPLS data plane

SRH - Segment Routing Header - IPv6 routing Extension header

SRv6 - Segment Routing with Ipv6 data plane with SRH

TE - Traffic Engineering

2. PPR Graph TLVs

2.1. IS-IS TLVs

This section describes the encoding of IS-IS PPR Tree TLV. This TLV can be seen as having 4 logical section viz., encoding of the PPR-Prefix (IS-IS Prefix), encoding of PPG-ID, encoding of path description with an ordered PDE (Path Description Element) Sub-TLVs, belonging to one or more Branch-IDs and a set of optional PPR attribute Sub-TLVs, which can be used to describe PPR Graph common parameters. Multiple instances of this TLV MAY be advertised in IS-IS LSPs with different PPG-ID Type and with corresponding Branch-ID/PDE Sub-TLVS. The PPR Graph TLV has Type TBD (suggested value xxx), and has the following format:

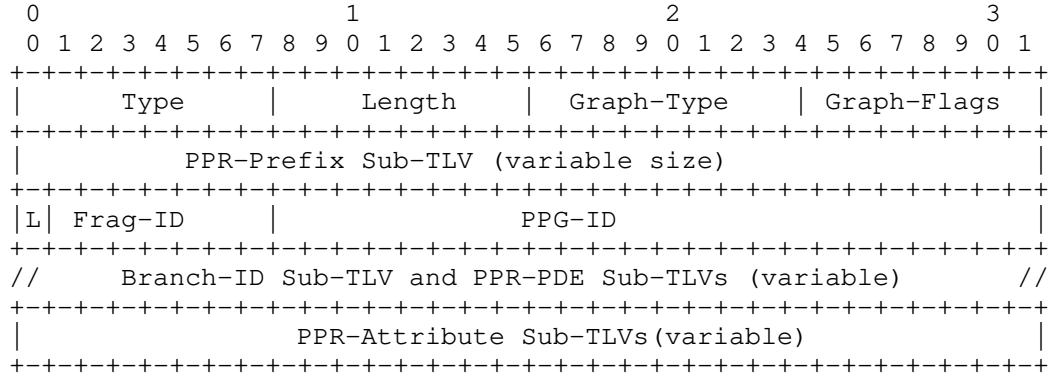


Figure 1: PPR Tree TLV Format

- Type - TBD (IANA) from IS-IS top level TLV registry.
- Length - Total length of the value field in bytes (variable).
- Graph-Type - 1 Octet value (0-255, IANA Registry TBD). Value 0 defines a PPR Tree structure (this document). PPR-Paths can also be encoded as PPR-Trees with a single branch.
- Graph-Flags - 1 Octet flags for this TLV are described below.
- Frag-ID - 1 Octet TLV Fragment-ID, with 7-bit Identifier value (0-127). L bit MUST be set if a graph has only one fragment or if it is the last Fragment of the graph. PPG-ID value for all fragments MUST be the same.

- PPG-ID - 3 byte Preferred Path Graph Identifier. Originator of the graph MUST ensure uniqueness across the domain.

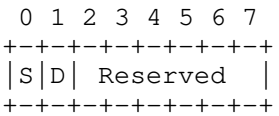
- Branch-ID Sub-TLV is defined in Section 2.1.1. This represents the branch-id of the structure followed by PDE Sub-TLVs in that branch. Different branches of the graph can be in different fragments of this TLV. However, a complete set of PDE Sub-TLVs MUST be specified in one TLV fragment.

- PPR-PDE Sub-TLV defined in [I-D.chunduri-lsr-isis-preferred-path-routing]. Additional information in the PPR-PDE Sub-TLV is described in Section 2.1.2.

- PPR-Attribute Sub-TLVs defined in [I-D.chunduri-lsr-isis-preferred-path-routing] are applicable here.

PPR-Flags field of PPR TLV has the following flag bits defined. These flags, at this point mostly related to applicability of this TLV in an L1 area or entire IS-IS domain or from where the PPR-Prefix is being originated:

PPR Graph-Flags Format



1. S - If set, the PPR Graph TLV MUST be flooded across the entire routing domain. If the S flag is not set, the PPR Graph TLV MUST NOT be leaked between IS-IS levels. This bit MUST NOT be altered during the TLV leaking

2. D - when the PPR Graph TLV is leaked from IS-IS level-2 to level-1, the D bit MUST be set. Otherwise, this bit MUST be clear. PPR TLVs with the D bit set MUST NOT be leaked from level-1 to level-2. This is to prevent TLV looping across levels.

3. Reserved - reserved bits for future use. Reserved bits MUST be reset on transmission and ignored on receive.

2.1.1. Branch-ID Sub-TLV

Branch-ID Sub-TLVs represent the branch of the graph described. This is a new Sub-TLV type (IANA TBD) in PPR TLV [I-D.chunduri-lsr-isis-preferred-path-routing]. Type TBD (Suggested

Value - IANA TBD), with a length of 1 byte, and Value is the branch identification number in the range of 0 to 255.

2.1.2. PPR PDE Sub-TLV

PPR PDE Sub-TLV is defined in [I-D.chunduri-lsr-isis-preferred-path-routing]. This document extends the same with the following:

1. PPR-PDE Flags (Bit position 2), S: Source Bit. Indicates the PPR head-end and MUST be set if this PDE corresponds to the same.

2. PPR-ID Sub-Sub-TLV: Type, length and value fields would be same as PPR-ID Sub-TLV defined in [I-D.chunduri-lsr-isis-preferred-path-routing]. This Sub-Sub-TLV MUST be present only when 'D' flag is set in the PPR-PDE Flags field.

PPR-PDE Flags field is defined in PPR-PDE Sub-TLV [I-D.chunduri-lsr-isis-preferred-path-routing].

2.2. OSPF TLVs

2.2.1. OSPFv2 TLVs

TBD.

2.2.2. OSPFv3 TLVs

TBD.

3. Encoding and Processing details

[I-D.chunduri-lsr-isis-preferred-path-routing] describes how a PPR path can be established. This document builds on the same base concept but expands the same with a graph structure as defined in Section 2. The key new encoding element here over prior PPR Paths is the existence of multiple Branches in the PPR Graph description.

Each Branch-ID sub-TLV is followed by ordered sequence of PDEs. A PPR Graph can be constructed from one or more PPR Branches. Branches are stitched together by using the same PDE in two branches. To simplify parsing of branches, only the last PDE of a branch can be stitched to another branch. In result, any PDE can only be a non-last PDE in one Branch but last PDE in more than one branch. A PPG-ID field is defined in this document. This MUST be unique in the domain and represents the graph structure as whole.

A complete Graph may not fit into maximum allowable size of the IS-IS TLV. To overcome this a 7 bit Frag-ID field is defined (Section 2). With this, a single PPR Graph is represented via one or more fragmented PPR Graph TLVs all having the same PPG-ID. Each Fragment carries the PPG-ID as well as a numeric Frag-ID from 0 to (N-1), when N fragments are needed to describe the PPR Graph (where N>1). In this case Fragment (N-1) MUST set the L bit to indicate it is the last fragment. The optional PPR Attribute Sub-TLVs which describe the Graph overall MUST be included in the last fragment only.

3.1. S And D bits in PDEs

In PPR Paths as defined in [I-D.chunduri-lsr-isis-preferred-path-routing], currently only a simple linear path structure for a destination node is possible. However, with a bit on path element source and a bit for destination (refer section) - same path ID/PPR-ID can be used to represent multiple paths if some of the nodes are also sources and terminating on the same destination node.

1. A Linear Path structure:
   PDE1 --> PDE2 --> PDE3 --> PDE4 --> PDE5
   [First PDE always Source and last PDE is always Destination]

2. A PPR Graph with S and D bits:
   PDE1(with-S-bit-set)-->PDE2-->PDE3(with-S-bit-set)..
   ..-->PDE4(with-D-bit-set)-->PDE5(with-D-bit-Set)

```
        ==>  PDE1 --> PDE2 --> PDE3 --> PDE4
        ==>  PDE1 --> PDE2 --> PDE3 --> PDE4 --> PDE5
        ==>  PDE3 --> PDE4
        ==>  PDE3 --> PDE4 --> PDE5
```

Figure 2: PPR Graph with S and D bits

In the above Figure 2 example, in (1) a linear path list of 5 nodes are described where PDE1 is the source/ingress-point and PDE5 is the destination/egress point of the path. In (2), the path can be defined in this document, where some PDEs can have S(ource) and/or D(estination) bit or both can be set. Here, PDE1 and PDE3 have the Source bit set, PDE4 and PDE5 the Destination bit set. This Branch structure is equivalent to the set of 4 PPR-PDE lists as shown: PDE1->PDE5, PDE1->PDE45, PDE3->PDE4, PDE3->PDE5. This reduces the amount of information that needs to be sent across the IGP and that needs to be processed by each node.

If the bits and branch structure were not used, the 4 PPR PDE lists would have required each a unique PPR-ID (and the resulting forwarding entries created), but the Branch requires only 2 PPR-IDs: one for both paths terminating in PDE4, and one for both paths terminating in PDE5.

3.2. Graph processing procedure example

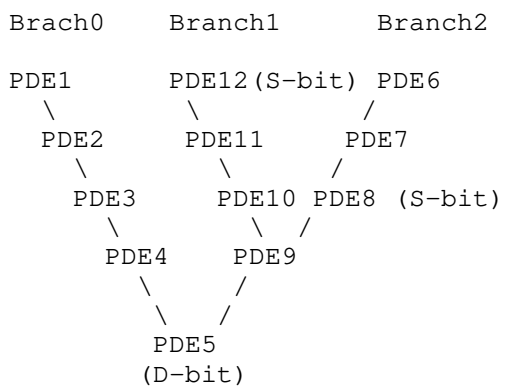


Figure 3: PPR Graph (Tree) Example

With a PPR Tree structure both flooding optimization and reduction in the number of SIDs needed at the destination can be achieved. To do this encoding as specified in Section 2 (a) Every PDE-ID can be non-last-PDE in at most one Branch. It can be last-PDE in one or more Branches (ex: PDE9). (b) Branches form a tree by joining nodes with same PDE-ID (PDE9 and PDE5 in the above example). Leafs of the tree must be S(ources), e.g.: PDE1, PDE12, PDE6. Root of the tree must be the only D(estination) of the tree (e.g.: PDE5).

How to build forwarding entry (referring to the Figure 3 above):

1. If PPR-ID in PDE of PPR Graph is indicating this node (example: PDE5): This node is D(estination) of this tree. Forwarding state is built for this PPR-Tree like for PPR-Path, no changes.

2. If PPR-ID is NOT indicating this node, then this node MAY be source (PDE12, PDE8) or midpoint (PDE9, neither source nor destination):

a. Node sequentially examines all branches until it finds a PDE with its own PDE-ID. It then establishes a forwarding entry for the PPR-ID indicated in the PPR header with the next-hop being the next PDE in the current branch.

b. This nodes PDE may be the last PDE in a Branch, for example PDE9 in Branch1. In this case, the node ignores this branch because it cannot build a complete forwarding entry from it. Instead, it will build the forwarding entry from another branch, e.g.: Node with PDE9 will build forwarding entry for destination PDE5 when it examines Branch2 because there it will have a next hop PDE5. After forwarding entry is built, node can stop examining rest of Branch or further Branches.

c. If node does not find its own PDE in any branch it is not on the graph and ignores this PPR-Graph.

4. Acknowledgements

Thanks to Yingzhen Qu and Richard Li for multiple discussions on this topic.

5. IANA Considerations

5.1. IS-IS IANA

This document requests the following new TLV in IANA IS-IS TLV code-point registry.

TLV #	Name
TBD	PPR Graph TLV

This document requests IANA to create a new Sub-TLV registry for PPR TLV Section 2 with the following initial entries (suggested values):

Sub-TLV #	Sub-TLV Name
TBD	Branch-ID (Section 2)

5.2. OSPFv2 IANA

5.3. OSPFv3 IANA

5.4. IGP Parameter IANA

This document requests additional IANA registries in an IANA managed registry "Interior Gateway Protocol (IGP) Parameters" for various PPR TLV parameters. The registration procedure is based on the "Expert Review" as defined in [RFC8126]. The suggested registry names are:

- "Graph-Type" - Types are an unsigned 8 bit numbers. Values are as defined in Section 2 of this document.

- "Graph-Flags" - 1 Octet. Bits as described in Section 2 of this document.

6. Security Considerations

Security concerns for IS-IS are addressed in [RFC5304] and [RFC5310]. Further security analysis for IS-IS protocol is done in [RFC7645] with detailed analysis of various security threats and why [RFC5304] should not be used in the deployments.

OSPF security extensions are described in [RFC2328] and [RFC7684] and these apply to the extensions specified in this document. While OSPF is under a single administrative domain, there can be deployments where potential attackers have access to one or more networks in the OSPF routing domain. In these deployments, stronger authentication mechanisms such as those specified in [RFC7474] SHOULD be used.

Advertisement of the additional information defined in this document introduces no new security concerns in IS-IS or OSPF protocols.

7. References

7.1. Normative References

[I-D.chunduri-lsr-isis-preferred-path-routing]
Chunduri, U., Li, R., White, R., Tantsura, J., Contreras, L., and Y. Qu, "Preferred Path Routing (PPR) in IS-IS", draft-chunduri-lsr-isis-preferred-path-routing-05 (work in progress), March 2020.

[I-D.chunduri-lsr-ospf-preferred-path-routing]
Chunduri, U., Qu, Y., White, R., Tantsura, J., and L. Contreras, "Preferred Path Routing (PPR) in OSPF", draft-chunduri-lsr-ospf-preferred-path-routing-04 (work in progress), March 2020.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

7.2. Informative References

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC7474] Bhatia, M., Hartman, S., Zhang, D., and A. Lindem, Ed., "Security Extension for OSPFv2 When Using Manual Key Management", RFC 7474, DOI 10.17487/RFC7474, April 2015, <https://www.rfc-editor.org/info/rfc7474>.

[RFC7645] Chunduri, U., Tian, A., and W. Lu, "The Keying and Authentication for Routing Protocol (KARP) IS-IS Security Analysis", RFC 7645, DOI 10.17487/RFC7645, September 2015, <https://www.rfc-editor.org/info/rfc7645>.

[RFC7684] Psenak, P., Gredler, H., Shakir, R., Henderickx, W., Tantsura, J., and A. Lindem, "OSPFv2 Prefix/Link Attribute Advertisement", RFC 7684, DOI 10.17487/RFC7684, November 2015, <https://www.rfc-editor.org/info/rfc7684>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Authors' Addresses

Uma Chunduri
Futurewei
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: umac.ietf@gmail.com

Toerless Eckert
Futurewei
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: tte+ietf@cs.fau.de

LSR Working Group
Internet-Draft
Intended status: Informational
Expires: May 28, 2021

U. Chunduri
Futurewei USA
J. Tantsura
Apstra, Inc.
S. Hegde
Juniper Networks

November 24, 2020

IS-IS Multi Topology Deployment Considerations
draft-chunduri-lsr-isis-mt-deployment-cons-04

Abstract

This document analyzes IS-IS Multi Topology (MT) applicability in various IS-IS deployments. This document explores the nuances around the terminology and usage of various IS-IS address families, topologies with different considerations, for choosing the right combination for a specific deployment scenario.

This document also discusses various ways one can deploy IPv6 only IS-IS topology.

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], RFC8174 [RFC8174] when, and only when they appear in all capitals, as shown here.

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 28, 2021.

Copyright Notice

Copyright (c) 2020 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

IS-IS originally developed for OSI [ISO.10589.1992] and extensions have been made available to support IPv4 [RFC1195]. A method for exchanging IPv6 routing information using the IS-IS routing protocol is specified in [RFC5308]. How to run a set of independent IP topologies with topology specific adjacencies, within a single IS-IS domain has been defined in IS-IS MT [RFC5120].

There are number of networks, including mobile backhaul networks seeking to use IPv6 only solutions. It is possible to conceive, various parts of the backhaul networks use IPv4 and appropriate migration strategy needed before eventually moving towards IPv6 only

network. While any IGP can be used in these networks, this document covers only IS-IS protocol aspects.

Various layer-3 DC fabric routing options (refs: openfabric, spine-leaf, controller-based) by changing or optimizing some aspects w.r.t adjacency formation, flooding optimizations, or/and mechanisms to automatically compute the location of the node in the fat tree topology are proposed recently and this document brings some of the multi topology deployment aspects relevant to these networks. Please note, part of the discussion around IS-IS MT is not specific to DC or CLOS fabrics and generally applicable to any IS-IS deployment but discussed here because of multiple proposals to use various forms of IS-IS in this context.

2. Need for MT in IS-IS networks

For mobile transport backhaul networks seeking only IPv6 network or transitioning from parts of the network with only IPv4, IS-IS MT is needed. For layer-3 DC fabric underlay, which provide reachability, only one address family (either IPv4 or IPv6) SHOULD be sufficient. However if either only IPv6 address family is needed in the underlay or deploying both IPv4 and IPv6 address families are desired discussion in Section 4 is relevant.

It is an unlikely requirement, where DC fabric to be partitioned logically to have different topologies in the underlay but this can happen in various scenarios as listed in Section 4.1. If one does the same to meet a particular requirement, it introduces a manageability complexity of these logical topologies. IS-IS MT [RFC5120] also designed to address the above need and discussion in Section 4.2 is relevant. It is worth noting, majority of the IS-IS deployments use MT primarily to have a separate logical topology for IPv6 address family.

3. Acronyms

IIH : IS-IS Hello Protocol Data Unit

LSP : Link State PDU

MT : Multi Topology

SPF : Shortest Path First

4. Topologies and Address Families

Terminology around IS-IS topologies and address families is somewhat confusing at best. Just to give an example, MT ID #2 defined in [RFC5120] says, it is "Reserved for IPv6 routing topology". While multiple MT ID's can be deployed in a network with IPv6 topologies, MT ID #2, perhaps referring to a first such topology with IPv6 only address family. This section details various topology and address family options possible with currently available IS-IS specifications with respective defined TLVs.

4.1. Single Topology Mode and Multiple Address Families

IS-IS with IPv4 address family and with wide-metrics [RFC5305] is widely deployed, with TLV 22 defined for IS Reachability and TLV 135 for IP (IPv4) reachability information . This is essentially a single topology for the entire IS-IS area/domain with a single address family (IPv4 unicast).

IS-IS can also be enabled with IPv6 unicast address family in a single topology mode along with IPv4 unicast address family. Here IPv6 uses the same underlying topology that is used for IPv4 and this can be done as specified in IS-IS IPv6 [RFC5308] which introduces TLV 236, an IPv6 reachability TLV. It is important to note same IS-IS adjacency is used for both address families and with a single SPF (decision process) both IPv4 and IPv6 reachability would be computed.

However, for the above to work effectively, both IPv4 and IPv6 address families MUST share a common network topology. That is to use IS-IS for IPv4 and IPv6 routing, any interface configured for IPv4 IS-IS MUST also be configured for IPv6 IS-IS, and vice versa. All routers within an IS-IS area (Level 1 routing) or domain (Level 2 routing) MUST also support the same set of address families: IPv4 only, IPv6 only, or both IPv4 and IPv6. Any discrepancy in the configuration w.r.t above can cause routing black holes and one such scenario is discussed below.

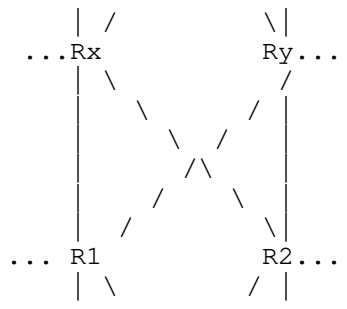


Figure 1: IS-IS with multiple address families

As shown, in the above diagram all routers in the network enabled with both IPv4 and IPv6 unicast address families at the IS level and single topology would be built. However, at a link level all but except one link, say if IPv6 is not configured on the link between the routers Rx and R2; due to a single IS-IS topology, the shortest path between Rx and R2 is the direct link and since IPv6 is not enabled on that link, Rx and R2 cannot exchange IPv6 data traffic even though there's an alternate path between them in the topology through Rx, R1, Ry and R2.

Hence to summarize the restrictions: all routers in the topology MUST support only IPv4, only IPv6 or both IPv4 and IPv6 address families on all links and node. In other words, network MUST be congruent. While this model is to simpler to operate, might not be flexible enough for some IS-IS deployments. Some examples where congruency is not possible as follows:

a. When IPv6 is getting introduced in the network legacy nodes that are IPv6 incapable.

b. Implementation issues causing IPv6 to be disabled on some nodes.

c. Hardware scale limitations causing IPv6 to be disabled on some low-end nodes.

4.2. Multiple Topology Mode and Multiple Address Families

Multi-topology IS-IS uses multiple SPFs to compute routes and removes the restriction that all interfaces MUST support all configured address families and that all routers in an IS-IS area or domain MUST support the same set of address families. This introduces the concept of topology specific adjacency with MT IS Reachability TLV 222 and MT capable IPv4 Reachability with TLV 235 and MT capable IPv6 Reachability with TLV 237.

When MT IS-IS is enabled with IPv4 and IPv6 address families, the routers build two topologies, one for each address family (IPv4 and IPv6) and can find the optimum path for each address family even when some links in the network support only one of them. IS-IS MT [RFC5120] defines MT ID #0 for backward compatibility, as the "standard" topology and this essentially operate as IS-IS single topology mode as specified in Section 4.1 and supports both IPv4 and IPv6 address families. MT ID #2 [RFC5120] is defined for IPv6 address family in MT mode.

4.2.1. Transition Mode

Most of the vendors supported MT transition feature (though some vendors disabled to avoid confusion around this) in the IS-IS networks to facilitate MT deployments without disrupting the single topology mode. The MT transition mode allows a network operating in single topology IS-IS IPv6 [RFC5308] to continue to work while upgrading routers to include MT IS-IS IPv6 support i.e., MT ID #2 with [RFC5120] . While in transition mode, both types of TLVs (single-topology with TLVs 22/236 and MT with TLVs 222/237) are sent in LSPs for all configured IPv6 addresses, nodes can continue to process these and operate in single topology mode though being in MT mode ("standard" IS-IS topology with MT ID #0). After all routers in the area or domain have been upgraded to support MT IPv6 transition mode can be removed from the configuration. Once all routers in the area or domain are operating in MT IPv6 mode, the topological restrictions of single-topology mode can be made no longer in effect.

When transition mode is enabled, the router advertises both MT TLVs and the old style IS-IS IPv6 TLVs but the topological restrictions of the single topology mode discussed above are in effect. However, there were instances while this mode is enabled and expectations for different result in the actual deployments.

4.3. IPv6 Only Topology

Though it is theoretically possible to build IPv6 only underlay (with TLV 236 for IPv6 reachability prefixes) in single topology mode as discussed in Section 4.1, lot of legacy implementations require IPv4 address families too be configured in single topology mode (ingrained code structures for IPv4 address family). IPv6 only DC underlay network can be built with multi topology adjacencies (TLV 222) and reachability prefixes (TLV 237) with MT ID #2 as discussed above in Section 4.2. With this, any other address family can be introduced including "standard" topology MT ID #0 (Single topology mode with both address families) and there are no restrictions on which address family has to enable on which link as specified in Section 4.1.

5. IS-IS MT and LFA

IP Fast Reroute (FRR) or Loop Free Alternative (LFA) computation in MT mode are described in detail in Section 5.2 of [RFC5120].

6. Acknowledgements

Thanks to Acee Lindem, Chris Hopps, Michael Abramson and Les Ginsberg for various inputs on this work.

7. IANA Considerations

This document has no actions for IANA.

8. Security Considerations

Security concerns for IS-IS are addressed in [RFC5304] and [RFC5310]. Further security analysis for IS-IS protocol is done in [RFC7645].

This document does not introduce any change in any of the IS-IS protocol or IS-IS protocol extensions. This document also does not introduce any new security issues other than as noted in the referenced IS-IS protocol extensions.

9. References

9.1. Normative References

[ISO.10589.1992]
International Organization for Standardization, "Intermediate system to intermediate system intra-domain-routing routine information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)", ISO Standard 10589, 1992.

[RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, DOI 10.17487/RFC1195, December 1990, <https://www.rfc-editor.org/info/rfc1195>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

9.2. Informative References

[RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <https://www.rfc-editor.org/info/rfc5120>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, DOI 10.17487/RFC5308, October 2008, <https://www.rfc-editor.org/info/rfc5308>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC7645] Chunduri, U., Tian, A., and W. Lu, "The Keying and Authentication for Routing Protocol (KARP) IS-IS Security Analysis", RFC 7645, DOI 10.17487/RFC7645, September 2015, <https://www.rfc-editor.org/info/rfc7645>.

[RFC8518] Sarkar, P., Ed., Chunduri, U., Ed., Hegde, S., Tantsura, J., and H. Gredler, "Selection of Loop-Free Alternates for Multi-Homed Prefixes", RFC 8518, DOI 10.17487/RFC8518, March 2019, <https://www.rfc-editor.org/info/rfc8518>.

Authors' Addresses

Uma Chunduri
Futurewei USA
2330 Central Expressway
Santa Clara, CA 95050
USA

Email: umac.ietf@gmail.com

Jeff Tantsura
Apstra, Inc.

Email: jefftant.ietf@gmail.com

Shraddha Hegde
Juniper Networks
Elnath-Exora Business Park Survey
Bangalore, Karnataka 560103
USA

Email: shraddha@juniper.net

LSR Working Group
Internet-Draft
Updates: 3563 5305 6232 6233 (if approved)
Intended status: Standards Track
Expires: October 5, 2019

L. Ginsberg
P. Wells
Cisco Systems
T. Li
Arista Networks
T. Przygienda
S. Hegde
Juniper Networks, Inc.

April 3, 2019

Invalid TLV Handling in IS-IS
draft-ginsberg-lsr-isis-invalid-tlv-03

Abstract

Key to the extensibility of the Intermediate System to Intermediate System (IS-IS) protocol has been the handling of unsupported and/or invalid Type/Length/Value (TLV) tuples. Although there are explicit statements in existing specifications, deployment experience has shown that there are inconsistencies in the behavior when a TLV which is disallowed in a particular Protocol Data Unit (PDU) is received.

This document discusses such cases and makes the correct behavior explicit in order to insure that interoperability is maximized.

This document when approved updates RFC3563, RFC5305, RFC6232, and RFC6233.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 October 5, 2019.

Copyright Notice

Copyright (c) 2019 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

The Intermediate System to Intermediate System (IS-IS) protocol utilizes Type/Length/Value (TLV) encoding for all content in the body of Protocol Data Units (PDUs). New extensions to the protocol are supported by defining new TLVs. In order to allow protocol

extensions to be deployed in a backwards compatible way an implementation is required to ignore TLVs that it does not understand. This behavior is also applied to sub-TLVs, which are contained within TLVs.

A corollary to ignoring unknown TLVs is having the validation of PDUs be independent from the validation of the TLVs contained in the PDU. PDUs which are valid MUST be accepted even if an individual TLV contained within that PDU is invalid in some way.

These behaviors are specified in existing protocol documents - principally [ISO10589] and [RFC5305]. In addition, the set of TLVs (and sub-TLVs) which are allowed in each PDU type is documented in the TLV Codepoints Registry (https://www.iana.org/assignments/isis-tlv-codepoints/isis-tlv-codepoints.xhtml) established by [RFC3563] and updated by [RFC6233] and [RFC7356].

This document is intended to clarify some aspects of existing specifications and thereby reduce the occurrence of non-conformant behavior seen in real world deployments. Although behaviors specified in existing protocol specifications are not changed, the clarifications contained in this document serve as updates to RFC 3563 (see Section 2), RFC 5304, and RFC 6233 (see Section 3).

2. TLV Codepoints Registry

[RFC3563] established the IANA managed IS-IS TLV Codepoints Registry for recording assigned TLV code points [TLV_CODEPOINTS]. The initial contents of this registry were based on [RFC3359].

The registry includes a set of columns indicating in which PDU types a given TLV is allowed:

IIH - TLV is allowed in Intermediate System to Intermediate System Hello (IIH) PDUs (Point-to-point and LAN)

LSP - TLV is allowed in Link State PDUs (LSP)

SNP - TLV is allowed in Sequence Number PDUs (SNP) (Partial Sequence Number PDUs (PSNP) and Complete Sequence Number PDUS (CSNP))

Purge - TLV is allowed in LSP Purges [RFC6233]

If "Y" is entered in a column it means the TLV is allowed in the corresponding PDU type.

If "N" is entered in a column it means the TLV is NOT allowed in the corresponding PDU type.

3. TLV Acceptance in PDUs

This section describes the correct behavior when a PDU is received which contains a TLV which is specified as disallowed in the TLV Codepoints Registry.

3.1. Handling of Disallowed TLVs in Received PDUs other than LSP Purges

[ISO10589] defines the behavior required when a PDU is received containing a TLV which is "not recognised". It states (see Sections 9.3 - 9.13):

"Any codes in a received PDU that are not recognised shall be ignored."

This is the model to be followed when a TLV is received which is disallowed. Therefore TLVs in a PDU (other than LSP purges) which are disallowed MUST be ignored and MUST NOT cause the PDU itself to be rejected by the receiving IS.

3.2. Special Handling of Disallowed TLVs in Received LSP Purges

When purging LSPs [ISO10589] recommends (but does not require) the body of the LSP (i.e., all TLVs) be removed before generating the purge. LSP purges which have TLVs in the body are accepted though any TLVs which are present "MUST" be ignored.

When cryptographic authentication [RFC5304] was introduced, this looseness when processing received purges had to be addressed in order to prevent attackers from being able to initiate a purge without having access to the authentication key. [RFC5304] therefore imposed strict requirements on what TLVs were allowed in a purge (authentication only) and specified that:

"ISes MUST NOT accept purges that contain TLVs other than the authentication TLV".

This behavior was extended by [RFC6232] which introduced the Purge Originator Identification (POI) TLV and [RFC6233] which added the "Purge" column to the TLV Codepoints registry to identify all the TLVs which are allowed in purges.

The behavior specified in [RFC5304] is not backwards compatible with the behavior defined by [ISO10589] and therefore can only be safely enabled when all nodes support cryptographic authentication. Similarly, the extensions defined by [RFC6233] are not compatible with the behavior defined in [RFC5304], therefore can only be safely enabled when all nodes support the extensions.

It is recommended that implementations provide controls for the enablement of behaviors that are not backward compatible.

3.3. Applicability to sub-TLVs

[RFC5305] introduced sub-TLVs, which are TLV tuples advertised within the body of a parent TLV. Registries associated with sub-TLVs are associated with the TLV Codepoints Registry and specify in which TLVs a given sub-TLV is allowed. As with TLVs, it is required that sub-TLVs which are disallowed MUST be ignored on receipt.

3.4. Correction to POI TLV Registry Entry

An error was introduced by [RFC6232] when specifying in which PDUs the POI TLV is allowed. Section 3 of [RFC6232] stated:

"The POI TLV SHOULD be found in all purges and MUST NOT be found in LSPs with a non-zero Remaining Lifetime."

However, the IANA section of the same document stated:

"The additional values for this TLV should be IIH:n, LSP:y, SNP:n, and Purge:y. "

The correct setting for "LSP" is "n". This document corrects that error.

4. TLV Validation and LSP Acceptance

The correct format of a TLV and its associated sub-TLVs if applicable are defined in the document(s) which introduce each codepoint. The definition SHOULD include what action to take when the format/content of the TLV does not conform to the specification (e.g., "MUST be ignored on receipt"). When making use of the information encoded in a given TLV (or sub-TLV) receiving nodes MUST verify that the TLV conforms to the standard definition. This includes cases where the length of a TLV/sub-TLV is incorrect and/or cases where the value field does not conform to the defined restrictions.

However, the unit of flooding for the IS-IS Update process is an LSP. The presence of a TLV (or sub-TLV) with content which does not conform to the relevant specification MUST NOT cause the LSP itself to be rejected. Failure to follow this requirement will result in inconsistent LSP Databases on different nodes in the network which will compromise the correct operation of the protocol.

LSP Acceptance rules are specified in [ISO10589] . Acceptance rules for LSP purges are extended by [RFC5304] [RFC5310] and further extended by [RFC6233].

[ISO10589] also specifies the behavior when an LSP is not accepted. This behavior is NOT altered by extensions to the LSP Acceptance rules i.e., regardless of the reason for the rejection of an LSP the Update process on the receiving router takes the same action.

5. IANA Considerations

IANA is requested to update the TLV Codepoints Registry to reference this document.

IANA is also requested to modify the entry for the POI TLV in the TLV Codepoints Registry to be:

IIH:n, LSP:n, SNP:n, and Purge:y.

6. Security Considerations

As this document makes no changes to the protocol there are no new security issues introduced.

The clarifications discussed in this document are intended to make it less likely that implementations will incorrectly process received LSPs, thereby also making it less likely that a bad actor could exploit a faulty implementaion.

Security concerns for IS-IS are discussed in [ISO10589], [RFC5304], and [RFC5310].

7. Acknowledgements

The authors would like to thank Alvaro Retana.

8. References

8.1. Normative References

[ISO10589]
International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routeing 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, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3563] Zinin, A., "Cooperative Agreement Between the ISOC/IETF and ISO/IEC Joint Technical Committee 1/Sub Committee 6 (JTC1/SC6) on IS-IS Routing Protocol Development", RFC 3563, DOI 10.17487/RFC3563, July 2003, <https://www.rfc-editor.org/info/rfc3563>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC6232] Wei, F., Qin, Y., Li, Z., Li, T., and J. Dong, "Purge Originator Identification TLV for IS-IS", RFC 6232, DOI 10.17487/RFC6232, May 2011, <https://www.rfc-editor.org/info/rfc6232>.

[RFC6233] Li, T. and L. Ginsberg, "IS-IS Registry Extension for Purges", RFC 6233, DOI 10.17487/RFC6233, May 2011, <https://www.rfc-editor.org/info/rfc6233>.

[RFC7356] Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, DOI 10.17487/RFC7356, September 2014, <https://www.rfc-editor.org/info/rfc7356>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[TLV_CODEPOINTS] IANA, "IS-IS TLV Codepoints web page (https://www.iana.org/assignments/isis-tlv-codepoints/isis-tlv-codepoints.xhtml)".

8.2. Informative References

[RFC3359] Przygienda, T., "Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System", RFC 3359, DOI 10.17487/RFC3359, August 2002, <https://www.rfc-editor.org/info/rfc3359>.

Authors' Addresses

Les Ginsberg
Cisco Systems

Email: ginsberg@cisco.com

Paul Wells
Cisco Systems

Email: pauwells@cisco.com

Tony Li
Arista Networks
5453 Great America Parkway
Santa Clara, California 95054
USA

Email: tony.li@tony.li

Tony Przygienda
Juniper Networks, Inc.
1194 N. Matilda Ave
Sunnyvale, California 94089
USA

Email: prz@juniper.net

Shraddha Hegde
Juniper Networks, Inc.
Embassy Business Park
Bangalore, KA 560093
India

Email: shraddha@juniper.net

LSR Working Group H. Smit, Ed.
Internet-Draft
Intended status: Standards Track G. Van de Velde
Expires: April 25, 2019 Nokia
October 22, 2018

IS-IS Sparse Link-State Flooding
draft-hsmit-lsr-isis-dnfm-00

Abstract

This document describes a technology extension to reduce link-state flooding in highly resilient dense networks. It does this by using simple and backwards-compatible extensions to reduce the number of adjacencies over which link-state flooding takes place.

"IS-IS Sparse Link-State Flooding" is an extension to the IS-IS routing protocol.

It is relatively easy to understand and implement. It is backwards compatible. It requires no per-node configuration. It uses a distributed algorithm, therefor no centralized computations are required. No complex computations are required on each node in the network. The algorithm has no requirements for the network topology. It can be deployed in a redundant way to improve robustness and convergence-times.

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 RFC 2119 [1].

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 April 25, 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

In dense network topologies, using massive ECMP or massive numbers of resilient links, the flooding algorithm of link-state protocols is highly redundant. This results in unnecessary overhead, potentially overloading control planes, decreasing robustness and slowing down convergence. Because of this percepted inefficiency, some operators have resorted to using BGP as the IGP in their data center networks. Draft-li-dynamic-flooding [3] describes this in more detail. However it is very clear that using an Exterior Gateway Protocol as an IGP is sub-optimal, if only due to the configuration overhead.

This document proposes a technology extension to reduce the number of interfaces over which a link-state protocol floods its updates in highly resilient networks. The result is a sparse flooding topology over a dense physical network topology. We describe details how to implement this algorithm for the IS-IS protocol [2]. This algorithm can be extended to other link-state routing protocols, like OSPF. However, no details for protocols other IS-IS are included in this document.

This proposal uses simple and backwards-compatible extensions. It is easy to understand and and relatively easy to implement for IS-IS coders. These proposed IS-IS extensions do not require additional configuration on every router. However, it might be beneficial for the operation of the algorithm to manually configure one or more routers as "anchors" in the network. The purpose of an "anchor" is explained in the next section of this document. This extension uses a distributed algorithm. No centralized calculations need to be performed. Each pair of routers decide for themselves where flooding can be suppressed. After ever regular SPF computation a router can adjust the interfaces over which it does flooding. This decision requires no computational-complex calculations.

2. High level overview of Sparse Link-State Flooding

The goal of the new Sparse Link-State Flooding algorithm is to create a tree of nodes and links, over which updates will be flooded. This tree is called "the flooding topology". The flooding topology includes all the nodes in the network. But it includes only a (small) subset of all available links in the physical network.

The idea is that the flooding topology starts at a single router in the network. This single router is called "the anchor". Routers that are adjacent to the anchor will "attach" or "clamp" themselves to the flooding topology. Making the flooding topology bigger. Their neighbors will "attach" themselves as well, making the flooding topology spread out. In the end all routers will be part of the

flooding topology. The flooding topology resembles a tree, with the anchor as the root of the tree.

The decision to flood or not flood over an adjacency is a local matter. This makes the algorithm a distributed algorithm. The flooding topology itself is not flooded through the network. Only the location of the anchor(s) is announced in LSPs. An anchor announces itself by including this information in its LSP. Two adjacent routers determine whether they need to exchange LSPs or not via a mechanism using a new TLV in hello PDUs (IIHs).

This algorithm can be run once, or multiple times in parallel. This creates one or more concurrent flooding topologies. This provides robustness and faster convergence to the flooding process. We envision that anchors are configured manually, like BGP's Route Reflectors. Or they can be elected automatically. For this the anchor-TLV contains a priority field, to allow operators to have influence on the location of the anchor(s).

3. The Sparse Link-State Flooding algorithm in detail

3.1. Role of the Anchor

Each flooding topology needs a root of its tree. The router acting as root is called "the anchor" of a flooding topology. An anchor router includes information in its LSP to announce that it wants to function as an anchor. This information can be encoded as a new TLV, or as a new capability in the existing IS-IS capability TLV. This choice is open for discussion.

The content of this new TLV includes a priority. If multiple routers advertise their willingness to act as an anchor, the anchor with the highest priority is chosen as the anchor. If multiple potential anchors have the same priority, then the router with the highest system-id is chosen as the anchor.

Besides announcing itself as an anchor in its LSP, the role of the anchor-route is purely passive. No extra actions are required of the anchor.

3.2. Bootstrapping the flooding

When a router boots, or when a new adjacency comes up, routers need to synchronize their LSDBs. The reason is that a network could have been partitioned in two separate parts. And flooding over the new adjacency might be the only way to make the two parts of the network aware of each other.

After the LSDBs are synchronized, and at least one SPF computation has been executed, the new algorithm can be used. An implementation could use a longer grace period to wait before using the new algorithm, to ensure all or most of the LSPs in a network have been received.

3.3. Determining which adjacency a router wants to flood over

The decision to do regular flooding, or suppress flooding, is done as follows. After each SPF computation, a router looks at the newly computed route towards the anchor. Each router wants to do flooding over the adjacency to a router that is closer to the anchor than it is itself. This guarantees that each router will do flooding with a router that is already part of the flooding topology.

If there are multiple (equal-cost) paths towards the anchor, one of the next-hop adjacencies of the route towards the anchor is chosen to flood over. It doesn't matter which adjacency that is, as long as the adjacent router is closer to the anchor.

When the flooding topology breaks, the two routers next to the point of breakage will notice. They will each generate a new LSP. And they will send out that new LSP over the old flooding topology. The LSP generated by the router that is still reachable through the old flooding topology will be received by all routers on their side of the breakage. This will trigger new SPF computations on all those routers. This SPF computation will compute a new path towards the anchor. The routers will now adjust their flooding topology according to the new path they have just computed. All routers in the network do this. New LSPs will be flooded over the new flooding topology. Which might trigger a follow-up SPF computation. Which might cause routers to adjust their flooding topology again. After a while all routers will have received all new LSPs. Which will guarantee that they will all compute a new correct flooding topology.

A requirement is that when routers start using an adjacency for their flooding topology, they need to synchronize LSDBs first. This is done by exchanging CSNPs. This can potentially be done more reliable and faster when doing IS-IS Flooding over TCP [4].

3.4. Determining where flooding can be suppressed

The decision whether to flood over an adjacency or not is a local matter. Only the two routers of the adjacency are involved in this decision. Both routers have a say in whether flooding will be suppressed or not.

This document defines a new TLV, called the Flooding-Suppression TLV, to be included in Hello PDUs (IIHs). This new TLV includes a field that indicates whether a router wants to do flooding over this interface, or wants to suppress flooding. The content of this TLV is set according to the decision made after each SPF, as explained in the previous section of this document.

As a result, a router keeps two new pieces of state for each adjacency.

- Does the router itself want to flood over this adjacency ? We'll call this the adjacency's "suppression-local-request-state".

- Does the neighbor want to flood over this adjacency ? We'll call this the adjacency's "suppression-neighor-request-state".

The suppression-local-request-state is determined after each SPF computation.

The suppression-neighbor-request-state is learned from examining the Flooding-Suppression TLV in each received IIH. If a router did not include the new Flooding-Suppression TLV in its IIH, it is assumed that the neighbor does want to flood over this adjacency.

When both "suppression-local-request-state" and "suppression-neighbor-request-state" are true, then the overall "suppression-state" of the interface is set to true. In that case flooding over the interface is to be suppressed. In all 3 other cases, where at least one of the two routers does not want to suppress flooding, flooding is done in the normal way.

So flooding over an adjacency is only suppressed when both neighbors have indicated that they want to suppress flooding over the adjacency. This means that when one of the two routers does not support this new algorithm, and thus does not include the new TLV in its IIH, flooding is always done. This makes the algorithm backwards compatible with routers that do not support this new extension of the protocol.

A router will always have one or more flooding adjacencies. One adjacency that the router itself needs, to "clamp" on to the part of the flooding topology that is closer to the anchor than it is itself. This adjacency points towards the anchor. And zero or more adjacencies that its neighbors, downstream of the anchor, use to clamp themselves onto the flooding topology. These adjacencies point away from the anchor.

4. Using multiple concurrent flooding topologies

It is possible to use more than one flooding topology in parallel. This requires more than one anchor. For each anchor a new flooding topology is built. These flooding topologies can co-exist without problems.

All that is required is that after each SPF computation, the router examines the shortest path to each anchor. And sets the local state of each adjacency according to this. This guarantees that the router will "clamp onto" each flooding topology.

To ensure an optimal use of parallel flooding topologies, all routers in an IS-IS flooding domain (area or level-2 backbone) should use the same number of parallel flooding topologies. This can be done through configuration. Or an easier way would be to include the number of parallel flooding topologies to use, inside the new Anchor TLV. When looking for Anchors, a router must first find all LSPs with the new Anchor TLV. It then selects the router with the highest Anchor-priority as the main anchor. If multiple router use the same priority, the router with the highest system-id is selected as the anchor. Once the main anchor has been determined, a router looks inside the new anchor-TLV to determine how many parallel flooding topologies it should use. It then selects that amount of anchors with the highest priorities, to set the flooding-state of adjacencies pointing towards those anchors.

Flooding suppression is a local matter. Therefore an implementation can decide to flood over more adjacencies than the minimum to build the minimal flooding topology. It can signal this through the Flooding-Supression TLV in its IIHs. This can improve robustness and convergence times, at the cost of some extra flooding overhead.

5. Benefits of the Sparse Link-State Flooding algorithm

The algorithm described in this document has a number of advantages.

- The algorithm is a distributed algorithm. Distributed algorithms are usually more robust than centralized algorithms. The flooding topology itself does not need to be flooded, which makes the algorithm easier when the flooding topology breaks.

- The algorithm is backwards compatible. No flag-day is required to introduce this new sparse-flooding extension. Older routers that do not support the new extension will obviously not include the flooding-state TLV in their IIHs. The result of this is that regular flooding is done over all adjacencies of those older

routers. This guarantees that older routers will never break the flooding topology.

- No extra computations have to be done to compute the flooding topology. Using the result of the regular SPF computation suffices to determine over which adjacencies a router wants to flood.

- The proposed algorithm is robust and guarantees that a flooding topology eventually heals so that all routers are included in the flooding again.

- Several instances of the algorithm can be run in parallel. This results in multiple parallel flooding topologies. Although parallel flooding topologies are not required for correct operation of the algorithm, it will help in speeding up the healing of the flooding topology. And thus convergence times in general.

6. Extensions to IS-IS PDUs

To implement this algorithm, we need two extensions of IS-IS PDUs.

6.1. Anchor TLV in LSPs

A new Anchor TLV in the LinkState PDUs. This TLV indicates that a router can be used as an anchor. This new TLV must include a priority field. And it should include a field that suggests how many parallel flooding topologies all routers should use.

6.2. Flooding-Suppression TLV in IIHs

A new Flooding-Suppression TLV in the IIH PDUs. This TLV is used to indicate to the neighbor if a router wants to suppress flooding over the adjacency. This new TLV holds three fields:

- Flooding suppression suggestion field: this field indicates whether the sending router would like to suppress flooding over this interface or not. The value of this field is set to the current "suppression-local-request-state". Note, only when two routers both indicate they want to suppress flooding, then flooding will indeed be suppressed.

- Resulting actual suppression field: this field indicates whether the sending router will or will not do flooding. The value of this field is set to the current "suppression-state" of the interface. This field is included only for debugging purposes. The first field (the received suppression-local-request-state

field) is used to make the flooding decision. The result of that decision is announced in the second field.

- The number of currently active flooding adjacencies. This field can be used by the receiving router to pick a flooding adjacency when there are multiple ECMP paths towards the anchor. A router can pick the upstream router with the least amount of flooding adjacencies. In dense networks with many parallel paths, this can help spreading out the load of flooding equally over multiple routers.

Backward compatibility: when a router does not include the Flooding-State TLV in the IIHs it sends out, it can be treated as if that router included the Flooding-State TLV while setting the first field to: "I do not want to suppress flooding".

7. Operations of the new Sparse Link-State Flooding algorithm

7.1. Flooding at the anchor itself

When a router is acting as the anchor, it floods over all its interfaces. It does include the Flooding-Suppression TLV in its IIHs, but it always sets the value inside the new TLV to "I do not want to suppress flooding".

7.2. New action after each SPF

At the end of each SPF computation, a router looks at the best-path to reach the anchor-router. The router sets the "suppression-local-request-state" for that adjacency to false. The router sets the "suppression-local-request-state" for all other adjacencies to true.

If the best-path to the anchor-router's is load-balanced over multiple adjacencies, the router picks one of those adjacencies as its own "upstream flooding adjacencies".

A router must take effort to ensure it changes its "upstream flooding adjacency" as little as possible. Switching its upstream flooding adjacency is not without cost. Every time an adjacency changes from suppressed flooding to normal flooding, the LSDBs of the two routers must be synchronized.

If the "suppression-local-request-state" changed for one or more adjacencies, compared to the state after the previous SPF computation, the router will re-compute the "suppression-state". If the "suppression-state" of an adjacency changes, the router will start or stop flooding over that adjacency.

7.3. When sending a IIH

When a router sends an IIH, it includes the new Flooding-Suppression TLV.

For adjacencies that were selected as "upstream flooding adjacency", the value of the Flooding-Suppression TLV must be set to: "I do not want to suppress flooding". For all other adjacencies the value must be set to: "I do want to suppress flooding".

7.4. When receiving a IIH

When a router receives an IIH, it checks for the existence of the new Flooding-Suppression TLV.

If it there is none, the state of the neighbor is assumed to be: "I do not want to suppress flooding".

If the "suppression-remote-request-state" changed for this adjacency, compared to the state after receiving the previous IIH, the router will re-compute the "suppression-state". If the "suppression-state" of an adjacency changes, the router will start or stop flooding over that adjacency.

7.5. When installing a new LSP in the LSDB

When a router receives a new LSP, it installs it in the LSDB. It will normally then set the IS-IS SRM (Send Routing Message) bits for all adjacencies (in UP state). Now, with the new algorithm, it will set SRM-bits for only the adjacencies that are part of the reduced flooding topology.

7.6. Preventing loops in the flooding topology

When the flooding topology changes, during a short period of time different routers can have a different view of the flooding topology. This can make the actual flooding topology in use be a random cyclic graph, instead of a non-cyclic tree. This is not a problem. The flooding algorithm in link-state protocols deals with this by default. An LSP is only (scheduled to be) flooded the first time it is received and installed in the LSDB.

The Sparse Link-State Flooding algorithm has some resemblance to the Spanning Tree Protocol used by transparent bridges. Transient forwarding loops can be a huge problem in the operation of a SPT network. However, while the flooding topology can be looping for short periods of time, this is not a problem at all. Because as described in the previous paragraph, link-state flooding will take

care of this by default. This works because routers keep copies of the LSPs they forward in their LSDB. This allows them to determine if they have received an LSP before or not. In STP routers have no recollection of data-frames that they have forwarded in the past. So in STP looping frames can not be recognized as looping.

To improve convergence times during changes of the flooding topology it is recommended that when a router changes the state of an adjacency from flooding to non-flooding, both routers keep flooding over this adjacency for a short period of time. A suggested value for this is 30 or 60 seconds. By doing this, during changes of the flooding topology, both the old and the new topology will be in use. This guarantees that LSPs are flooded as quickly as possible. This will also help in repairing the flooding topology itself.

7.7. Fall-back to classic full flooding

When a router thinks it might have got behind on flooding, it can always fall back to normal flooding behaviour. It omits including the Flooding-Suppression TLV from its IIHs. Consequently, classic flooding will allow guaranteed synchronization of its IS-IS LSDB with all neighbors. This can be done on all adjacencies at once, or on subset.

8. Security Considerations

This draft introduces no new security considerations.

9. IANA Considerations

This document requests a new TLV and sub-TLV for IS-IS.

10. References

10.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://xml.resource.org/public/rfc/html/rfc2119.html>.

10.2. Informative References

[2] International Standard 10589, "Intermediate System to Intermediate System intra- domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473), Second Edition.", 2002.

[3] Li, T. and P. Psenak, "Dynamic Flooding on Dense Graphs", June 2018.

[4] Smit, H. and G. Van De Velde, "IS-IS Flooding over TCP", October 2018.

Authors' Addresses

Henk Smit (editor)
NL

Email: hhw.smit@xs4all.nl

Gunter Van de Velde
Nokia
Copernicuslaan 50
Antwerp
BE

Email: gunter.van_de_velde@nokia.com

Link-State Routing
Internet-Draft
Intended status: Standards Track
Expires: April 15, 2019

H. Smit, Ed.

G. Van de Velde
Nokia

October 12, 2018

IS-IS Flooding over TCP
draft-hsmit-lsr-isis-flooding-over-tcp-00

Abstract

This document proposes a solution to use TCP for IS-IS flooding. The proposed solution is a relative simple extension to implement. IS-IS flooding over TCP brings BGP's property of scalable transport via TCP to Link-State protocols.

This proposal defines a new TLV in point-to-point IIHs to signal the intent of a router to do flooding over TCP, and it defines a small header to encapsulate IS-IS PDUs in the TCP byte-stream.

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 RFC 2119 [1].

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 April 15, 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

IP Fabric Networks in data-centers are growing larger and larger. The number of routers in such fabrics are pushing the boundaries of routing protocols. Therefor new ideas are explored, and existing protocols are being enhanced (RFC 7938 [2], RIFT [5] and LSVR [4]).

This document improves an existing protocol, IS-IS. One of the scaling limitations of IS-IS is its flooding algorithm. BGP is known to be a routing protocol of high scale. A key property and important benefit of BGP is the fact that BGP uses TCP as transport mechanism. Introducing TCP to IS- IS flooding would bring a major positive scaling property from BGP to IS-IS.

2. General scaling factors of IS-IS

IS-IS is a highly scalable Interior Gateway Protocol (IGP). IS-IS is defined in ISO-10589 [3]. Networks with thousands of routers have been deployed. When bigger networks are build, certain parts of the algorithm become a limitation to the scalability of IS-IS.

Several sub-components of the IS-IS protocol have an impact on its scalability.

- The number of adjacenies. For each adjacency periodic IIHs have to be exchanged. Each adjacency has to be included in the router's Link-State PDU (LSP). When building a dynamic routing protocol, this work has to be done in some form or another. Not much can be done to improve the scalability of this effort.

- Flooding has a large impact on scalability of IS-IS. Obviously the number of LSPs in an area has an impact on the operation of IS-IS. Also the number of interfaces over which a router must flood has an impact on the operation of IS-IS. But the flooding algorithm itself has elements that limit scalability. Improving these sub-algorithms will have a positive impact on scalability.

- Dijkstra's Shortest-Path First algorithm. This algorithm is at the heart of Link-State protocols. This algorithm is computationally reasonably efficient. One could build better implementations, that do partial route-computation and do incremental SPF. Or that check the bi-directionality of each link in advance of running the SPF. One could run the regular SPF and the computations for LFA and rLFA in parallel. But the SPF algorithm itself can not be improved upon easily.

2.1. Current scaling limitations of IS-IS flooding

With current implementations of the IS-IS protocol, the flooding algorithms have the largest impact on protocol scalability. Three issues are particularly of concern.

2.1.1. Packet pacing and throughput

The first issue is packet pacing of LSPs. If routers would send large bursts of routing protocol packets to other routers, there is a risk that the receiving router might drop those packets. This risks increases when a router has multiple neighbors that might all be sending large amount of routing-protocol packets at the same time. Dropped packets cause re-transmissions, delays in convergence, or even worse things. The solution is packet pacing.

ISO-10589 suggests a router should wait 30 milliseconds between sending of two consecutive LSPs. This will give the receiving router time to process pending incoming packets, before input-queues get overwhelmed. This means that two routers can exchange at most 33 LSPs per second. If a router boots, and has an empty LSDB, in a network with 10000 routers (and thus at least 10000 LSPs), it will take up to 300 seconds before the new router has acquired the full LSDB.

Decreasing the inter-packet gap will speed this up, but it might have a negative impact on overall network stability. More dynamic or adaptive packet-pacing algorithms could be envisioned, but those are not public nor standardized. If such algorithms would be developed, they would probably end up including many aspects of the existing TCP protocol.

2.1.2. Reliable flooding on point-to-point interfaces

The second issue is implementing reliable flooding over point-to-point interfaces. The following algorithm is used when a LSP needs to be flooded:

- When a new LSP is received, the router sets the SRM-bits for this LSP for all interfaces (except the interface on which the new LSP was received).
- For each interface a pacing-timer is started (if not running yet).
- When that pacing-timer expires, the router will find an LSP with its SRM-bit set for that interface. It will transmit the LSP out over the interface.

- The router will then clear the SRM-bit for that LSP. It will set a bit indicating that this LSP has not been acknowledged yet. And it will start a retransmit-timer for that LSP/interface combination.

- When a PSNP is received for this LSP on this interface, the router will clear the bit that indicates that no acknowledgement was received yet.

- When the retransmit-timer fires, the router will check whether the retransmit-it has been cleared yet. If so, the router stops the retransmit-timer and is done. If the retransmit-bit has not been cleared, then the router sets the SRM-bit for this interface/LSP combination again, and start the pacing-timer for the interface (if not still running).

Note that when flooding LSPs, the router needs to keep a retransmit-timer per LSP/interface combination. These timers run typically for 5 seconds, or until an acknowledgement (PSNP) is received. In a network with only a few hunderd LSPs, then 5 seconds * 33 LSPs/second results in only 165 LSPs being flooded. If the router has 100 interfaces, this can cause the router to have 16500 simultanously running timers. If a router falls behind processing PSNPs, or when PSNPs are being dropped, this number could increase to even larger numbers. The conclusion is that reliability of flooding LSPs over point-to-point interfaces does not come free. And in networks under stress, the cost can become even higher.

2.1.3. Unreliability of CSNPs

The third issue of concern is the unreliability of CNSPs. CSNPs are used when flooding over multi-point interfaces. But CSNPs are also used to synchronize LSDBs over adjacencies on point-to-point interfaces. This happens right after a new adjacency over a point-to-point interface is established. The algorithm used after a new adjacency comes up is:

- The router sets the SRM-bit for this interface on all LSPs in its LSDB.

- The router creates CSNPs describing all LSPs in its LSDB. It sends these CSNPs to the new neighbor.

- The router waits a limited amount of time, hoping to receive all the CSNPs from its new neighbors.

- For every LSP in every CSNP received from its new neighbor, the router checks to compare its version of the LSP with its neighbors

version of the LSP. If the versions are the same, the router clears the SRM-bit for that LSP/interface. Versions are compared using the LSP sequence-number (and checksum, TTL, etc).

- The router starts the packet-pacing timer, and starts sending to the new neighbor, LSPs that still have the SRM-bit set for that interface.

When the number of LSPs in the LSDB grows to large numbers, the number of CSNPs needed increases to large numbers as well. There can be only descriptions of 91 LSPs in a typical CSNP. If a network has 10000 routers, and thus 10000+ LSPs, it takes 110 CNSPs to describe the whole LSDB. If any of the CSNPs that get exchanged during adjacency synchronization gets dropped, the sending router will transmit 91 LSPs per dropped CNSPs, regardless whether that was necessary or not.

3. Improvements for IS-IS flooding

BGP is considered to be a highly scaleable routing protocol. It is used to carry all routes in the Global Internet. It is used to carry large numbers of customer routes in Provider networks that supply VPN-services. But BGP has downsides too. BGP typically requires extra configuration, and in dense topologies routing-churn can be experienced, because BGP does so-called path-hunting.

The main property of BGP that contributes to good scaling is the fact that BGP uses TCP for its transport. Using TCP has certain benefits for a routing protocol. TCP supplies reliability through retransmissions and acknowledgements. TCP supplies high throughput through its windowing mechanism and by potentially packing small chunks of user-data into larger TCP segments. TCP supplies a crude form of multi-threading by separating transmission and retransmission of data from the user process, and letting other tasks or the kernel take care of that. When a routing protocol uses TCP, it does not need to burden itself anymore with tasks like retransmission, acknowledgements, flow-control, or seeking high bandwidth and throughput. It also doesn't need to do extras to use multi-threading for reliable transmission.

3.1. Using TCP to do IS-IS flooding

This document proposes a relatively simple way to do IS-IS flooding over TCP.

Routers remain to establish new adjacencies using IIHs via the classic IS-IS mechanism. When using IS-IS TCP extensions Routers remain sending periodic IIHs via the classic mechanism to maintain

adjacencies. However, after establishing a new adjacency and successfully establish a corresponding TCP-connection, LSPs and SNPs are sent only over the TCP-connection.

4. Negotiating Flooding over TCP

Before two routers can start flooding over TCP, they need to agree on this new way of transport. Negotiating is done via a new TLV in the IS-to-IS Hello PDUs (IIH). When a router intends to do flooding over TCP, it includes this new TLV in its p2p IIHs. The existence of the TLV in the IIHs is an indication to the other router that it wants to use TCP for flooding.

The size of the TLV is variable. The value field contains the IP address and TCP port-number on which the router is accepting a new TCP connection for flooding. The router with the lowest System-ID initiates the TCP connection to the other router. The router with the highest System-ID never tries to set up a new connection. It just listens on its advertised TCP port-number and accepts the TCP connection from the router with the lower System-ID.

If only one, or neither of both routers include the new TLV in their IIHs, then flooding will not be done over TCP. Instead the classic IS-IS flooding algorithm is used, as described in ISO-10589.

Flooding over TCP is only supported on point-2-point interfaces.

4.1. The new TLV to indicate that a router wants to flood over TCP

This document proposes a new TLV for IIH messages. The existence of this TLV in an IIH signals the receiving router that the sending router is willing to do flooding over TCP.

The content of the TLV are 2 or more sub-TLVs. These sub-TLVs indicate the TCP port-number on which the advertising router is listening to accept new TCP-connections, and 1 or more sub-TLVs that indicate the IPv4- or IPv6-address on which the router is listening to accept new TCP-connections.

The new TLV consists of:

- 1 octet of TLV-Type,
- 1 octet of TLV-Length,
- 10 to 255 octets of TLV-Value, containing sub-TLVs.

The defined sub-TLVs are:

- TLV type 1, Length 2 octets. TCP port number.

- TLV type 2, Length 4 octets. IPv4 address. The sending router is listening on this IPv4-address to open a TCP-connection for IS-IS flooding.

- TLV type 3, Length 16 octets. IPv6 address. The sending router is listening on this IPv6-address to open a TCP-connection for IS-IS flooding.

Example of the layout of the new Flooding-over-TCP TLV. This example advertises an IPv4-address to connect to.

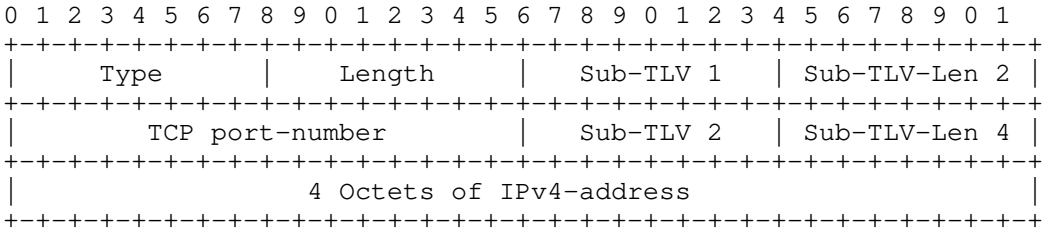


Figure 1

The new TLV is to be included only once in each IIH. A router MUST NOT include more than one TCP port number sub-TLV. A router MAY include multiple IPv4- or IPv6-address sub-TLVs. The destination IP-address(es) SHOULD be addresses that are also included in the IP-Interface-Addresses TLVs (TLV 132 for IPv4 or TLV 233 for IPv6).

5. Format of messages over TCP

The content of the messages that are transmitted over the TCP connection are traditional IS-IS PDUs. IIHs, SNPs and LSPs can all be transmitted over the TCP connection. No TLV-format or other extensible format is needed, because new information is to be included inside IIHs, SNPs or LSPs themselves. Therefor the format of messages over TCP itself does not need to be changed, and does not need to be extensible.

Each IS-IS PDU that is sent over TCP is to be preceeded by a header, functionaing as a marker. This header consists of:

- Four octets of marker. The content of this marker is always 0x69 0x73 0x69 0x73. This marker has the same function as the marker

in a BGP-header. It enables the receiver to check whether messages inside the TCP-bytestream have gone out of sync.

- Two octets of message-length. The IS-IS PDU itself also has a length-field, inside the message-specific header. The length-field here can be used to verify no octets are missing and that there are no extra trailing octets.

The type of IS-IS PDU can be derived from the PDU itself, by looking at the "PDU Type" field in the common IS-IS PDU header.

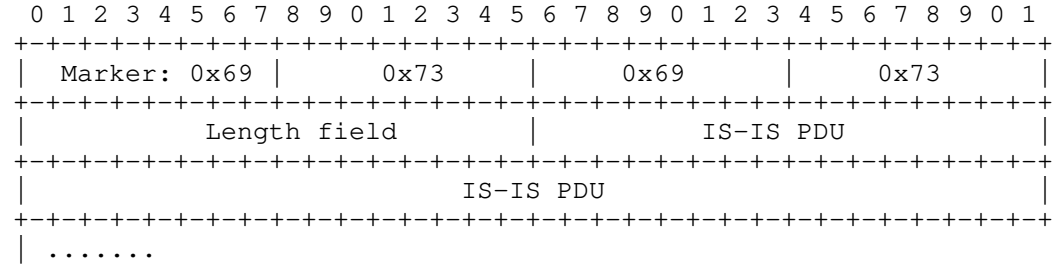


Figure 2

6. New behaviour of IS-IS flooding

When IS-IS does its flooding over TCP, the algorithms to transmit and receive LSPs change slightly. The biggest difference with the standard algorithms from ISO-10589 are the facts that the sending router does not need to do retransmission and that the receiving router does not need to send PSNPs to acknowledge receipt of LSPs.

6.1. Establishing a new IS-IS adjacency

Initially the Router looks for the new TLV in the IIH. If the other router included this TLV in its IIH, flooding over TCP is initiated. The router with the lowest System-ID initiates a TCP-connection to the other router. The TCP port-number and destination IP-address is learned from the new TLV in the IIH.

After the TCP session is established, a router will:

- Send a regular IIH over the TCP connection. The IIH is the same as when it would have been encapsulated straight into a layer-2 header. This IIH allows the other router to verify the identity and authentication of the remote router.

- Wait for receipt of an IIH from the remote router. This IIH is used to verify the identity and authentication of the remote router.

- Set the SRM-bit for this interface on all the LSPs in the LSDB.

- Send a number of CSNPs over the TCP connection. These CNSPs MUST describe the whole LSDB of the sending router. The last CSNP should desribe the last lexographical LSP in the LSDB. The end-id in the CSNP would be FFFF.FFFF.FFFF.FF-FF.

- Process all incoming CSNPs from the remote router. When a CSNP is received, check your own LSDB, and clear the SRM-bits on LSPs that both routers have in common. If the remote router has a version of the LSP that is newer, do not set the SSN-bit. It is not necessary to explicitely request for the newer LSP. The remote router will send it anyway.

- When the last CSNP has been received, walk the LSDB and send any LSPs that still have the SRM-bit set for this interface.

- No retransmission needs to be one by either router. TCP will take care of retransmission.

6.2. Behaviour during the existence of an IS-IS adjacency

The actions that a router has to take when receiving a new LSP are simplified compared to classic flooding.

- When a router receives an LSP, it checks if it has that LSP already in its LSDB. And it checks whether the version of the received LSP is newer or not.

- If the version is the same, the router does nothing.

- If the version of the received LSP is older than the LSP in the LSDB, the router sets the SRM-bit for the LSP. At some point in time, the router will then send its own LSP back to the other router.

- If the version of the recieved LSP is newer than the LSP in the LSDB, the router sets the SRM-bits for this LSP for all interfaces, except the interface it received the newer version of the LSP from.

- The receiving router does not set the SSN-bit and does not send an acknowledgement (PSNP).

- Periodically, or event driven, the router will check its LSDB for LSPs with the SRM-bit set. When it finds such LSPs, it will send as many of those LSPs to neighbors, via TCP. There is no packet-pacing. All flow-control is handled by TCP. After sending one or more LSPs, the router does not set any state to indicate that the LSP needs retransmission. The router does not expect an acknowledgement (PSNP). No retransmission-timer needs to be started. Just sending the LSPs is enough.

7. Considerations regarding IS-IS flooding over TCP

7.1. Flooding over TCP is only done on point-to-point interfaces

Flooding over TCP is not supported for multi-point interfaces. The nature of classic flooding between multiple routers on a LAN is too complex to simply replace by TCP connections. Therefor the new flooding-over-TCP TLV should only be included in point-to-point IIH.

Care must be taken that when a large network consists mostly of point-to-point interfaces, there are no multi-point between routers left in the network. Doing classic flooding over those multi-point interfaces might require substantial more resources than flooding on routers with only point-to-point interfaces.

7.2. Unnumbered interfaces and reachability of the interface ip-address

When a router tries to open a TCP connection to another router, it uses the TCP port-number and an IP-address is has learned from the new flooding-over-TCP TLV. This destination address can be any advertised IP-address that is from a prefix shared between the two routers.

However, it is possible that both routers use "ip unnumbered" on the point-to-point interface. In that case, the remote destination ip-address might not appear in the sender's routing table. Typically routes are installed in the routing table only after doing an SPF computation and learning how to reach all IP-prefixes that are included in LSPs. Typically routers do not install routes in the routing table for IP-addresses learned from the IP-Interface-Addresses TLV in IIHs. When a router is planning to do flooding over TCP, and does not have opened a TCP connection yet, it will not have all the LSPs in its LSDB necessary to learn how to reach the IP-address from the new Flooding-over-TCP TLV, or from the IP-Interface-Addresses TVL.

Therefor it is recommended that when a router does flooding over TCP, and one of its interfaces is configured as "unnumbered", that router SHOULD install host-routes (/32s or /128s) in its routing table, so

that TCP will be able to open a connection to the router on the other end of an adjacency. These host-routes can be interface-routes for the IP-address(es) learned from the new Flooding-over-TCP TLV in the p2p IIHs.

7.3. Multiple levels of hierarchy on one interface

IS-IS flooding over TCP is only defined for point-to-point interfaces. Over point-to-point interfaces, only one type of IIH PDU is sent, even when the interface is used by both level-1 and level-2 routing. This means that IS-IS flooding over TCP is negotiated in only one location (inside the p2p IIH). Two routers use a single TCP-connection, even when doing both level-1 and level-2 routing over that interface.

The packet-types of LSPs and SNPs identify whether the packet is level-1 or level-2. Therefor no confusion can occur when receiving both level-1 and level-2 PDUs over the same TCP connection.

7.4. Downsides of using TCP for IS-IS flooding

When TCP-segments are dropped, TCP will retransmit those segments a little while later. In the mean time, new versions might arrive of the LSPs that are in the TCP buffers. Therefor TCP might retransmit stale LSPs. Which it would not have done if flooding was done via the standard way. This causes only a slight inefficient use of resources. Ultimately the current versions of those LSPs will be transmitted. To protect against this, it is recommended to not make the TCP window-size larger than the default.

7.5. What to do if the TCP connection breaks

If a TCP connection gets closed or reset, the router with the lowest System-ID MUST periodically try to re-open the TCP connection. Both routers MUST NOT declare the adjacency down. An existing adjacency must stay established as long as IIHs are exchanged and the holding-time timer doesn’t expire.

The benefit of this behaviour is that it allows IS-IS implementations a certain flexibility. E.g. when an IS-IS process on a router is restarted, and the TCP connection is re-established, this will not bring down the adjacency. Or a router can switch over to the Hot Standby Control Plane, or do In-Service Software-Upgrades (ISSU) without causing adjacencies to go down.

7.6. What to do if the TCP connection can not be set up

It could happen that two routers can exchange IS-IS PDUs fine, but they can not set up a TCP connection. What needs to be done in this case is open for discussion.

8. Security Considerations

IS-IS as defined in ISO-10589 encapsulates IS-IS PDUs straight into a layer-2 header. One of the benefits of this is that remote attackers can not send IS-IS messages to a targeted router that is several ip-hops away. Using TCP for IS-IS flooding would potentially open up IS-IS routers to these forms of attacks.

The common way for a protocol to protect itself against these remote attack is using the TTL-field in the IP-header of TCP-segments.

When a router send a TCP-segment with IS-IS flooding data, it MUST set the TTL of the IP-header to 255. When a router receives a TCP-segment with IS-IS flooding data, it MUST check to see if the TTL is still 255. If a router receives a TCP-segment with IS-IS flooding data, and the TTL is less than 255, the router MUST ignore and drop the TCP-segment.

Identification and Authentication. When a new TCP-session is established to flood over, each router MUST first send a regular IIH over the TCP-session. This allows each router to verify that the other side of the TCP-connection is who they expect it to be. The IIH has the System-ID and the Interface-ID of the sending router. Regular authentication methods will place an authentication-TLV inside the IIH. Regardless of the fact whether routers flood over layer-2 or flood over TCP, these authentication mechanisms can be used to verify the other side of the TCP-connection. Sending a regular IIH for verification and authentication, instead of having our own new method, guaranteees that Flooding-over-TCP will use new authentication mechanisms when those get developed in the future.

9. Acknowledgements

The authors would like to express thanks to Filip Martin, Dirk Goethals and Koen Leclerq for their suggestions and comments.

10. Contributor Addresses

Below is a list of other contributing authors in alphabetical order:

Figure 3

11. IANA Considerations

This document requests one new TLV code-point, to be used in IIHs

12. References

12.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://xml.resource.org/public/rfc/html/rfc2119.html>.

[2] Lapukhov, P., Premji, A., and J. Mitchell, Ed., "Use of BGP for Routing in Large-Scale Data Centers", RFC 7938, DOI 10.17487/RFC7938, August 2016, <https://www.rfc-editor.org/info/rfc7938>.

12.2. Informative References

[3] International Standard 10589, "Intermediate System to Intermediate System intra- domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode network service (ISO 8473), Second Edition.", 2002.

[4] Patel, K., Lindem, A., and W. Henderickx, "Link State Vector Routing (LSVR)", August 2018.

[5] Przygienda, T., Sharma, A., Thubert, P., Atlas, A., and J. Drake, "Routing in Fat Trees (RIFT)", June 2018.

Authors' Addresses

Henk Smit (editor)

Email: hhw.smit@xs4all.nl

Gunter Van de Velde
Nokia
Copernicuslaan 50
2018 Antwerp
Belgium

Email: gunter.van_de_velde@nokia.com

Networking Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 31, 2020

L. Ginsberg
P. Psenak
Cisco Systems
S. Previdi
Huawei
W. Henderickx
Nokia
J. Drake
Juniper Networks

June 29, 2020

IS-IS Application-Specific Link Attributes
draft-ietf-isis-te-app-19

Abstract

Existing traffic engineering related link attribute advertisements have been defined and are used in RSVP-TE deployments. Since the original RSVP-TE use case was defined, additional applications (e.g., Segment Routing Policy, Loop Free Alternate) that also make use of the link attribute advertisements have been defined . In cases where multiple applications wish to make use of these link attributes, the current advertisements do not support application-specific values for a given attribute, nor do they support indication of which applications are using the advertised value for a given link. This document introduces new link attribute advertisements that address both of these shortcomings.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 December 31, 2020.

Copyright Notice

Copyright (c) 2020 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

Advertisement of link attributes by the Intermediate-System-to-Intermediate-System (IS-IS) protocol in support of traffic engineering (TE) was introduced by [RFC5305] and extended by [RFC5307], [RFC6119], [RFC7308], and [RFC8570]. Use of these extensions has been associated with deployments supporting Traffic Engineering over Multiprotocol Label Switching (MPLS) in the presence of the Resource Reservation Protocol (RSVP) - more succinctly referred to as RSVP-TE [RFC3209].

For the purposes of this document an application is a technology that makes use of link attribute advertisements - examples of which are listed in Section 3.

In recent years new applications that have use cases for many of the link attributes historically used by RSVP-TE have been introduced. Such applications include Segment Routing Policy (SR Policy) [I-D.ietf-spring-segment-routing-policy] and Loop Free Alternates (LFA) [RFC5286]. This has introduced ambiguity in that if a deployment includes a mix of RSVP-TE support and SR Policy support (for example) it is not possible to unambiguously indicate which advertisements are to be used by RSVP-TE and which advertisements are to be used by SR Policy. If the topologies are fully congruent this may not be an issue, but any incongruence leads to ambiguity.

An example where this ambiguity causes a problem is a network where RSVP-TE is enabled only on a subset of its links. A link attribute is advertised for the purpose of another application (e.g. SR Policy) for a link that is not enabled for RSVP-TE. As soon as the router that is an RSVP-TE head-end sees the link attribute being advertised for that link, it assumes RSVP-TE is enabled on that link, even though it is not. If such RSVP-TE head-end router tries to

setup an RSVP-TE path via that link, it will result in a path setup failure.

An additional issue arises in cases where both applications are supported on a link but the link attribute values associated with each application differ. Current advertisements do not support advertising application-specific values for the same attribute on a specific link.

This document defines extensions that address these issues. Also, as evolution of use cases for link attributes can be expected to continue in the years to come, this document defines a solution that is easily extensible to the introduction of new applications and new use cases.

2. Requirements Discussion

As stated previously, evolution of use cases for link attributes can be expected to continue. Therefore, any discussion of existing use cases is limited to requirements that are known at the time of this writing. However, in order to determine the functionality required beyond what already exists in IS-IS, it is only necessary to discuss use cases that justify the key points identified in the introduction, which are:

1. Support for indicating which applications are using the link attribute advertisements on a link

2. Support for advertising application-specific values for the same attribute on a link

[RFC7855] discusses use cases/requirements for Segment Routing (SR). Included among these use cases is SR Policy which is defined in [I-D.ietf-spring-segment-routing-policy]. If both RSVP-TE and SR Policy are deployed in a network, link attribute advertisements can be used by one or both of these applications. As there is no requirement for the link attributes advertised on a given link used by SR Policy to be identical to the link attributes advertised on that same link used by RSVP-TE, there is a clear requirement to indicate independently which link attribute advertisements are to be used by each application.

As the number of applications that may wish to utilize link attributes may grow in the future, an additional requirement is that the extensions defined allow the association of additional applications to link attributes without altering the format of the advertisements or introducing new backwards compatibility issues.

Finally, there may still be many cases where a single attribute value can be shared among multiple applications, so the solution must minimize advertising duplicate link/attribute pairs whenever possible.

3. Legacy Advertisements

There are existing advertisements used in support of RSVP-TE. These advertisements include sub-TLVs for TLVs 22, 23, 25, 141, 222, and 223 and TLVs for Shared Risk Link Group (SRLG) advertisement.

Sub-TLV values are defined in the Sub-TLVs for TLVs 22, 23, 25, 141, 222, and 223 registry.

TLVs are defined in the TLV Codepoints Registry.

3.1. Legacy sub-TLVs

Sub-TLVs for TLVs 22, 23, 25, 141, 222, and 223

Type	Description
3	Administrative group (color)
9	Maximum link bandwidth
10	Maximum reservable link bandwidth
11	Unreserved bandwidth
14	Extended Administrative Group
18	TE Default Metric
33	Unidirectional Link Delay
34	Min/Max Unidirectional Link Delay
35	Unidirectional Delay Variation
36	Unidirectional Link Loss
37	Unidirectional Residual Bandwidth
38	Unidirectional Available Bandwidth
39	Unidirectional Utilized Bandwidth

3.2. Legacy SRLG Advertisements

TLV 138 GMPLS-SRLG
Supports links identified by IPv4 addresses and unnumbered links

TLV 139 IPv6 SRLG
Supports links identified by IPv6 addresses

Note that [RFC6119] prohibits the use of TLV 139 when it is possible to use TLV 138.

4. Advertising Application-Specific Link Attributes

Two new code points are defined in support of Application-Specific Link Attribute (ASLA) Advertisements:

1) ASLA sub-TLV for TLVs 22, 23, 25, 141, 222, and 223 (defined in Section 4.2).

2)Application-Specific Shared Risk Link Group (SRLG) TLV (defined in Section 4.3).

In support of these new advertisements, an application identifier bit mask is defined that identifies the application(s) associated with a given advertisement (defined in Section 4.1).

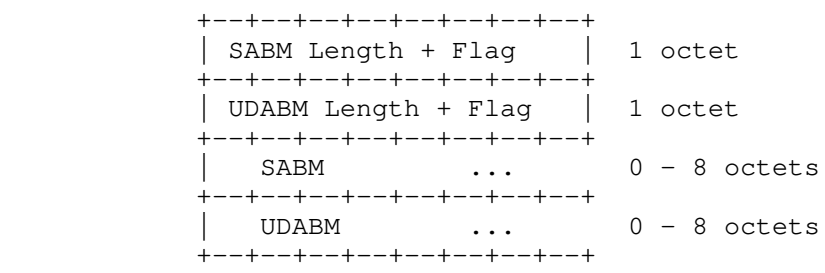
In addition to supporting the advertisement of link attributes used by standardized applications, link attributes can also be advertised for use by user defined applications. Such applications are not subject to standardization and are outside the scope of this document.

The following sections define the format of these new advertisements.

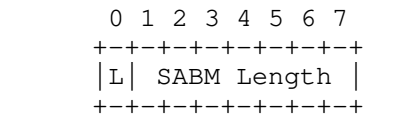
4.1. Application Identifier Bit Mask

Identification of the set of applications associated with link attribute advertisements utilizes two bit masks. One bit mask is for standard applications where the definition of each bit is defined in a new IANA controlled registry. A second bit mask is for non-standard User Defined Applications (UDAs).

The encoding defined below is used by both the Application-Specific Link Attributes sub-TLV and the Application-Specific SRLG TLV.



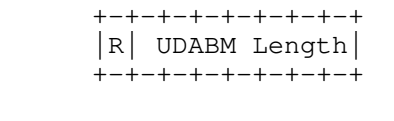
SABM Length + Flag (1 octet)
Standard Application Identifier Bit Mask
Length + Flag



L-flag: Legacy Flag.
See Section 4.2 for a description of how
this flag is used.

SABM Length: Indicates the length in octets (0-8) of the
Standard Application Identifier Bit Mask. The length SHOULD
be the minimum required to send all bits that are set.

UDABM Length + Flag (1 octet)
User Defined Application Identifier Bit Mask
Length + Flag



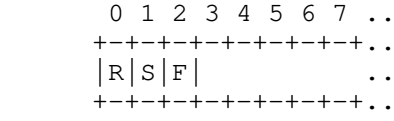
R: Reserved. SHOULD be transmitted as 0 and
MUST be ignored on receipt

UDABM Length: Indicates the length in octets (0-8) of the
User Defined Application Identifier Bit Mask. The length SHOULD
be the minimum required to send all bits that are set.

SABM (variable length)
Standard Application Identifier Bit Mask

(SABM Length * 8) bits

This field is omitted if SABM Length is 0.



R-bit: Set to specify RSVP-TE

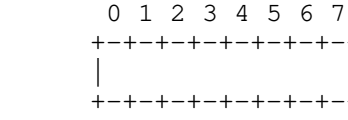
S-bit: Set to specify Segment Routing Policy

F-bit: Set to specify Loop Free Alternate (LFA)

(includes all LFA types)

UDABM (variable length)
User Defined Application Identifier Bit Mask

(UDABM Length * 8) bits



This field is omitted if UDABM Length is 0.

NOTE: SABM/UDABM Length is arbitrarily limited to 8 octets in order to insure that sufficient space is left to advertise link attributes without overrunning the maximum length of a sub-TLV.

Standard Application Identifier Bits are defined/sent starting with Bit 0.

User Defined Application Identifier Bits have no relationship to Standard Application Identifier Bits and are not managed by IANA or any other standards body. It is recommended that bits are used starting with Bit 0 so as to minimize the number of octets required to advertise all UDAs.

In the case of both SABM and UDABM, the following rules apply:

- Undefined bits that are transmitted MUST be transmitted as 0 and MUST be ignored on receipt
- Bits that are not transmitted MUST be treated as if they are set to 0 on receipt.
- Bits that are not supported by an implementation MUST be ignored on receipt.

.

4.2. Application-Specific Link Attributes sub-TLV

A new sub-TLV for TLVs 22, 23, 25, 141, 222, and 223 is defined that supports specification of the applications and application-specific attribute values.

Type: 16 (temporarily assigned by IANA)
Length: Variable (1 octet)
Value:

Application Identifier Bit Mask
(as defined in Section 4.1)

Link Attribute sub-sub-TLVs - format matches the existing formats defined in [RFC5305], [RFC7308], and [RFC8570]

If the SABM or UDABM length in the Application Identifier Bit Mask is greater than 8, the entire sub-TLV MUST be ignored.

When the L-flag is set in the Application Identifier Bit Mask, all of the applications specified in the bit mask MUST use the legacy advertisements for the corresponding link found in TLVs 22, 23, 25, 141, 222, and 223 or TLV 138 or TLV 139 as appropriate. Link attribute sub-sub-TLVs for the corresponding link attributes MUST NOT be advertised for the set of applications specified in the Standard/ User Application Identifier Bit Masks and all such advertisements MUST be ignored on receipt.

Multiple Application-Specific Link Attribute sub-TLVs for the same link MAY be advertised. When multiple sub-TLVs for the same link are advertised, they SHOULD advertise non-conflicting application/ attribute pairs. A conflict exists when the same application is associated with two different values for the same link attribute for a given link. In cases where conflicting values for the same application/attribute/link are advertised the first advertisement received in the lowest numbered LSP SHOULD be used and subsequent advertisements of the same attribute SHOULD be ignored.

For a given application, the setting of the L-flag MUST be the same in all sub-TLVs for a given link. In cases where this constraint is violated, the L-flag MUST be considered set for this application.

If link attributes are advertised associated with zero length Application Identifier Bit Masks for both standard applications and user defined applications, then any Standard Application and/or any User Defined Application is permitted to use that set of link attributes so long as there is not another set of attributes advertised on that same link that is associated with a non-zero length Application Identifier Bit Mask with a matching Application Identifier Bit set.

A new registry of sub-sub-TLVs is to be created by IANA that defines the link attribute sub-sub-TLV code points. This document defines a

sub-sub-TLV for each of the existing sub-TLVs listed in Section 3.1 except as noted below. The format of the sub-sub-TLVs matches the format of the corresponding legacy sub-TLV and IANA is requested to assign the legacy sub-TLV identifier to the corresponding sub-sub-TLV.

4.2.1. Special Considerations for Maximum Link Bandwidth

Maximum link bandwidth is an application independent attribute of the link. When advertised using the Application-Specific Link Attributes sub-TLV, multiple values for the same link MUST NOT be advertised. This can be accomplished most efficiently by having a single advertisement for a given link where the Application Identifier Bit Mask identifies all the applications that are making use of the value for that link.

It is also possible to advertise the same value for a given link multiple times with disjoint sets of applications specified in the Application Identifier Bit Mask. This is less efficient but still valid.

It is also possible to advertise a single advertisement with zero length SABM and UDABM so long as the constraints discussed in Section 4.2 and Section 6.2 are acceptable.

If different values for Maximum Link Bandwidth for a given link are advertised, all values MUST be ignored.

4.2.2. Special Considerations for Reservable/Unreserved Bandwidth

Maximum Reservable Link Bandwidth and Unreserved Bandwidth are attributes specific to RSVP-TE. When advertised using the Application-Specific Link Attributes sub-TLV, bits other than the RSVP-TE (R-bit) MUST NOT be set in the Application Identifier Bit Mask. If an advertisement of Maximum Reservable Link Bandwidth or Unreserved Bandwidth is received with bits other than the RSVP-TE bit set, the advertisement MUST be ignored.

4.2.3. Considerations for Extended TE Metrics

[RFC8570] defines a number of dynamic performance metrics associated with a link. It is conceivable that such metrics could be measured specific to traffic associated with a specific application. Therefore this document includes support for advertising these link attributes specific to a given application. However, in practice it may well be more practical to have these metrics reflect the performance of all traffic on the link regardless of application. In such cases, advertisements for these attributes will be associated

with all of the applications utilizing that link. This can be done either by explicitly specifying the applications in the Application Identifier Bit Mask or by using a zero length Application Identifier Bit Mask.

4.3. Application-Specific SRLG TLV

A new TLV is defined to advertise application-specific SRLGs for a given link. Although similar in functionality to TLV 138 [RFC5307] and TLV 139 [RFC6119], a single TLV provides support for IPv4, IPv6, and unnumbered identifiers for a link. Unlike TLVs 138/139, it utilizes sub-TLVs to encode the link identifiers in order to provide the flexible formatting required to support multiple link identifier types.

```
Type: 238 (Temporarily assigned by IANA)
Length: Number of octets in the value field (1 octet)
Value:
  Neighbor System-ID + pseudo-node ID (7 octets)
  Application Identifier Bit Mask
   (as defined in Section 4.1)
  Length of sub-TLVs (1 octet)
  Link Identifier sub-TLVs (variable)
  0 or more SRLG Values (Each value is 4 octets)
```

The following Link Identifier sub-TLVs are defined. The values chosen are intentionally matching the equivalent sub-TLVs from [RFC5305], [RFC5307], and [RFC6119].

Type	Description
4	Link Local/Remote Identifiers [RFC5307]
6	IPv4 interface address [RFC5305]
8	IPv4 neighbor address [RFC5305]
12	IPv6 Interface Address [RFC6119]
13	IPv6 Neighbor Address [RFC6119]

At least one set of link identifiers (IPv4, IPv6, or Link Local/ Remote) MUST be present. Multiple occurrences of the same identifier type MUST NOT be present. TLVs that do not meet this requirement MUST be ignored.

Multiple TLVs for the same link MAY be advertised.

When the L-flag is set in the Application Identifier Bit Mask, SRLG values MUST NOT be included in the TLV. Any SRLG values that are advertised MUST be ignored. Based on the link identifiers advertised the corresponding legacy TLV (see Section 3.2) can be identified and

the SRLG values advertised in the legacy TLV MUST be used by the set of applications specified in the Application Identifier Bit Mask.

For a given application, the setting of the L-flag MUST be the same in all TLVs for a given link. In cases where this constraint is violated, the L-flag MUST be considered set for this application.

5. Attribute Advertisements and Enablement

This document defines extensions to support the advertisement of application-specific link attributes.

Whether the presence of link attribute advertisements for a given application indicates that the application is enabled on that link depends upon the application. Similarly, whether the absence of link attribute advertisements indicates that the application is not enabled depends upon the application.

In the case of RSVP-TE, the advertisement of application-specific link attributes implies that RSVP is enabled on that link. The absence of RSVP-TE application-specific link attributes in combination with the absence of legacy advertisements implies that RSVP is not enabled on that link.

In the case of SR Policy, advertisement of application-specific link attributes does not indicate enablement of SR Policy on that link. The advertisements are only used to support constraints that may be applied when specifying an explicit path. SR Policy is implicitly enabled on all links that are part of the Segment Routing enabled topology independent of the existence of link attribute advertisements.

In the case of LFA, advertisement of application-specific link attributes does not indicate enablement of LFA on that link. Enablement is controlled by local configuration.

If, in the future, additional standard applications are defined to use this mechanism, the specification defining this use MUST define the relationship between application-specific link attribute advertisements and enablement for that application.

This document allows the advertisement of application-specific link attributes with no application identifiers i.e., both the Standard Application Identifier Bit Mask and the User Defined Application Identifier Bit Mask are not present (See Section 4.1). This supports the use of the link attribute by any application. In the presence of an application where the advertisement of link attribute advertisements is used to infer the enablement of an application on

that link (e.g., RSVP-TE), the absence of the application identifier leaves ambiguous whether that application is enabled on such a link. This needs to be considered when making use of the "any application" encoding.

6. Deployment Considerations

This section discuss deployment considerations associated with the use of application-specific link attribute advertisements.

6.1. Use of Legacy Advertisements

Bit Identifiers for Standard Applications are defined in Section 4.1. All of the identifiers defined in this document are associated with applications that were already deployed in some networks prior to the writing of this document. Therefore, such applications have been deployed using the legacy advertisements. The Standard Applications defined in this document may continue to use legacy advertisements for a given link so long as at least one of the following conditions is true:

- The application is RSVP-TE
- The application is SR Policy or LFA and RSVP-TE is not deployed anywhere in the network
- The application is SR Policy or LFA, RSVP-TE is deployed in the network, and both the set of links on which SR Policy and/or LFA advertisements are required and the attribute values used by SR Policy and/or LFA on all such links is fully congruent with the links and attribute values used by RSVP-TE

Under the conditions defined above, implementations that support the extensions defined in this document have the choice of using legacy advertisements or application-specific advertisements in support of SR Policy and/or LFA. This will require implementations to provide controls specifying which type of advertisements are to be sent/ processed on receive for these applications. Further discussion of the associated issues can be found in Section 6.3.

New applications that future documents define to make use of the advertisements defined in this document MUST NOT make use of legacy advertisements. This simplifies deployment of new applications by eliminating the need to support multiple ways to advertise attributes for the new applications.

6.2. Use of Zero Length Application Identifier Bit Masks

Link attribute advertisements associated with zero length Application Identifier Bit Masks for both standard applications and user defined applications are usable by any application, subject to the restrictions specified in Section 4.2. If support for a new application is introduced on any node in a network in the presence of such advertisements, these advertisements are permitted to be used by the new application. If this is not what is intended, then existing advertisements MUST be readvertised with an explicit set of applications specified before a new application is introduced.

6.3. Interoperability, Backwards Compatibility and Migration Concerns

Existing deployments of RSVP-TE, SR Policy, and/or LFA utilize the legacy advertisements listed in Section 3. Routers that do not support the extensions defined in this document will only process legacy advertisements and are likely to infer that RSVP-TE is enabled on the links for which legacy advertisements exist. It is expected that deployments using the legacy advertisements will persist for a significant period of time. Therefore deployments using the extensions defined in this document in the presence of routers that do not support these extensions need to be able to interoperate with the use of legacy advertisements by the legacy routers. The following sub-sections discuss interoperability and backwards compatibility concerns for a number of deployment scenarios.

6.3.1. Multiple Applications: Common Attributes with RSVP-TE

In cases where multiple applications are utilizing a given link, one of the applications is RSVP-TE, and all link attributes for a given link are common to the set of applications utilizing that link, interoperability is achieved by using legacy advertisements and sending application-specific advertisements with L-flag set and no link attribute values. This avoids duplication of link attribute advertisements.

6.3.2. Multiple Applications: All Attributes Not Shared with RSVP-TE

In cases where one or more applications other than RSVP-TE are utilizing a given link and one or more link attribute values are not shared with RSVP-TE, it is necessary to use application-specific advertisements as defined in this document. Attributes for applications other than RSVP-TE MUST be advertised using application-specific advertisements that have the L-flag clear. In cases where some link attributes are shared with RSVP-TE, this requires duplicate advertisements for those attributes.

These guidelines apply to cases where RSVP-TE is not using any advertised attributes on a link and to cases where RSVP-TE is using some link attribute advertisements on the link but some link attributes cannot be shared with RSVP-TE.

6.3.3. Interoperability with Legacy Routers

For the applications defined in this document, routers that do not support the extensions defined in this document will send and receive only legacy link attribute advertisements. So long as there is any legacy router in the network that has any of the applications enabled, all routers MUST continue to advertise link attributes using legacy advertisements. In addition, the link attribute values associated with the set of applications supported by legacy routers (RSVP-TE, SR Policy, and/or LFA) are always shared since legacy routers have no way of advertising or processing application-specific values. Once all legacy routers have been upgraded, migration from legacy advertisements to ASLA advertisements can be achieved via the following steps:

1) Send ASLA advertisements while continuing to advertise using legacy (all advertisements are then duplicated). Receiving routers continue to use legacy advertisements.

2) Enable the use of the ASLA advertisements on all routers

3) Remove legacy advertisements

When the migration is complete, it then becomes possible to advertise incongruent values per application on a given link.

Note that the use of the L-flag is of no value in the migration.

Documents defining new applications that make use of the application-specific advertisements defined in this document MUST discuss interoperability and backwards compatibility issues that could occur in the presence of routers that do not support the new application.

6.3.4. Use of Application-Specific Advertisements for RSVP-TE

The extensions defined in this document support RSVP-TE as one of the supported applications. This allows that RSVP-TE could eventually utilize the application-specific advertisements. This can be done in the following step-wise manner:

1) Upgrade all routers to support the extensions in this document

2)Advertise all legacy link attributes using ASLA advertisements with L-flag clear and R-bit set. At this point both legacy and application-specific advertisements are being sent.

3)Remove legacy advertisements

7. IANA Considerations

This section lists the protocol code point changes introduced by this document and the related IANA changes required.

For new registries defined under IS-IS TLV Codepoints Registry with registration procedure "Expert Review", guidance for designated experts can be found in [RFC7370].

7.1. Application-Specific Link Attributes sub-TLV

This document defines a new sub-TLV in the Sub-TLVs for TLVs 22, 23, 25, 141, 222, and 223 registry. See Section 4.2

Type	Description	22	23	25	141	222	223
16	Application-Specific Link Attributes	y	y	y(s)	y	y	y

7.2. Application-Specific SRLG TLV

This document defines one new TLV in the IS-IS TLV Codepoints Registry. See Section 4.3

Type	Description	IIH	LSP	SNP	Purge
238	Application-Specific SRLG	n	y	n	n

7.3. Application-Specific Link Attributes sub-sub-TLV Registry

This document requests a new IANA registry under the IS-IS TLV Codepoints Registry be created to control the assignment of sub-sub-TLV codepoints for the Application-Specific Link Attributes sub-TLV defined in Section 7.1. The suggested name of the new registry is "sub-sub-TLV code points for application-specific link attributes". The registration procedure is "Expert Review" as defined in [RFC8126]. The following assignments are made by this document:

Type	Description	Encoding Reference
0-2	Unassigned	
3	Administrative group (color)	RFC5305
4-8	Unassigned	
9	Maximum link bandwidth	RFC5305
10	Maximum reservable link bandwidth	RFC5305
11	Unreserved bandwidth	RFC5305
12-13	Unassigned	
14	Extended Administrative Group	RFC7308
15-17	Unassigned	
18	TE Default Metric	RFC5305
19-32	Unassigned	
33	Unidirectional Link Delay	RFC8570
34	Min/Max Unidirectional Link Delay	RFC8570
35	Unidirectional Delay Variation	RFC8570
36	Unidirectional Link Loss	RFC8570
37	Unidirectional Residual Bandwidth	RFC8570
38	Unidirectional Available Bandwidth	RFC8570
39	Unidirectional Utilized Bandwidth	RFC8570
40-255	Unassigned	

Note to IANA: For future codepoints, in cases where the document that defines the encoding is different from the document that assigns the codepoint, the encoding reference MUST be to the document that defines the encoding.

Note to designated experts: If a link attribute can be advertised both as a sub-TLV of TLVs 22, 23, 25, 141, 222, and 223 and as a sub-sub-TLV of the Application-Specific Link Attributes sub-TLV defined in this document, then the same numerical code should be assigned to the link attribute whenever possible.

7.4. Link Attribute Application Identifier Registry

This document requests a new IANA registry be created, under the category of "Interior Gateway Protocol (IGP) Parameters", to control the assignment of Application Identifier Bits. The suggested name of the new registry is "Link Attribute Applications". The registration policy for this registry is "Expert Review" [RFC8126]. Bit definitions SHOULD be assigned such that all bits in the lowest available octet are allocated before assigning bits in the next octet. This minimizes the number of octets that will need to be transmitted. The following assignments are made by this document:

Bit #	Name
0	RSVP-TE (R-bit)
1	Segment Routing Policy (S-bit)
2	Loop Free Alternate (F-bit)
3-63	Unassigned

7.5. SRLG sub-TLVs

This document requests a new IANA registry be created under the IS-IS TLV Codepoints Registry to control the assignment of sub-TLV types for the application-specific SRLG TLV. The suggested name of the new registry is "Sub-TLVs for TLV 238". The registration procedure is "Expert Review" as defined in [RFC8126]. The following assignments are made by this document:

Value	Description	Encoding Reference
0-3	Unassigned	
4	Link Local/Remote Identifiers	[RFC5307]
5	Unassigned	
6	IPv4 interface address	[RFC5305]
7	Unassigned	
8	IPv4 neighbor address	[RFC5305]
9-11	Unassigned	
12	IPv6 Interface Address	[RFC6119]
13	IPv6 Neighbor Address	[RFC6119]
14-255	Unassigned	

Note to IANA: For future codepoints, in cases where the document that defines the encoding is different from the document that assigns the codepoint, the encoding reference MUST be to the document that defines the encoding.

8. Security Considerations

Security concerns for IS-IS are addressed in [ISO10589], [RFC5304], and [RFC5310]. While IS-IS is deployed under a single administrative domain, there can be deployments where potential attackers have access to one or more networks in the IS-IS routing domain. In these deployments, the stronger authentication mechanisms defined in the aforementioned documents SHOULD be used.

This document defines a new way to advertise link attributes. Tampering with the information defined in this document may have an effect on applications using it, including impacting Traffic Engineering as discussed in [RFC8570]. As the advertisements defined

in this document limit the scope to specific applications, the impact of tampering is similarly limited in scope.

9. Acknowledgements

The authors would like to thank Eric Rosen and Acee Lindem for their careful review and content suggestions.

10. References

10.1. Normative References

[ISO10589] International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routeing 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, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <https://www.rfc-editor.org/info/rfc5307>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC6119] Harrison, J., Berger, J., and M. Bartlett, "IPv6 Traffic Engineering in IS-IS", RFC 6119, DOI 10.17487/RFC6119, February 2011, <https://www.rfc-editor.org/info/rfc6119>.

[RFC7308] Osborne, E., "Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)", RFC 7308, DOI 10.17487/RFC7308, July 2014, <https://www.rfc-editor.org/info/rfc7308>.

[RFC7370] Ginsberg, L., "Updates to the IS-IS TLV Codepoints Registry", RFC 7370, DOI 10.17487/RFC7370, September 2014, <https://www.rfc-editor.org/info/rfc7370>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8570] Ginsberg, L., Ed., Previdi, S., Ed., Giacalone, S., Ward, D., Drake, J., and Q. Wu, "IS-IS Traffic Engineering (TE) Metric Extensions", RFC 8570, DOI 10.17487/RFC8570, March 2019, <https://www.rfc-editor.org/info/rfc8570>.

10.2. Informative References

[I-D.ietf-spring-segment-routing-policy] Filsfils, C., Sivabalan, S., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-07 (work in progress), May 2020.

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

[RFC5286] Atlas, A., Ed. and A. Zinin, Ed., "Basic Specification for IP Fast Reroute: Loop-Free Alternates", RFC 5286, DOI 10.17487/RFC5286, September 2008, <https://www.rfc-editor.org/info/rfc5286>.

[RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <https://www.rfc-editor.org/info/rfc7855>.

Authors' Addresses

Les Ginsberg
Cisco Systems
821 Alder Drive
Milpitas, CA 95035
USA

Email: ginsberg@cisco.com

Peter Psenak
Cisco Systems
Apollo Business Center Mlynske nivy 43
Bratislava 821 09
Slovakia

Email: ppsenak@cisco.com

Stefano Previdi
Huawei

Email: stefano@previdi.net

Wim Henderickx
Nokia
Copernicuslaan 50
Antwerp 2018 94089
Belgium

Email: wim.henderickx@nokia.com

John Drake
Juniper Networks

Email: jdrake@juniper.net

IS-IS for IP Internets
Internet-Draft
Obsoletes: 5306 (if approved)
Intended status: Standards Track
Expires: March 22, 2020

L. Ginsberg
P. Wells
Cisco Systems, Inc.

September 19, 2019

Restart Signaling for IS-IS
draft-ietf-lsr-isis-rfc5306bis-09

Abstract

This document describes a mechanism for a restarting router to signal to its neighbors that it is restarting, allowing them to reestablish their adjacencies without cycling through the down state, while still correctly initiating database synchronization.

This document additionally describes a mechanism for a router to signal its neighbors that it is preparing to initiate a restart while maintaining forwarding plane state. This allows the neighbors to maintain their adjacencies until the router has restarted, but also allows the neighbors to bring the adjacencies down in the event of other topology changes.

This document additionally describes a mechanism for a restarting router to determine when it has achieved Link State Protocol Data Unit (LSP) database synchronization with its neighbors and a mechanism to optimize LSP database synchronization, while minimizing transient routing disruption when a router starts.

This document obsoletes RFC 5306.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 March 22, 2020.

Copyright Notice

Copyright (c) 2019 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. Overview

The Intermediate System to Intermediate System (IS-IS) routing protocol [RFC1195] [ISO10589] is a link state intra-domain routing protocol. Normally, when an IS-IS router is restarted, temporary disruption of routing occurs due to events in both the restarting router and the neighbors of the restarting router.

The router that has been restarted computes its own routes before achieving database synchronization with its neighbors. The results of this computation are likely to be non-convergent with the routes computed by other routers in the area/domain.

Neighbors of the restarting router detect the restart event and cycle their adjacencies with the restarting router through the down state. The cycling of the adjacency state causes the neighbors to regenerate their LSPs describing the adjacency concerned. This in turn causes a temporary disruption of routes passing through the restarting router.

In certain scenarios, the temporary disruption of the routes is highly undesirable. This document describes mechanisms to avoid or minimize the disruption due to both of these causes.

When an adjacency is reinitialized as a result of a neighbor restarting, a router does three things:

1. It causes its own LSP(s) to be regenerated, thus triggering SPF runs throughout the area (or in the case of Level 2, throughout the domain).

2. It sets SRMflags on its own LSP database on the adjacency concerned.

3. In the case of a Point-to-Point link, it transmits a complete set of Complete Sequence Number PDUs (CSNPs), over the adjacency.

In the case of a restarting router process, the first of these is highly undesirable, but the second is essential in order to ensure synchronization of the LSP database.

The third action above minimizes the number of LSPs that must be exchanged and, if made reliable, provides a means of determining when the LSP databases of the neighboring routers have been synchronized.

This is desirable whether or not the router is being restarted (so that the overload bit can be cleared in the router's own LSP, for example).

This document describes a mechanism for a restarting router to signal to its neighbors that it is restarting. The mechanism further allows the neighbors to reestablish their adjacencies with the restarting router without cycling through the down state, while still correctly initiating database synchronization.

This document additionally describes a mechanism for a restarting router to determine when it has achieved LSP database synchronization with its neighbors and a mechanism to optimize LSP database synchronization and minimize transient routing disruption when a router starts.

It is assumed that the three-way handshake [RFC5303] is being used on Point-to-Point circuits.

2. Conventions Used in This Document

If the control and forwarding functions in a router can be maintained independently, it is possible for the forwarding function state to be maintained across a resumption of control function operations. This functionality is assumed when the terms "restart/restarting" are used in this document.

The terms "start/starting" are used to refer to a router in which the control function has either commenced operations for the first time or has resumed operations, but the forwarding functions have not been maintained in a prior state.

The terms "(re)start/(re)starting" are used when the text is applicable to both a "starting" and a "restarting" router.

The terms "normal IIH" or "IIH normal" refer to IS-IS Hellos (IIHs) in which the Restart TLV (defined later in this document) has no flags set.

3. Approach

3.1. Timers

Three additional timers, T1, T2, and T3, are required to support the mechanisms defined in this document. Timers T1 and T2 are used both by a restarting router and a starting router. Timer T3 is used only by a restarting router.

NOTE: These timers are NOT applicable to a router which is preparing to do a planned restart.

An instance of the timer T1 is maintained per interface, and indicates the time after which an unacknowledged (re)start attempt will be repeated. A typical value is 3 seconds.

An instance of the timer T2 is maintained for each LSP database (LSPDB) present in the system. For example, for a Level 1/2 system, there will be an instance of the timer T2 for Level 1 and an instance for Level 2. This is the maximum time that the system will wait for LSPDB synchronization. A typical value is 60 seconds.

A single instance of the timer T3 is maintained for the entire system. It indicates the time after which the router will declare that it has failed to achieve database synchronization (by setting the overload bit in its own LSP). This is initialized to 65535 seconds, but is set to the minimum of the remaining times of received IIHs containing a restart TLV with the Restart Acknowledgement (RA) set and an indication that the neighbor has an adjacency in the "UP" state to the restarting router. (See Section 3.2.1a.)

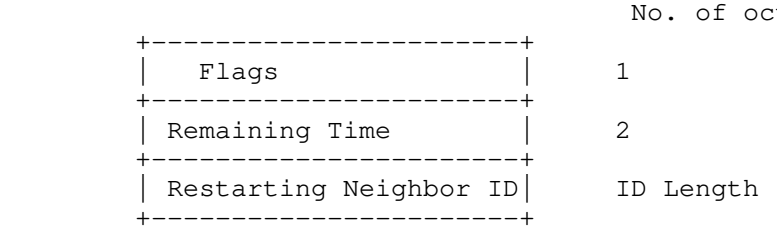
3.2. Restart TLV

A new TLV is defined to be included in IIH PDUs. The TLV includes flags that are used to convey information during a (re)start. The absence of this TLV indicates that the sender supports none of the functionality defined in this document. Therefore, if a router supports any of the functionality defined in this document it MUST include this TLV in all transmitted IIHs.

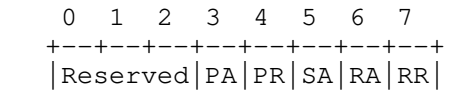
Type 211

Length: Number of octets in the Value field (1 to (3 + ID Length))

Value



Flags (1 octet)



RR - Restart Request
RA - Restart Acknowledgement
SA - Suppress adjacency advertisement
PR - Restart is planned
PA - Planned restart acknowledgement

Remaining Time (2 octets)

Remaining holding time (in seconds).

Required when the RA, PR, or PA bit is set. Otherwise this field SHOULD be omitted when sent and MUST be ignored when received.

Restarting Neighbor System ID (ID Length octets)

The System ID of the neighbor to which an RA/PA refers.

Required when the RA or PA bit is set. Otherwise this field SHOULD be omitted when sent and MUST be ignored when received.

Note: Very early draft versions of the restart functionality did not include the Restarting Neighbor System ID in the TLV. RFC 5306 allowed for the possibility of interoperating with legacy implementations by stating that a router that is expecting an RA on a LAN circuit should assume that the acknowledgement is directed at the local system if the TLV is received with RA set and Restarting Neighbor System ID is not present. It is an implementation choice whether to continue to accept (on a LAN) a TLV with RA set and Restarting Neighbor System ID absent. Note that the omission of the Restarting Neighbor System ID only introduces ambiguity in the case where there are multiple systems on a LAN simultaneously performing restart.

The RR and SA flags may both be set in the TLV under the conditions described in Section 3.3.2. All other combinations where multiple flags are set are invalid and MUST NOT be transmitted. Received TLVs which have invalid flag combinations set MUST be ignored.

3.2.1. Use of RR and RA Bits

The RR bit is used by a (re)starting router to signal to its neighbors that a (re)start is in progress, that an existing adjacency SHOULD be maintained even under circumstances when the normal operation of the adjacency state machine would require the adjacency to be reinitialized, to request a set of CSNPs, and to request setting of the SRMflags.

The RA bit is sent by the neighbor of a (re)starting router to acknowledge the receipt of a restart TLV with the RR bit set.

When the neighbor of a (re)starting router receives an IIH with the restart TLV having the RR bit set, if there exists on this interface an adjacency in state "UP" with the same System ID, and in the case of a LAN circuit, with the same source LAN address, then, irrespective of the other contents of the "Intermediate System Neighbors" option (LAN circuits) or the "Point-to-Point Three-Way Adjacency" option (Point-to-Point circuits):

a. the state of the adjacency is not changed. If this is the first IIH with the RR bit set that this system has received associated with this adjacency, then the adjacency is marked as being in "Restart mode" and the adjacency holding time is refreshed -- otherwise, the holding time is not refreshed. The "remaining time" transmitted according to (b) below MUST reflect the actual time after which the adjacency will now expire. Receipt of an IIH with the RR bit reset will clear the "Restart mode" state. This procedure allows the restarting router to cause the neighbor to maintain the adjacency long enough for restart to successfully complete, while also preventing repetitive restarts from maintaining an adjacency indefinitely. Whether or not an adjacency is marked as being in "Restart mode" has no effect on adjacency state transitions.

b. immediately (i.e., without waiting for any currently running timer interval to expire, but with a small random delay of a few tens of milliseconds on LANs to avoid "storms") transmit over the corresponding interface an IIH including the restart TLV with the RR bit clear and the RA bit set, in the case of Point-to-Point adjacencies having updated the "Point-to-Point Three-Way Adjacency" option to reflect any new values received from the (re)starting router. (This allows a restarting router to quickly acquire the correct information to place in its hellos.) The "Remaining Time" MUST be set to the current time (in seconds) before the holding timer on this adjacency is due to expire. If the corresponding interface is a LAN interface, then the Restarting Neighbor System ID SHOULD be set to the System ID of

the router from which the IIH with the RR bit set was received. This is required to correctly associate the acknowledgement and holding time in the case where multiple systems on a LAN restart at approximately the same time. This IIH SHOULD be transmitted before any LSPs or SNPs are transmitted as a result of the receipt of the original IIH.

c. if the corresponding interface is a Point-to-Point interface, or if the receiving router has the highest LnRouterPriority (with the highest source MAC (Media Access Control) address breaking ties) among those routers to which the receiving router has an adjacency in state "UP" on this interface whose IIHs contain the restart TLV, excluding adjacencies to all routers which are considered in "Restart mode" (note the actual DIS is NOT changed by this process), initiate the transmission over the corresponding interface of a complete set of CSNPs, and set SRMflags on the corresponding interface for all LSPs in the local LSP database.

Otherwise (i.e., if there was no adjacency in the "UP" state to the System ID in question), process the IIH as normal by reinitializing the adjacency and setting the RA bit in the returned IIH.

3.2.2. Use of the SA Bit

The SA bit is used by a starting router to request that its neighbor suppress advertisement of the adjacency to the starting router in the neighbor's LSPs.

A router that is starting has no maintained forwarding function state. This may or may not be the first time the router has started. If this is not the first time the router has started, copies of LSPs generated by this router in its previous incarnation may exist in the LSP databases of other routers in the network. These copies are likely to appear "newer" than LSPs initially generated by the starting router due to the reinitialization of LSP fragment sequence numbers by the starting router. This may cause temporary blackholes to occur until the normal operation of the update process causes the starting router to regenerate and flood copies of its own LSPs with higher sequence numbers. The temporary blackholes can be avoided if the starting router's neighbors suppress advertising an adjacency to the starting router until the starting router has been able to propagate newer versions of LSPs generated by previous incarnations.

When a router receives an IIH with the restart TLV having the SA bit set, if there exists on this interface an adjacency in state "UP" with the same System ID, and in the case of a LAN circuit, with the same source LAN address, then the router MUST suppress advertisement

of the adjacency to the neighbor in its own LSPs. Until an IIH with the SA bit clear has been received, the neighbor advertisement MUST continue to be suppressed. If the adjacency transitions to the "UP" state, the new adjacency MUST NOT be advertised until an IIH with the SA bit clear has been received.

Note that a router that suppresses advertisement of an adjacency MUST NOT use this adjacency when performing its SPF calculation. In particular, if an implementation follows the example guidelines presented in [ISO10589], Annex C.2.5, Step 0:b) "pre-load TENT with the local adjacency database", the suppressed adjacency MUST NOT be loaded into TENT.

3.2.3. Use of PR and PA Bits

The PR bit is used by a router which is planning to initiate a restart to signal to its neighbors that it will be restarting. The router sending an IIH with PR bit set SHOULD set the "remaining time" to a value greater than the expected control plane restart time. The PR bit SHOULD remain set in IIHs until the restart is initiated.

The PA bit is sent by the neighbor of a router planning to restart to acknowledge receipt of a restart TLV with the PR bit set.

When the neighbor of a router planning a restart receives an IIH with the restart TLV having the PR bit set, if there exists on this interface an adjacency in state "UP" with the same System ID, and in the case of a LAN circuit, with the same source LAN address, then:

a. if this is the first IIH with the PR bit set that this system has received associated with this adjacency, then the adjacency is marked as being in "Planned Restart state" and the adjacency holding time is refreshed -- otherwise, the holding time is not refreshed. The holding time SHOULD be set to the "remaining time" specified in the received IIH with PR set. The "remaining time" transmitted according to (b) below MUST reflect the actual time after which the adjacency will now expire. Receipt of an IIH with the PR bit reset will clear the "Planned Restart state" and cause the receiving router to set the adjacency hold time to the locally configured value. This procedure allows the router planning a restart to cause the neighbor to maintain the adjacency long enough for restart to successfully complete. Whether or not an adjacency is marked as being in "Planned Restart state" has no effect on adjacency state transitions.

b. immediately (i.e., without waiting for any currently running timer interval to expire, but with a small random delay of a few tens of milliseconds on LANs to avoid "storms") transmit over the

corresponding interface an IIH including the restart TLV with the PR bit clear and the PA bit set. The "Remaining Time" MUST be set to the current time (in seconds) before the holding timer on this adjacency is due to expire. If the corresponding interface is a LAN interface, then the Restarting Neighbor System ID SHOULD be set to the System ID of the router from which the IIH with the PR bit set was received. This is required to correctly associate the acknowledgement and holding time in the case where multiple systems on a LAN are planning a restart at approximately the same time.

NOTE: Receipt of an IIH with PA bit set indicates to the router planning a restart that the neighbor is aware of the planned restart and - in the absence of topology changes as described below - will maintain the adjacency for the "remaining time" included in the IIH with PA set.

By definition, a restarting router maintains forwarding state across the control plane restart (see Section 2). But while a control plane restart is in progress it is expected that the restarting router will be unable to respond to topology changes. It is therefore useful to signal a planned restart so that the neighbors of the restarting router can determine whether it is safe to maintain the adjacency if other topology changes occur prior to the completion of the restart. Signalling a planned restart in the absence of maintained forwarding plane state is likely to lead to significant traffic loss and MUST NOT be done.

Neighbors of the router which has signaled planned restart SHOULD maintain the adjacency in a planned restart state until it receives an IIH with the RR bit set, receives an IIH with both PR and RR bits clear, or the adjacency holding time expires - whichever occurs first. Neighbors which choose not to follow the recommended behavior need to consider the impact on traffic delivery of not using the restarting router for forwarding traffic during the restart period.

While the adjacency is in planned restart state some or all of the following actions MAY be taken:

a. if additional topology changes occur, the adjacency which is in planned restart state MAY be brought down even though the hold time has not yet expired. Given that the neighbor which has signaled a planned restart is not expected to update its forwarding plane in response to signalling of the topology changes (since it is restarting) traffic which transits that node is at risk of being improperly forwarded. On a LAN circuit, if the router in planned restart state is the DIS at any supported level, the adjacency(ies) SHOULD be brought down whenever any LSP

update is either generated or received, so as to trigger a new DIS election. Failure to do so will compromise the reliability of the Update Process on that circuit. What other criteria are used to determine what topology changes will trigger bringing the adjacency down is a local implementation decision.

b. if a BFD [RFC5880] session to the neighbor which signals a planned restart is in the UP state and subsequently goes DOWN, the event MAY be ignored since it is possible this is an expected side effect of the restart. Use of the Control Plane Independent state as signalled in BFD control packets SHOULD be considered in the decision to ignore a BFD Session DOWN event.

c. on a Point-to-Point circuit, transmission of LSPs, CSNPs, and PSNPs MAY be suppressed. It is expected that the PDUs will not be received.

Use of the PR bit provides a means to safely support restart periods which are significantly longer than standard holdtimes.

3.3. Adjacency (Re)Acquisition

Adjacency (re)acquisition is the first step in (re)initialization. Restarting and starting routers will make use of the RR bit in the restart TLV, though each will use it at different stages of the (re)start procedure.

3.3.1. Adjacency Reacquisition during Restart

The restarting router explicitly notifies its neighbor that the adjacency is being reacquired, and hence that it SHOULD NOT reinitialize the adjacency. This is achieved by setting the RR bit in the restart TLV. When the neighbor of a restarting router receives an IIH with the restart TLV having the RR bit set, if there exists on this interface an adjacency in state "UP" with the same System ID, and in the case of a LAN circuit, with the same source LAN address, then the procedures described in Section 3.2.1 are followed.

A router that does not support the restart capability will ignore the restart TLV and reinitialize the adjacency as normal, returning an IIH without the restart TLV.

On restarting, a router initializes the timer T3, starts the timer T2 for each LSPDB, and for each interface (and in the case of a LAN circuit, for each level) starts the timer T1 and transmits an IIH containing the restart TLV with the RR bit set.

On a Point-to-Point circuit, the restarting router SHOULD set the "Adjacency Three-Way State" to "Init", because the receipt of the acknowledging IIH (with RA set) MUST cause the adjacency to enter the "UP" state immediately.

On a LAN circuit, the LAN-ID assigned to the circuit SHOULD be the same as that used prior to the restart. In particular, for any circuits for which the restarting router was previously DIS, the use of a different LAN-ID would necessitate the generation of a new set of pseudonode LSPs, and corresponding changes in all the LSPs referencing them from other routers on the LAN. By preserving the LAN-ID across the restart, this churn can be prevented. To enable a restarting router to learn the LAN-ID used prior to restart, the LAN-ID specified in an IIH with RR set MUST be ignored.

Transmission of "normal IIHs" is inhibited until the conditions described below are met (in order to avoid causing an unnecessary adjacency initialization). Upon expiry of the timer T1, it is restarted and the IIH is retransmitted as above.

When a restarting router receives an IIH a local adjacency is established as usual, and if the IIH contains a restart TLV with the RA bit set (and on LAN circuits with a Restart Neighbor System ID that matches that of the local system), the receipt of the acknowledgement over that interface is noted. When the RA bit is set and the state of the remote adjacency is "UP", then the timer T3 is set to the minimum of its current value and the value of the "Remaining Time" field in the received IIH.

On a Point-to-Point link, receipt of an IIH not containing the restart TLV is also treated as an acknowledgement, since it indicates that the neighbor is not restart capable. However, since no CSNP is guaranteed to be received over this interface, the timer T1 is cancelled immediately without waiting for a complete set of CSNPs. Synchronization may therefore be deemed complete even though there are some LSPs which are held (only) by this neighbor (see Section 3.4). In this case, we also want to be certain that the neighbor will reinitialize the adjacency in order to guarantee that the SRMflags have been set on its database, thus ensuring eventual LSPDB synchronization. This is guaranteed to happen except in the case where the Adjacency Three-Way State in the received IIH is "UP" and the Neighbor Extended Local Circuit ID matches the extended local circuit ID assigned by the restarting router. In this case, the restarting router MUST force the adjacency to reinitialize by setting the local Adjacency Three-Way State to "DOWN" and sending a normal IIH.

In the case of a LAN interface, receipt of an IIH not containing the restart TLV is unremarkable since synchronization can still occur so long as at least one of the non-restarting neighboring routers on the LAN supports restart. Therefore, T1 continues to run in this case. If none of the neighbors on the LAN are restart capable, T1 will eventually expire after the locally defined number of retries.

In the case of a Point-to-Point circuit, the "LocalCircuitID" and "Extended Local Circuit ID" information contained in the IIH can be used immediately to generate an IIH containing the correct three-way handshake information. The presence of "Neighbor Extended Local Circuit ID" information that does not match the value currently in use by the local system is ignored (since the IIH may have been transmitted before the neighbor had received the new value from the restarting router), but the adjacency remains in the initializing state until the correct information is received.

In the case of a LAN circuit, the source neighbor information (e.g., SNPAAddress) is recorded and used for adjacency establishment and maintenance as normal.

When BOTH a complete set of CSNPs (for each active level, in the case of a Point-to-Point circuit) and an acknowledgement have been received over the interface, the timer T1 is cancelled.

Once the timer T1 has been cancelled, subsequent IIHs are transmitted according to the normal algorithms, but including the restart TLV with both RR and RA clear.

If a LAN contains a mixture of systems, only some of which support the new algorithm, database synchronization is still guaranteed, but the "old" systems will have reinitialized their adjacencies.

If an interface is active, but does not have any neighboring router reachable over that interface, the timer T1 would never be cancelled, and according to Section 3.4.1.1, the SPF would never be run. Therefore, timer T1 is cancelled after some predetermined number of expirations (which MAY be 1).

3.3.2. Adjacency Acquisition during Start

The starting router wants to ensure that in the event that a neighboring router has an adjacency to the starting router in the "UP" state (from a previous incarnation of the starting router), this adjacency is reinitialized. The starting router also wants neighboring routers to suppress advertisement of an adjacency to the starting router until LSP database synchronization is achieved. This is achieved by sending IIHs with the RR bit clear and the SA bit set

in the restart TLV. The RR bit remains clear and the SA bit remains set in subsequent transmissions of IIHs until the adjacency has reached the "UP" state and the initial T1 timer interval (see below) has expired.

Receipt of an IIH with the RR bit clear will result in the neighboring router utilizing normal operation of the adjacency state machine. This will ensure that any old adjacency on the neighboring router will be reinitialized.

Upon receipt of an IIH with the SA bit set, the behavior described in Section 3.2.2 is followed.

Upon starting, a router starts timer T2 for each LSPDB.

For each interface (and in the case of a LAN circuit, for each level), when an adjacency reaches the "UP" state, the starting router starts a timer T1 and transmits an IIH containing the restart TLV with the RR bit clear and SA bit set. Upon expiry of the timer T1, it is restarted and the IIH is retransmitted with both RR and SA bits set (only the RR bit has changed state from earlier IIHs).

Upon receipt of an IIH with the RR bit set (regardless of whether or not the SA bit is set), the behavior described in Section 3.2.1 is followed.

When an IIH is received by the starting router and the IIH contains a restart TLV with the RA bit set (and on LAN circuits with a Restart Neighbor System ID that matches that of the local system), the receipt of the acknowledgement over that interface is noted.

On a Point-to-Point link, receipt of an IIH not containing the restart TLV is also treated as an acknowledgement, since it indicates that the neighbor is not restart capable. Since the neighbor will have reinitialized the adjacency, this guarantees that SRMflags have been set on its database, thus ensuring eventual LSPDB synchronization. However, since no CSNP is guaranteed to be received over this interface, the timer T1 is cancelled immediately without waiting for a complete set of CSNPs. Synchronization may therefore be deemed complete even though there are some LSPs that are held (only) by this neighbor (see Section 3.4).

In the case of a LAN interface, receipt of an IIH not containing the restart TLV is unremarkable since synchronization can still occur so long as at least one of the non-restarting neighboring routers on the LAN supports restart. Therefore, T1 continues to run in this case. If none of the neighbors on the LAN are restart capable, T1 will eventually expire after the locally defined number of retries. The

usual operation of the update process will ensure that synchronization is eventually achieved.

When BOTH a complete set of CSNPs (for each active level, in the case of a Point-to-Point circuit) and an acknowledgement have been received over the interface, the timer T1 is cancelled. Subsequent IIHs sent by the starting router have the RR and RA bits clear and the SA bit set in the restart TLV.

Timer T1 is cancelled after some predetermined number of expirations (which MAY be 1).

When the T2 timer(s) are cancelled or expire, transmission of "normal IIHs" will begin.

3.3.3. Multiple Levels

A router that is operating as both a Level 1 and a Level 2 router on a particular interface MUST perform the above operations for each level.

On a LAN interface, it MUST send and receive both Level 1 and Level 2 IIHs and perform the CSNP synchronizations independently for each level.

On a Point-to-Point interface, only a single IIH (indicating support for both levels) is required, but it MUST perform the CSNP synchronizations independently for each level.

3.4. Database Synchronization

When a router is started or restarted, it can expect to receive a complete set of CSNPs over each interface. The arrival of the CSNP(s) is now guaranteed, since an IIH with the RR bit set will be retransmitted until the CSNP(s) are correctly received.

The CSNPs describe the set of LSPs that are currently held by each neighbor. Synchronization will be complete when all these LSPs have been received.

When (re)starting, a router starts an instance of timer T2 for each LSPDB as described in Section 3.3.1 or Section 3.3.2. In addition to normal processing of the CSNPs, the set of LSPIDs contained in the first complete set of CSNPs received over each interface is recorded, together with their remaining lifetime. In the case of a LAN interface, a complete set of CSNPs MUST consist of CSNPs received from neighbors that are not restarting. If there are multiple interfaces on the (re)starting router, the recorded set of LSPIDs is

the union of those received over each interface. LSPs with a remaining lifetime of zero are NOT so recorded.

As LSPs are received (by the normal operation of the update process) over any interface, the corresponding LSPID entry is removed (it is also removed if an LSP arrives before the CSNP containing the reference). When an LSPID has been held in the list for its indicated remaining lifetime, it is removed from the list. When the list of LSPIDs is empty and the timer T1 has been cancelled for all the interfaces that have an adjacency at this level, the timer T2 is cancelled.

At this point, the local database is guaranteed to contain all the LSP(s) (either the same sequence number or a more recent sequence number) that were present in the neighbors' databases at the time of (re)starting. LSPs that arrived in a neighbor's database after the time of (re)starting may or may not be present, but the normal operation of the update process will guarantee that they will eventually be received. At this point, the local database is deemed to be "synchronized".

Since LSPs mentioned in the CSNP(s) with a zero remaining lifetime are not recorded, and those with a short remaining lifetime are deleted from the list when the lifetime expires, cancellation of the timer T2 will not be prevented by waiting for an LSP that will never arrive.

3.4.1. LSP Generation and Flooding and SPF Computation

The operation of a router starting, as opposed to restarting, is somewhat different. These two cases are dealt with separately below.

3.4.1.1. Restarting

In order to avoid causing unnecessary routing churn in other routers, it is highly desirable that the router's own LSPs generated by the restarting system are the same as those previously present in the network (assuming no other changes have taken place). It is important therefore not to regenerate and flood the LSPs until all the adjacencies have been re-established and any information required for propagation into the local LSPs is fully available. Ideally, the information is loaded into the LSPs in a deterministic way, such that the same information occurs in the same place in the same LSP (and hence the LSPs are identical to their previous versions). If this can be achieved, the new versions may not even cause SPF to be run in other systems. However, provided the same information is included in the set of LSPs (albeit in a different order, and possibly different

LSPs), the result of running the SPF will be the same and will not cause churn to the forwarding tables.

In the case of a restarting router, none of the router's own LSPs are transmitted, nor are the router's own forwarding tables updated while the timer T3 is running.

Redistribution of inter-level information MUST be regenerated before this router's LSP is flooded to other nodes. Therefore, the Level-n non-pseudonode LSP(s) MUST NOT be flooded until the other level's T2 timer has expired and its SPF has been run. This ensures that any inter-level information that is to be propagated can be included in the Level-n LSP(s).

During this period, if one of the router's own (including pseudonodes) LSPs is received, which the local router does not currently have in its own database, it is NOT purged. Under normal operation, such an LSP would be purged, since the LSP clearly should not be present in the global LSP database. However, in the present circumstances, this would be highly undesirable, because it could cause premature removal of a router's own LSP -- and hence churn in remote routers. Even if the local system has one or more of the router's own LSPs (which it has generated, but not yet transmitted), it is still not valid to compare the received LSP against this set, since it may be that as a result of propagation between Level 1 and Level 2 (or vice versa), a further router's own LSP will need to be generated when the LSP databases have synchronized.

During this period, a restarting router SHOULD send CSNPs as it normally would. Information about the router's own LSPs MAY be included, but if it is included it MUST be based on LSPs that have been received, not on versions that have been generated (but not yet transmitted). This restriction is necessary to prevent premature removal of an LSP from the global LSP database.

When the timer T2 expires or is cancelled indicating that synchronization for that level is complete, the SPF for that level is run in order to derive any information that is required to be propagated to another level, but the forwarding tables are not yet updated.

Once the other level's SPF has run and any inter-level propagation has been resolved, the router's own LSPs can be generated and flooded. Any own LSPs that were previously ignored, but that are not part of the current set of own LSPs (including pseudonodes), MUST then be purged. Note that it is possible that a Designated Router change may have taken place, and consequently the router SHOULD purge

those pseudonode LSPs that it previously owned, but that are now no longer part of its set of pseudonode LSPs.

When all the T2 timers have expired or been cancelled, the timer T3 is cancelled and the local forwarding tables are updated.

If the timer T3 expires before all the T2 timers have expired or been cancelled, this indicates that the synchronization process is taking longer than the minimum holding time of the neighbors. The router's own LSP(s) for levels that have not yet completed their first SPF computation are then flooded with the overload bit set to indicate that the router's LSPDB is not yet synchronized (and therefore other routers MUST NOT compute routes through this router). Normal operation of the update process resumes, and the local forwarding tables are updated. In order to prevent the neighbor's adjacencies from expiring, IIHs with the normal interface value for the holding time are transmitted over all interfaces with neither RR nor RA set in the restart TLV. This will cause the neighbors to refresh their adjacencies. The router's own LSP(s) will continue to have the overload bit set until timer T2 has expired or been cancelled.

3.4.1.2. Starting

In the case of a starting router, as soon as each adjacency is established, and before any CSNP exchanges, the router's own zeroth LSP is transmitted with the overload bit set. This prevents other routers from computing routes through the router until it has reliably acquired the complete set of LSPs. The overload bit remains set in subsequent transmissions of the zeroth LSP (such as will occur if a previous copy of the router's own zeroth LSP is still present in the network) while any timer T2 is running.

When all the T2 timers have been cancelled, the router's own LSP(s) MAY be regenerated with the overload bit clear (assuming the router is not in fact overloaded, and there is no other reason, such as incomplete BGP convergence, to keep the overload bit set) and flooded as normal.

Other LSPs owned by this router (including pseudonodes) are generated and flooded as normal, irrespective of the timer T2. The SPF is also run as normal and the Routing Information Base (RIB) and Forwarding Information Base (FIB) updated as routes become available.

To avoid the possible formation of temporary blackholes, the starting router sets the SA bit in the restart TLV (as described in Section 3.3.2) in all IIHs that it sends.

When all T2 timers have been cancelled, the starting router MUST transmit IIHs with the SA bit clear.

4. State Tables

This section presents state tables that summarize the behaviors described in this document. Other behaviors, in particular adjacency state transitions and LSP database update operation, are NOT included in the state tables except where this document modifies the behaviors described in [ISO10589] and [RFC5303].

The states named in the columns of the tables below are a mixture of states that are specific to a single adjacency (ADJ suppressed, ADJ Seen RA, ADJ Seen CSNP) and states that are indicative of the state of the protocol instance (Running, Restarting, Starting, SPF Wait).

Three state tables are presented from the point of view of a running router, a restarting router, and a starting router.

4.1. Running Router

Event	Running	ADJ suppressed
RX PR	Set Planned Restart state. Update hold time Send PA	
RX PR clr and RR clr	Clear Planned Restart State Restore holdtime to local value	
RX RR	Maintain ADJ State Send RA Set SRM,send CSNP (Note 1) Update Hold Time, set Restart Mode (Note 2)	
RX RR clr	Clr Restart mode	
RX SA	Suppress IS neighbor TLV in LSP(s) Goto ADJ Suppressed	
RX SA clr		Unsuppress IS neighbor TLV in LSP(s) Goto Running

Note 1: CSNPs are sent by routers in accordance with Section 3.2.1c

Note 2: If Restart Mode clear

4.2. Restarting Router

Event	Restarting	ADJ Seen RA	ADJ Seen CSNP	SPF Wait
Restart planned	Send PR			
Planned restart canceled	Send PR clr			

RX PA	Proceed with planned restart			
Router restarts	Send IIH/RR ADJ Init Start T1,T2,T3			
RX RR	Send RA			
RX RA	Adjust T3 Goto ADJ Seen RA		Cancel T1 Adjust T3	
RX CSNP set	Goto ADJ Seen CSNP	Cancel T1		
RX IIH w/o Restart TLV	Cancel T1 (Point-to-point only)			
T1 expires	Send IIH/RR Restart T1	Send IIH/RR Restart T1	Send IIH/RR Restart T1	
T1 expires nth time	Send IIH/ normal	Send IIH/ normal	Send IIH/ normal	
T2 expires	Trigger SPF Goto SPF Wait			
T3 expires	Set overload bit Flood local LSPs Update fwd plane			
LSP DB Sync	Cancel T2, and T3 Trigger SPF Goto SPF wait			
All SPF done				Clear overload bit Update fwd plane Flood local LSPs Goto Running

4.3. Starting Router

Event	Starting	ADJ Seen RA	ADJ Seen CSNP
Router starts	Send IIH/SA Start T1,T2		
RX RR	Send RA		
RX RA	Goto ADJ Seen RA		Cancel T1
RX CSNP Set	Goto ADJ Seen CSNP	Cancel T1	
RX IIH w no Restart TLV	Cancel T1 (Point-to-Point only)		
ADJ UP	Start T1 Send local LSPs with overload bit set		
T1 expires	Send IIH/RR and SA Restart T1	Send IIH/RR and SA Restart T1	Send IIH/RR and SA Restart T1
T1 expires nth time	Send IIH/SA	Send IIH/SA	Send IIH/SA
T2 expires	Clear overload bit Send IIH normal Goto Running		
LSP DB Sync	Cancel T2 Clear overload bit Send IIH normal		

5. IANA Considerations

This document defines the following IS-IS TLV that is listed in the IS-IS TLV codepoint registry:

Type	Description	IIH	LSP	SNP	Purge
211	Restart TLV	y	n	n	n

IANA is requested to update the entry in registry to point to this document.

6. Security Considerations

Any new security issues raised by the procedures in this document depend upon the ability of an attacker to inject a false but apparently valid IIH, the ease/difficulty of which has not been altered.

If the RR bit is set in a false IIH, neighbors who receive such an IIH will continue to maintain an existing adjacency in the "UP" state and may (re)send a complete set of CSNPs. While the latter action is wasteful, neither action causes any disruption in correct protocol operation.

If the RA bit is set in a false IIH, a (re)starting router that receives such an IIH may falsely believe that there is a neighbor on the corresponding interface that supports the procedures described in this document. In the absence of receipt of a complete set of CSNPs on that interface, this could delay the completion of (re)start procedures by requiring the timer T1 to time out the locally defined maximum number of retries. This behavior is the same as would occur on a LAN where none of the (re)starting router's neighbors support the procedures in this document and is covered in Sections 3.3.1 and 3.3.2.

If the SA bit is set in a false IIH, this could cause suppression of the advertisement of an IS neighbor, which could either continue for an indefinite period or occur intermittently with the result being a possible loss of reachability to some destinations in the network and/or increased frequency of LSP flooding and SPF calculation.

If the PR bit is set in a false IIH, neighbors who receive such an IIH could modify the holding time of an existing adjacency inappropriately. In the event of topology changes, the neighbor might also choose to not flood the topology updates and/or bring the adjacency down in the false belief that the forwarding plane of the router identified as the source of the false IIH is not currently processing announced topology changes. This would result in unnecessary forwarding disruption.

If the PA bit is set in a false IIH, a router that receives such an IIH may falsely believe that the neighbor on the corresponding interface supports the planned restart procedures defined in this document. If such a router is planning to restart it might then proceed to initiate a restart in the false expectation that the neighbor has updated its holding time as requested. This may result

in the neighbor bringing down the adjacency while the receiving router is restarting, causing unnecessary disruption to forwarding.

The possibility of IS-IS PDU spoofing can be reduced by the use of authentication as described in [RFC1195] and [ISO10589], and especially the use of cryptographic authentication as described in [RFC5304] and [RFC5310].

7. Manageability Considerations

These extensions that have been designed, developed, and deployed for many years do not have any new impact on management and operation of the IS-IS protocol via this standardization process.

8. Acknowledgements

For RFC 5306 the authors acknowledged contributions made by Jeff Parker, Radia Perlman, Mark Schaefer, Naiming Shen, Nischal Sheth, Russ White, and Rena Yang.

The authors of this updated version acknowledge the contribution of Mike Shand, co-auther of RFC 5306.

9. Normative References

[ISO10589] International Organization for Standardization, "Intermediate system to Intermediate system intra-domain routeing 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.

[RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, DOI 10.17487/RFC1195, December 1990, <https://www.rfc-editor.org/info/rfc1195>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5303] Katz, D., Saluja, R., and D. Eastlake 3rd, "Three-Way Handshake for IS-IS Point-to-Point Adjacencies", RFC 5303, DOI 10.17487/RFC5303, October 2008, <https://www.rfc-editor.org/info/rfc5303>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

Appendix A. Summary of Changes from RFC 5306

This document extends RFC 5306 by introducing support for signalling the neighbors of a restarting router that a planned restart is about to occur. This allows the neighbors to be aware of the state of the restarting router so that appropriate action may be taken if other topology changes occur while the planned restart is in progress. Since the forwarding plane of the restarting router is maintained based upon the pre-restart state of the network, additional topology changes introduce the possibility that traffic may be lost if paths via the restarting router continue to be used while the restart is in progress.

In support of this new functionality two new flags have been introduced:

PR - Restart is planned
PA - Planned restart acknowledgement

No changes to the post restart exchange between the restarting router and its neighbors have been introduced.

Authors' Addresses

Les Ginsberg
Cisco Systems, Inc.

Email: ginsberg@cisco.com

Paul Wells
Cisco Systems, Inc.

Email: pauwells@cisco.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: December 30, 2018

T. Li
Arista Networks

June 28, 2018

Level 1 Area Abstraction for IS-IS
draft-li-area-abstraction-00

Abstract

Link state routing protocols have hierarchical abstraction already built into them. However, when lower levels are used for transit, they must expose their internal topologies, leading to scale issues.

To avoid this, this document discusses extensions to the IS-IS routing protocol that would allow level 1 areas to provide transit, yet only inject an abstraction of the topology into level 2.

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 December 30, 2018.

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

The IS-IS routing protocol IS-IS [ISO10589] currently supports a two level hierarchy of abstraction. The fundamental unit of abstraction is the 'area', which is a (hopefully) connected set of systems running IS-IS at the same level. Level 1, the lowest level, is abstracted by routers that participate in both Level 1 and Level 2, and they inject area information into Level 2. Level 2 systems seeking to access Level 1, use this abstraction to compute the shortest path to the Level 1 area. The full topology database of Level 1 is not injected into Level 2, only a summary of the address space contained within the area, so the scalability of the Level 2 link state database is protected.

This works well if the Level 1 area is tangential to the Level 2 area. This also works well if there are a number of routers in both Level 1 and Level 2 and they are adjacent, so Level 2 traffic will never need to transit Level 1 only routers. Level 1 will not contain any Level 2 topology, and Level 2 will only contain area abstractions for Level 1.

Unfortunately, this scheme does not work so well if the Level 1 area needs to provide transit for Level 2 traffic. For Level 2 shortest path first (SPF) computations to work correctly, the transit topology must also appear in the Level 2 link state database. This implies that all routers that could possibly provide transit, plus any links that might also provide Level 2 transit must also become part of the

Level 2 topology. If this is a relatively tiny portion of the Level 1 area, this is not onerous.

However, with today's data center topologies, this is problematic. A common application is to use a Layer 3 Leaf-Spine (L3LS) topology, which is a folded 3-stage Clos [Clos] fabric. It can also be thought of as a complete bipartite graph. In such a topology, the desire is to use Level 1 to contain the routing of the entire L3LS topology and then to use Level 2 for the remainder of the network. Leaves in the L3LS topology are appropriate for connection outside of the data center itself, so they would provide connectivity for Level 2. If there are multiple connections to Level 2 for redundancy, or to other areas, these too would also be made to the leaves in the topology. This creates a difficulty because there are now multiple Level 2 leaves in the topology, with connectivity between the leaves provide by the spines.

Following the rules of IS-IS, all spine routers would necessarily be part of the Level 2 topology, plus all links between a Level 2 leaf and the spines. In the limit, where all leaves need to support Level 2, it implies that the entire L3LS topology becomes part of Level 2. This is seriously problematic as it more than doubles the link state database held in the L3LS topology and eliminates any benefits of the hierarchy.

1.1. 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 RFC 2119 [RFC2119].

2. Area Abstraction

We propose to completely abstract away the Level 2 topology of the Level 1 area, making the entire area look like a single system directly connected to all of the area's Level 2 neighbors. By only providing an abstraction of the topology, Level 2's requirement for connectivity can be satisfied without the full overhead of the area's internal topology. It then becomes the responsibility of the Level 1 area to ensure the forwarding connectivity that's advertised.

We propose to implement Area Abstraction by having a Level 2 pseudonode that represents the entire Level 1 area. This is the only LSP from the area that will be injected into the overall Level 2 link state database.

There are three classes of routers that we need to be concerned with in this discussion:

Area Leader The Area Leader is a router in the Level 1 area that is elected to represent the Level 1 area by injecting an LSP into the Level 2 link state database.

Area Edge Router An Area Edge Router is a router that is part of the Level 1 area and has at least one Level 2 interface outside of the Area.

Area Neighbor An Area Neighbor is a Level 2 router that is outside of the Level 1 Area.

The Area Leader has several responsibilities. First, it must inject a pseudonode identifier into the Level 1 link state database. This is the Area Pseudonode Identifier. Second, the Area Leader must generate the pseudonode LSP for the Area.

All Area Edge Routers learn the Area Pseudonode Identifier from the Level 1 link state database and use that as the identifier in their Level 2 IS-IS Hello PDUs on interfaces outside the Level 1 area. The Area Edge Routers MUST also maintain an Level 2 adjacency with the Area Leader, either via a direct link or via a tunnel.

Area Edge Routers MUST be able to provide transit to Level 2 traffic. We propose that the Area Edge Routers use Segment Routing (SR) [I-D.ietf-spring-segment-routing] and, during Level 2 SPF computation, use the SR forwarding path to reach the exit Area Edge Routers.

2.1. Area Leader Election

The Area Leader is selected using the election mechanisms described in Dynamic Flooding for IS-IS [I-D.li-dynamic-flooding].

2.2. LSP Generation

Area Edge Routers generate a Level 2 LSP that includes adjacencies to any Area Neighbors and the Area Leader. This LSP is not advertised outside of the area.

The Area Leader uses the Level 2 LSPs generated by the Area Edge Routers to generate the Area Pseudonode LSP. This LSP is originated using the Area Pseudonode Identifier and includes adjacencies for all of the Area Neighbors that have been advertised by the Area Edge Routers. The Area Pseudonode LSP is the only LSP that is injected into the overall Level 2 link state database, with all other Level 2 LSPs from the area being filtered out at the area boundary.

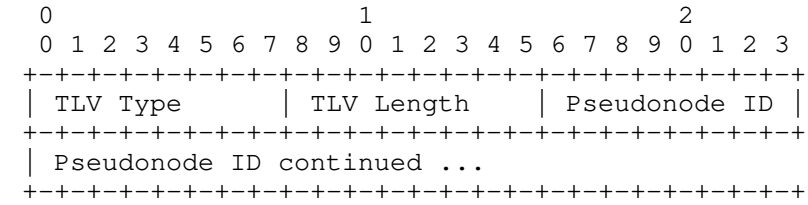
2.3. Redundancy

If the Area Leader fails, another candidate may become Area Leader and MUST regenerate the Area Pseudonode LSP. The failure of the Area Leader is not visible outside of the area and appears to simply be an update of the Area Pseudonode LSP.

3. Area Pseudonode TLV

The Area Pseudonode TLV allows the Area Leader to advertise the existence of an Area Pseudonode Identifier. This TLV is injected into one of the Area Leader's Level 1 LSPs.

The format of the Area Pseudonode TLV is:



TLV Type: XXX

TLV Length: 2 + (length of a system ID + 1)

Pseudonode ID: A pseudonode ID, which is the length of a system ID plus one octet. field.

4. Acknowledgements

To be written.

5. IANA Considerations

This memo requests that IANA allocate and assign one code point from the IS-IS TLV Codepoints registry for the Area Pseudonode TLV.

6. Security Considerations

This document introduces no new security issues. Security of routing within a domain is already addressed as part of the routing protocols themselves. This document proposes no changes to those security architectures.

7. References

7.1. Normative References

[I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-15 (work in progress), January 2018.

[I-D.li-dynamic-flooding]
Li, T. and P. Psenak, "Dynamic Flooding on Dense Graphs", draft-li-dynamic-flooding-05 (work in progress), June 2018.

[ISO10589]
International Organization for Standardization, "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Nov. 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

7.2. Informative References

[Clos] Clos, C., "A Study of Non-Blocking Switching Networks", The Bell System Technical Journal Vol. 32(2), DOI 10.1002/j.1538-7305.1953.tb01433.x, March 1953, <http://dx.doi.org/10.1002/j.1538-7305.1953.tb01433.x>.

Author's Address

Tony Li
Arista Networks
5453 Great America Parkway
Santa Clara, California 95054
USA

Email: tony.li@tony.li

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: December 30, 2018

T. Li
Arista Networks

June 28, 2018

Hierarchical IS-IS
draft-li-hierarchical-isis-00

Abstract

The IS-IS routing protocol was originally defined with a two level hierarchical structure. This was adequate for the networks at the time. As we continue to expand the scale of our networks, it is apparent that additional hierarchy would be a welcome degree of flexibility in network design.

This document defines IS-IS Levels 3 through 8.

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 December 30, 2018.

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

The IS-IS routing protocol IS-IS [ISO10589] currently supports a two level hierarchy of abstraction. The fundamental unit of abstraction is the 'area', which is a (hopefully) connected set of systems running IS-IS at the same level. Level 1, the lowest level, is abstracted by routers that participate in both Level 1 and Level 2.

Practical considerations, such as the size of an area's link state database, cause network designers to restrict the number of routers in any given area. Concurrently, the dominance of scale-out architectures based around small routers has created a situation where the scalability limits of the protocol are going to become critical in the foreseeable future.

The goal of this document is to enable additional hierarchy within IS-IS by creating additional hierarchy. Each additional level of hierarchy has a multiplicative effect on scale, so the addtion of six levels should be a significant improvement. While all six levels may not be needed in the short term, it is apparent that the original designers of IS-IS reserved enough space for these levels, and defining six additional levels is only slightly harder than adding a

single level, so it makes some sense to expand the design for the future.

The modifications described herein are designed to be fully backward compatible.

Section references in this document are references to sections of IS-IS [ISO10589].

1.1. 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 RFC 2119 [RFC2119].

2. PDU changes

In this section, we enumerate all of the redefinitions of protocol header fields necessary to add additional levels.

2.1. Circuit Type

In the fixed header of some IS-IS PDUs, a field is named 'Reserved/ Circuit Type' (Section 9.5). The high order six bits are reserved, with the low order two bits indicating Level 1 (bit 1) and Level 2 (bit 2).

This field is renamed to be 'Circuit Type'. The bits are redefined as follows:

1. Level 1
2. Level 2
3. Level 3
4. Level 4
5. Level 5
6. Level 6
7. Level 7
8. Level 8

The value of zero (no bits set) is reserved. PDUs with a Circuit Type of zero SHALL be ignored.

The set bits of the Circuit Type MUST be contiguous. If bit n and bit m are set in the Circuit Type, then all bits in the interval [n:m] must be set.

2.2. PDU Type

The fixed header of IS-IS PDUs contains an octet with three reserved bits and the 'PDU Type' field. The three reserved bits are transmitted as zero and ignored on receipt. (Section 9.5)

To allow for additional PDU space, this entire octet is renamed the 'PDU Type' field.

3. Additional PDUs

3.1. LAN IS to IS hello PDU (LAN-HELLO-PDU)

The 'LAN IS to IS hello PDU' (LAN-HELLO-PDU) is identical in format to the 'Level 2 LAN IS to IS hello PDU' (Section 9.6), except that the PDU Type has value AAA. The LAN-HELLO-PDU MUST be used instead of the 'Level 1 LAN IS to IS hello PDU' (Section 9.5) or the 'Level 2 LAN IS to IS hello PDU' on any circuit that has one or more of Level 3 through Level 8 enabled.

3.2. Point-to-point IS to IS hello PDU (P2P-HELLO-PDU)

The 'Point-to-point IS to IS hello PDU' can be used on circuits of any Level without modification.

3.3. Level n Link State PDU (Ln-LSP-PDU)

The 'Level n Link State PDU' (Ln-LSP-PDU) has the same format as the 'Level 2 Link State PDU' (Section 9.9), except for the PDU Type. The PDU Types for Levels 3 through 8 are defined as follows:

Level 3 (L3-LSP-PDU): BBB

Level 4 (L4-LSP-PDU): CCC

Level 5 (L5-LSP-PDU): DDD

Level 6 (L6-LSP-PDU): EEE

Level 7 (L7-LSP-PDU): FFF

Level 8 (L8-LSP-PDU): GGG

3.4. Level n complete sequence numbers PDU (Ln-CSNP-PDU)

The 'Level n complete sequence numbers PDU' (Ln-CSNP-PDU) has the same format as the 'Level 2 complete sequence numbers PDU' (Section 9.11), except for the PDU Type. The PDU Types for Levels 3 through 8 are defined as follows:

Level 3 (L3-CSNP-PDU): HHH

Level 4 (L4-CSNP-PDU): III

Level 5 (L5-CSNP-PDU): JJJ

Level 6 (L6-CSNP-PDU): KKK

Level 7 (L7-CSNP-PDU): LLL

Level 8 (L8-CSNP-PDU): MMM

3.5. Level n partial sequence numbers PDU (Ln-PSNP-PDU)

The 'Level 2 partial sequence numbers PDU' (Ln-PSNP-PDU) has the same format as the 'Level 2 partial sequence numbers PDU' (Section 9.13), except for the PDU Type. The PDU Types for Levels 3 through 8 are defined as follows:

Level 3 (L3-PSNP-PDU): NNN

Level 4 (L4-PSNP-PDU): OOO

Level 5 (L5-PSNP-PDU): PPP

Level 6 (L6-PSNP-PDU): QQQ

Level 7 (L7-PSNP-PDU): RRR

Level 8 (L8-PSNP-PDU): SSS

4. Inheritance of TLVs

All existing Level 2 TLVs may be used in the corresponding Level 3 through Level 8 PDUs. When used in a Level 3 through Level 8 PDU, the semantics of these TLVs will be applied to the Level of the containing PDU. If the original semantics of the PDU was carrying a reference to Level 1 in a Level 2 TLV, then the semantics of the TLV at level N will be a reference to level N-1. The intent is to retain the original semantics of the TLV at the higher level.

5. Acknowledgements

The author would like to thank Dinesh Dutt for inspiring this document.

6. IANA Considerations

This document makes many requests to IANA, as follows:

6.1. PDU Type

The existing IS-IS PDU registry currently supports values 0-31. This should be expanded to support the values 0-255. The existing value assignments should be retained. Value 255 should be reserved.

6.2. New PDUs

IANA is requested to allocate values from the IS-IS PDU registry for the following:

LAN-HELLO-PDU: AAA

L3-LSP-PDU: BBB

L4-LSP-PDU: CCC

L5-LSP-PDU: DDD

L6-LSP-PDU: EEE

L7-LSP-PDU: FFF

L8-LSP-PDU: GGG

L3-CSNP-PDU: HHH

L4-CSNP-PDU: III

L5-CSNP-PDU: JJJ

L6-CSNP-PDU: KKK

L7-CSNP-PDU: LLL

L8-CSNP-PDU: MMM

L3-PSNP-PDU: NNN

L4-PSNP-PDU: OOO

L5-PSNP-PDU: PPP

L6-PSNP-PDU: QQQ

L7-PSNP-PDU: RRR

L8-PSNP-PDU: SSS

To allow for PDU types to be defined independent of this document, the above values should be allocated from the range 32-254.

7. Security Considerations

This document introduces no new security issues. Security of routing within a domain is already addressed as part of the routing protocols themselves. This document proposes no changes to those security architectures.

8. Normative References

[ISO10589] International Organization for Standardization, "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Nov. 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

Author's Address

Tony Li
Arista Networks
5453 Great America Parkway
Santa Clara, California 95054
USA

Email: tony.li@tony.li

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: June 6, 2019

T. Li, Ed.
Arista Networks
P. Psenak, Ed.
L. Ginsberg
Cisco Systems, Inc.
T. Przygienda
Juniper Networks, Inc.
D. Cooper
CenturyLink
L. Jalil
Verizon
S. Dontula
ATT

December 3, 2018

Dynamic Flooding on Dense Graphs
draft-li-lsr-dynamic-flooding-02

Abstract

Routing with link state protocols in dense network topologies can result in sub-optimal convergence times due to the overhead associated with flooding. This can be addressed by decreasing the flooding topology so that it is less dense.

This document discusses the problem in some depth and an architectural solution. Specific protocol changes for IS-IS, OSPFv2, and OSPFv3 are described in this document.

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 June 6, 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

In recent years, there has been increased focus on how to address the dynamic routing of networks that have a bipartite (a.k.a. spine-leaf or leaf-spine), Clos [Clos], or Fat Tree [Leiserson] topology. Conventional Interior Gateway Protocols (IGPs, i.e., IS-IS [ISO10589], OSPFv2 [RFC2328], and OSPFv3 [RFC5340]) under-perform, redundantly flooding information throughout the dense topology, leading to overloaded control plane inputs and thereby creating operational issues. For practical considerations, network architects have resorted to applying unconventional techniques to address the problem, e.g., applying BGP in the data center [RFC7938]. However it is very clear that using an Exterior Gateway Protocol as an IGP is sub-optimal, if only due to the configuration overhead.

The primary issue that is demonstrated when conventional mechanisms are applied is the poor reaction of the network to topology changes. Normal link state routing protocols rely on a flooding algorithm for state distribution within an area. In a dense topology, this flooding algorithm is highly redundant, resulting in unnecessary

overhead. Each node in the topology receives each link state update multiple times. Ultimately, all of the redundant copies will be discarded, but only after they have reached the control plane and been processed. This creates issues because significant link state database updates can become queued behind many redundant copies of another update. This delays convergence as the link state database does not stabilize promptly.

In a real world implementation, the packet queues leading to the control plane are necessarily of finite size, so if the flooding rate exceeds the update processing rate for long enough, the control plane will be obligated to drop incoming updates. If these lost updates are of significance, this will further delay stabilization of the link state database and the convergence of the network.

This is not a new problem. Historically, when routing protocols have been deployed in networks where the underlying topology is a complete graph, there have been similar issues. This was more common when the underlying link layer fabric presented the network layer with a full mesh of virtual connections. This was addressed by reducing the flooding topology through IS-IS Mesh Groups [RFC2973], but this approach requires careful configuration of the flooding topology.

Thus, the root problem is not limited to massively scalable data centers. It exists with any dense topology at scale.

This problem is not entirely surprising. Link state routing protocols were conceived when links were very expensive and topologies were sparse. The fact that those same designs are sub-optimal in a dense topology should not come as a huge surprise. The fundamental premise that was addressed by the original designs was an environment of extreme cost and scarcity. Technology has progressed to the point where links are cheap and common. This represents a complete reversal in the economic fundamentals of network engineering. The original designs are to be commended for continuing to provide correct operation to this point, and optimizations for operation in today’s environment are to be expected.

1.1. 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 RFC 2119 [RFC2119].

2. Problem Statement

In a dense topology, the flooding algorithm that is the heart of conventional link state routing protocols causes a great deal of redundant messaging. This is exacerbated by scale. While the protocol can survive this combination, the redundant messaging is unnecessary overhead and delays convergence. Thus, the problem is to provide routing in dense, scalable topologies with rapid convergence.

3. Solution Requirements

A solution to this problem must then meet the following requirements:

Requirement 1 Provide a dynamic routing solution. Reachability must be restored after any topology change.

Requirement 2 Provide a significant improvement in convergence.

Requirement 3 The solution should address a variety of dense topologies. Just addressing a complete bipartite topology such as K5,8 is insufficient. Multi-stage Clos topologies must also be addressed, as well as topologies that are slight variants. Addressing complete graphs is a good demonstration of generality.

Requirement 4 There must be no single point of failure. The loss of any link or node should not unduly hinder convergence.

Requirement 5 Dense topologies are subgraphs of much larger topologies. Operational efficiency requires that the dense subgraph not operate in a radically different manner than the remainder of the topology. While some operational differences are permissible, they should be minimized. Changes to nodes outside of the dense subgraph are not acceptable. These situations occur when massively scaled data centers are part of an overall larger wide-area network. Having a second protocol operating just on this subgraph would add much more complexity at the edge of the subgraph where the two protocols would have to inter-operate.

4. Dynamic Flooding

We have observed that the combination of the dense topology and flooding on the physical topology in a scalable network is sub-optimal. However, if we decouple the flooding topology from the physical topology and only flood on a greatly reduced portion of that topology, we can have efficient flooding and retain all of the resilience of existing protocols. A node that supports flooding on the decoupled flooding topology is said to support dynamic flooding.

In this idea, the flooding topology is computed within an IGP area with the dense topology either centrally on an elected node, termed the Area Leader, or in a distributed manner on all nodes that are supporting Dynamic Flooding. If the flooding topology is computed centrally, it is encoded into and distributed as part of the normal link state database. We call this the centralized mode of operation. If the flooding topology is computed in a distributed fashion, we call this the distributed mode of operation. Nodes within such an IGP area would only flood on the flooding topology. On links outside of the normal flooding topology, normal database synchronization mechanisms (i.e., OSPF database exchange, IS-IS CSNPs) would apply, but flooding may not. Details are described in Section 6. New link state information that arrives from outside of the flooding topology suggests that the sender has a different or no flooding topology information and that the link state update should be flooded on the flooding topology as well.

The flooding topology covers the full set of nodes within the area, but excludes some of the links that standard flooding would employ.

Since the flooding topology is computed prior to topology changes, it does not factor into the convergence time and can be done when the topology is stable. The speed of the computation and its distribution, in the case of a centralized mode, is not a significant issue.

If a node does not have any flooding topology information when it receives new link state information, it should flood according to standard flooding rules. This situation will occur when the dense topology is first established, but is unlikely to recur.

When centralized mode is used and if, during a transient, there are multiple flooding topologies being advertised, then nodes should flood link state updates on all of the flooding topologies. Each node should locally evaluate the election of the Area Leader for the IGP area and first flood on its flooding topology. The rationale behind this is straightforward: if there is a transient and there has been a recent change in Area Leader, then propagating topology information promptly along the most likely flooding topology should be the priority.

During transients, it is possible that loops will form in the flooding topology. This is not problematic, as the legacy flooding rules would cause duplicate updates to be ignored. Similarly, during transients, it is possible that the flooding topology may become disconnected. Section 6.7.11 discusses how such conditions are handled.

4.1. Applicability

In a complete graph, this approach is appealing because it drastically decreases the flooding topology without the manual configuration of mesh groups. By controlling the diameter of the flooding topology, as well as the maximum degree node in the flooding topology, convergence time goals can be met and the stability of the control plane can be assured.

Similarly, in a massively scaled data center, where there are many opportunities for redundant flooding, this mechanism ensures that flooding is redundant, with each leaf and spine well connected, while ensuring that no update need make too many hops and that no node shares an undue portion of the flooding effort.

In a network where only a portion of the nodes support Dynamic Flooding, the remaining nodes will continue to perform standard flooding. This is not an issue for correctness, as no node can become isolated.

Flooding that is initiated by nodes that support Dynamic Flooding will remain within the flooding topology until it reaches a legacy node, which will resume legacy flooding. Standard flooding will be bounded by nodes supporting Dynamic Flooding, which can help limit the propagation of unnecessary flooding. Whether or not the network can remain stable in this condition is unknown and may be very dependent on the number and location of the nodes that support Dynamic Flooding.

During incremental deployment of dynamic flooding an area will consist of one or more sets of connected nodes that support dynamic flooding and one or more sets of connected nodes that do not, i.e., nodes that support standard flooding. The flooding topology is the union of these sets of nodes. Each set of nodes that does not support dynamic flooding needs to be part of the flooding topology and such a set of nodes may provide connectivity between two or more sets of nodes that support dynamic flooding.

4.2. Leader election

A single node within the dense topology is elected as an Area Leader.

A generalization of the mechanisms used in existing Designated Router (OSPF) or Designated Intermediate-System (IS-IS) elections suffices. The elected node is known as the Area Leader.

In the case of centralized mode, the Area Leader is responsible for computing and distributing the flooding topology. When a new Area

Leader is elected and has distributed new flooding topology information, then any prior Area Leaders should withdraw any of their flooding topology information from their link state database entries.

In the case of distributed mode, the distributed algorithm advertised by the Area Leader MUST be used by all nodes that participate in Dynamic Flooding.

Not every node needs to be a candidate to be Area Leader within an area, as a single candidate is sufficient for correct operation. For redundancy, however, it is strongly RECOMMENDED that there be multiple candidates.

4.3. Computing the Flooding Topology

There is a great deal of flexibility in how the flooding topology may be computed. For resilience, it needs to at least contain a cycle of all nodes in the dense subgraph. However, additional links could be added to decrease the convergence time. The trade-off between the density of the flooding topology and the convergence time is a matter for further study. The exact algorithm for computing the flooding topology in the case of the centralized computation need not be standardized, as it is not an interoperability issue. Only the encoding of the result needs to be documented. In the case of distributed mode, all nodes in the IGP area need to use the same algorithm to compute the flooding topology. It is possible to use private algorithms to compute flooding topology, so long as all nodes in the IGP area use the same algorithm.

While the flooding topology should be a covering cycle, it need not be a Hamiltonian cycle where each node appears only once. In fact, in many relevant topologies this will not be possible e.g., K5,8. This is fortunate, as computing a Hamiltonian cycle is known to be NP-complete.

A simple algorithm to compute the topology for a complete bipartite graph is to simply select unvisited nodes on each side of the graph until both sides are completely visited. If the number of nodes on each side of the graph are unequal, then revisiting nodes on the less populated side of the graph will be inevitable. This algorithm can run in O(N) time, so is quite efficient.

While a simple cycle is adequate for correctness and resiliency, it may not be optimal for convergence. At scale, a cycle may have a diameter that is half the number of nodes in the graph. This could cause an undue delay in link state update propagation. Therefore it may be useful to have a bound on the diameter of the flooding topology. Introducing more links into the flooding topology would

reduce the diameter, but at the trade-off of possibly adding redundant messaging. The optimal trade-off between convergence time and graph diameter is for further study.

Similarly, if additional redundancy is added to the flooding topology, specific nodes in that topology may end up with a very high degree. This could result in overloading the control plane of those nodes, resulting in poor convergence. Thus, it may be optimal to have an upper bound on the degree of nodes in the flooding topology. Again, the optimal trade-off between graph diameter, node degree, and convergence time, and topology computation time is for further study.

If the leader chooses to include a multi-node broadcast LAN segment as part of the flooding topology, all of the connectivity to that LAN segment should be included as well. Once updates are flooded onto the LAN, they will be received by every attached node.

4.4. Topologies on Complete Bipartite Graphs

Complete bipartite graph topologies have become popular for data center applications and are commonly called leaf-spine or spine-leaf topologies. In this section, we discuss some flooding topologies that are of particular interest in these networks.

4.4.1. A Minimal Flooding Topology

We define a Minimal Flooding Topology on a complete bipartite graph as one in which the topology is connected and each node has at least degree two. This is of interest because it guarantees that the flooding topology has no single points of failure.

In practice, this implies that every leaf node in the flooding topology will have a degree of two. As there are usually more leaves than spines, the degree of the spines will be higher, but the load on the individual spines can be evenly distributed.

This type of flooding topology is also of interest because it scales well. As the number of leaves increases, we can construct flooding topologies that perform well. Specifically, for n spines and m leaves, if m >= n(n/2-1), then there is a flooding topology that has a diameter of four.

4.4.2. Xia Topologies

We define a Xia Topology on a complete bipartite graph as one in which all spine nodes are bi-connected through leaves with degree two, but the remaining leaves all have degree one and are evenly distributed across the spines.

Constructively, we can create a Xia topology by iterating through the spines. Each spine can be connected to the next spine by selecting any unused leaf. Since leaves are connected to all spines, all leaves will have a connection to both the first and second spine and we can therefore choose any leaf without loss of generality. Continuing this iteration across all of the spines, selecting a new leaf at each iteration, will result in a path that connects all spines. Adding one more leaf between the last and first spine will produce a cycle of n spines and n leaves.

At this point, m-n leaves remain unconnected. These can be distributed evenly across the remaining spines, connected by a single link.

Xia topologies represent a compromise that trades off increased risk and decreased performance for lower flooding amplification. Xia topologies will have a larger diameter. For m spines, the diameter will be m + 2.

In a Xia topology, some leaves are singly connected. This represents a risk in that in some failures, convergence may be delayed. However, there may be some alternate behaviors that can be employed to mitigate these risks. If a leaf node sees that its single link on the flooding topology has failed, it can compensate by performing a database synchronization check with a different spine. Similarly, if a leaf determines that its connected spine on the flooding topology has failed, it can compensate by performing a database synchronization check with a different spine. In both of these cases, the synchronization check is intended to ameliorate any delays in link state propagation due to the fragmentation of the flooding topology.

The benefit of this topology is that flooding load is easily understood. Each node in the spine cycle will never receive an update more than twice. For m leaves and n spines, a spine never transmits more than (m/n +1) updates.

4.4.3. Optimization

If two nodes are adjacent on the flooding topology and there are a set of parallel links between them, then any given update MUST be flooded over a single one of those links. Selection of the specific link is implementation specific.

4.5. Encoding the Flooding Topology

There are a variety of ways that the flooding topology could be encoded efficiently. If the topology was only a cycle, a simple list of the nodes in the topology would suffice. However, this is insufficiently flexible as it would require a slightly different encoding scheme as soon as a single additional link is added. Instead, we choose to encode the flooding topology as a set of intersecting paths, where each path is a set of connected edges.

Other encodings are certainly possible. We have attempted to make a useful trade off between simplicity, generality, and space.

5. Protocol Elements

5.1. IS-IS TLVs

The following TLVs/sub-TLVs are added to IS-IS:

1. A sub-TLV that an IS may inject into its LSP to indicate its preference for becoming Area Leader.

2. A sub-TLV that an IS may inject into its LSP to indicate that it supports Dynamic Flooding and the algorithms that it supports for distributed mode, if any.

3. A TLV to carry the list of system IDs that compromise the flooding topology for the area.

4. A TLV to carry a path which is part of the flooding topology

5. A TLV that requests flooding from the adjacent node

5.1.1. IS-IS Area Leader Sub-TLV

The Area Leader Sub-TLV allows a system to:

1. Indicate its eligibility and priority for becoming Area Leader.

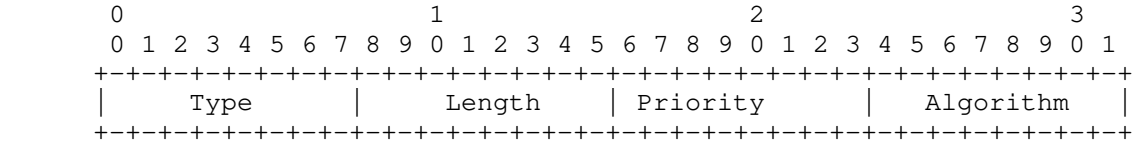
2. Indicate whether centralized or distributed mode is to be used to compute the flooding topology in the area.

3. Indicate the algorithm identifier for the algorithm that is used to compute the flooding topology in distributed mode.

Intermediate Systems (nodes) that are not advertising this Sub-TLV are not eligible to become Area Leader.

The Area Leader is the node with the numerically highest Area Leader priority in the area. In the event of ties, the node with the numerically highest system ID is the Area Leader. Due to transients during database flooding, different nodes may not agree on the Area Leader.

The Area Leader Sub-TLV is advertised as a Sub-TLV of the IS-IS Router Capability TLV-242 that is defined in [RFC7981] and has the following format:



Type: TBD1

Length: 2

Priority: 0-255, unsigned integer

Algorithm: a numeric identifier in the range 0-255 that identifies the algorithm used to calculate the flooding topology. The following values are defined:

0: Centralized computation by the Area Leader.

1-127: Standardized distributed algorithms. Individual values are are to be assigned according to the "Specification Required" policy defined in [RFC8126] (see Section 7.3).

128-254: Private distributed algorithms. Individual values are are to be assigned according to the "Private Use" policy defined in [RFC8126] (see Section 7.3).

255: Reserved

5.1.2. IS-IS Dynamic Flooding Sub-TLV

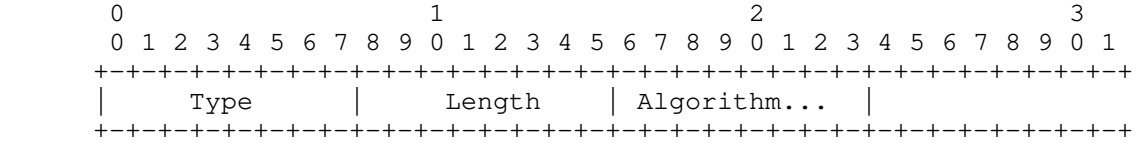
The Dynamic Flooding Sub-TLV allows a system to:

1. Indicate that it supports Dynamic Flooding. This is indicated by the advertisement of this Sub-TLV.

2. Indicate the set of algorithms that it supports for distributed mode, if any.

In incremental deployments, understanding which nodes support Dynamic Flooding can be used to optimize the flooding topology. In distributed mode, knowing the capabilities of the nodes can allow the Area Leader to select the optimal algorithm.

The Dynamic Flooding Sub-TLV is advertised as a Sub-TLV of the IS-IS Router Capability TLV (242) [RFC7981] and has the following format:



Type: TBD7

Length: 0-255; number of Algorithms

Algorithm: zero or more numeric identifiers in the range 0-255 that identifies the algorithm used to calculate the flooding topology, as described in Section 5.1.1.

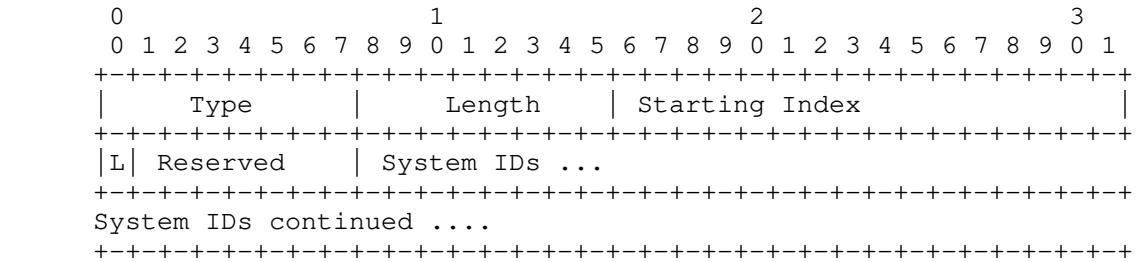
5.1.3. IS-IS Area System IDs TLV

IS-IS Area System IDs TLV is only used in centralized mode.

The Area System IDs TLV is used by the Area Leader to enumerate the system IDs that it has used in computing the flooding topology. Conceptually, the Area Leader creates a list of system IDs for all nodes in the area, assigning indices to each system, starting with index 0.

Because the space in a single TLV is small, more than one TLV may be required to encode all of the system IDs in the area. This TLV may be present in multiple LSPs.

The format of the Area System IDs TLV is:



Type: TBD2

Length: 3 + (System ID length * (number of System IDs))

Starting index: The index of the first system ID that appears in this TLV.

L (Last): This bit is set if the index of the last system ID that appears in this TLV is equal to the last index in the full list of system IDs for the area.

System IDs: A concatenated list of system IDs for the area.

If there are multiple IS-IS Area System IDs TLVs with the L bit set advertised by the same node, the TLV which specifies the smaller maximum index is used and the other TLV(s) with L bit set are ignored. TLVs which specify system IDs with indices greater than that specified by the TLV with the L bit set are also ignored.

5.1.4. IS-IS Flooding Path TLV

IS-IS Flooding Path TLV is only used in centralized mode.

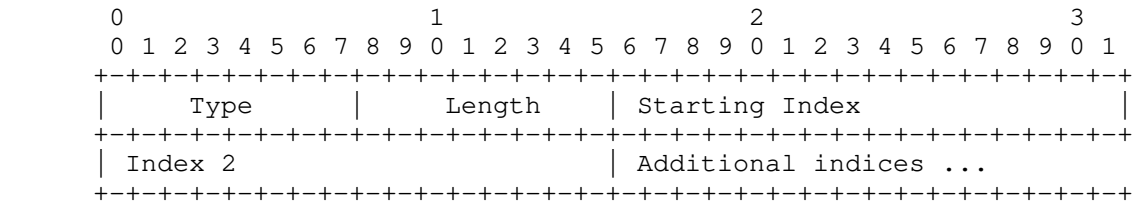
The Flooding Path TLV is used to denote a path in the flooding topology. The goal is an efficient encoding of the links of the topology. A single link is a simple case of a path that only covers two nodes. A connected path may be described as a sequence of indices: (I1, I2, I3, ...), denoting a link from the system with index 1 to the system with index 2, a link from the system with index 2 to the system with index 3, and so on.

If a path exceeds the size that can be stored in a single TLV, then the path may be distributed across multiple TLVs by the replication of a single system index.

Complex topologies that are not a single path can be described using multiple TLVs.

The Flooding Path TLV contains a list of system indices relative to the systems advertised through the Area System IDs TLV. At least 2 indices must be included in the TLV. Due to the length restriction of TLVs, this TLV can contain at most 126 system indices.

The Flooding Path TLV has the format:



Type: TBD3

Length: 2 * (number of indices in the path)

Starting index: The index of the first system in the path.

Index 2: The index of the next system in the path.

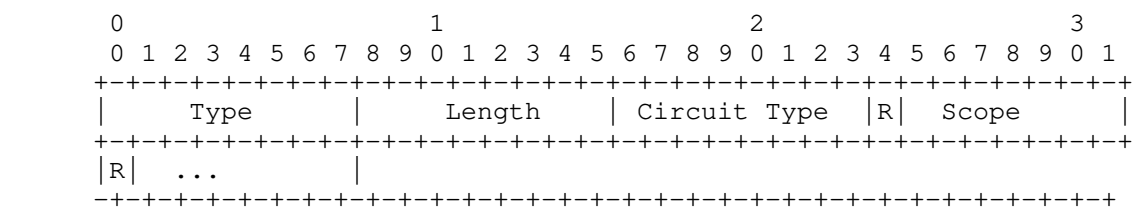
Additional indices (optional): A sequence of additional indices to systems along the path.

5.1.5. IS-IS Flooding Request TLV

The Flooding Request TLV allows a system to request an adjacent node to enable flooding towards it on a specific link in the case where the connection to adjacent node is not part of the existing flooding topology.

Nodes that support Dynamic Flooding MAY include the Flooding Request TLV in its IIH PDUs.

The Flooding Request TLV has the format:



Type: TBD9

Length: 1 + number of advertised Flooding Scopes

Circuit Type - circuit type as specified in IS-IS [ISO10589]

R bit: MUST be 0 and is ignored on receipt.

Scope: Flooding Scope for which the flooding is requested as defined by LSP Flooding Scope Identifier Registry defined by [RFC7356]. Inclusion of flooding scopes is optional and is only necessary if [RFC7356] is supported.

Circuit Flooding Scope MUST NOT be sent in the Flooding Request TLV and MUST be ignore if received.

If flooding was disabled on the received link due to Dynamic Flooding, then flooding MUST be temporarily enabled over the link for the specified Circuit Type(s) and Flooding Scope(s) received in the in the Flooding Request TLV. Flooding MUST be enabled until the Circuit Type or Flooding Scope is no longer advertised in the Flooding Request TLV or the TLV no longer appears in IIH PDUs received on the link.

When the flooding is temporarily enabled on the link for any Circuit Type or Flooding Scope due to received Flooding Request TLV, the receiver MUST perform standard database synchronization for the corresponding Circuit Type(s) and Flooding Scope(s) on the link. In the case of IS-IS, this results in setting SRM bit for all related LSPs on the link and sending CSNPs.

So long as the Flooding Request TLV is being received flooding MUST not be disabled for any of the Circuit Types or Flooding Scopes present in the Flooding Request TLV even if the connection between the neighbors is removed from the flooding topology. Flooding for such Circuit Types or Flooding Scopes MUST continue on the link and be considered as temporarily enabled.

5.2. OSPF LSAs and TLVs

This section defines new LSAs and TLVs for both OSPFv2 and OSPFv3.

Following objects are added:

1. A TLV that is used to advertise the preference for becoming Area Leader.

2. A TLV that is used to indicate the support for Dynamic Flooding and the algorithms that the advertising node supports for distributed mode, if any.

3. OSPFv2 Opaque LSA and OSPFv3 LSA to advertise the flooding topology for centralized mode.

4. A TLV to carry the list of system IDs that compromise the flooding topology for the area.

5. A TLV to carry a path which is part of the flooding topology.

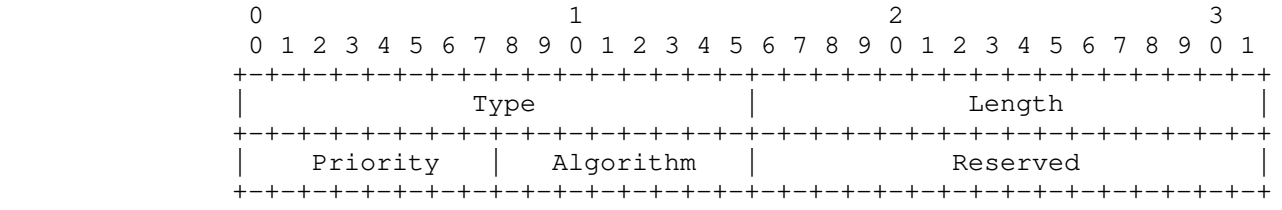
6. The bit in the LLS Type 1 Extended Options and Flags requests flooding from the adjacent node.

5.2.1. OSPF Area Leader Sub-TLV

The usage of the OSPF Area Leader Sub-TLV is identical to IS-IS and is described in Section 5.1.1.

The OSPF Area Leader Sub-TLV is used by both OSPFv2 and OSPFv3.

The OSPF Area Leader Sub-TLV is advertised as a top-level TLV of the RI LSA that is defined in [RFC7770] and has the following format:



Type: TBD4

Length: 4 octets

Priority: 0-255, unsigned integer

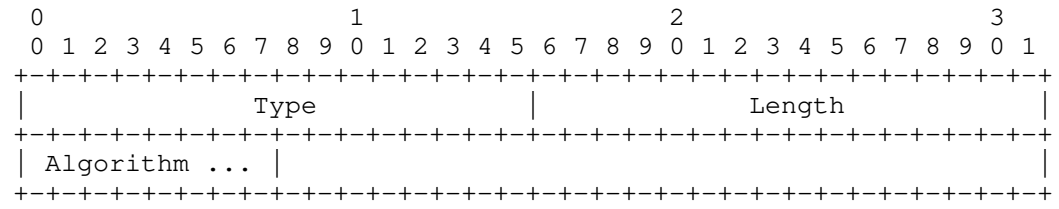
Algorithm: as defined in Section 5.1.1.

5.2.2. OSPF Dynamic Flooding Sub-TLV

The usage of the OSPF Dynamic Flooding Sub-TLV is identical to IS-IS and is described in Section 5.1.2.

The OSPF Dynamic Flooding Sub-TLV is used by both OSPFv2 and OSPFv3.

The OSPF Dynamic Flooding Sub-TLV is advertised as a top-level TLV of the RI LSA that is defined in [RFC7770] and has the following format:



Type: TBD8

Length: number of Algorithms

Algorithm: as defined in Section 5.1.1.

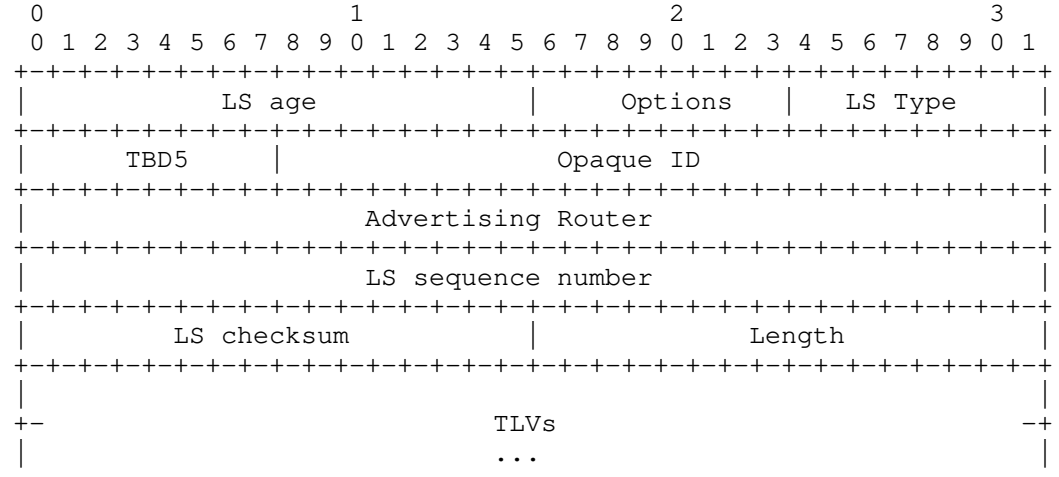
5.2.3. OSPFv2 Dynamic Flooding Opaque LSA

The OSPFv2 Dynamic Flooding Opaque LSA is only used in centralized mode.

The OSPFv2 Dynamic Flooding Opaque LSA is used to advertise additional data related to the dynamic flooding in OSPFv2. OSPFv2 Opaque LSAs are described in [RFC5250].

Multiple OSPFv2 Dynamic Flooding Opaque LSAs can be advertised by an OSPFv2 router. The flooding scope of the OSPFv2 Dynamic Flooding Opaque LSA is area-local.

The format of the OSPFv2 Dynamic Flooding Opaque LSA is as follows:



OSPFv2 Dynamic Flooding Opaque LSA

The opaque type used by OSPFv2 Dynamic Flooding Opaque LSA is TBD. The opaque type is used to differentiate the various type of OSPFv2 Opaque LSAs and is described in section 3 of [RFC5250]. The LS Type is 10. The LSA Length field [RFC2328] represents the total length (in octets) of the Opaque LSA including the LSA header and all TLVs (including padding).

The Opaque ID field is an arbitrary value used to maintain multiple Dynamic Flooding Opaque LSAs. For OSPFv2 Dynamic Flooding Opaque LSAs, the Opaque ID has no semantic significance other than to differentiate Dynamic Flooding Opaque LSAs originated by the same OSPFv2 router.

The format of the TLVs within the body of the OSPFv2 Dynamic Flooding Opaque LSA is the same as the format used by the Traffic Engineering Extensions to OSPF [RFC3630].

The Length field defines the length of the value portion in octets (thus a TLV with no value portion would have a length of 0). The TLV is padded to 4-octet alignment; padding is not included in the length field (so a 3-octet value would have a length of 3, but the total size of the TLV would be 8 octets). Nested TLVs are also 32-bit aligned. For example, a 1-octet value would have the length field set to 1, and 3 octets of padding would be added to the end of the value portion of the TLV. The padding is composed of zeros.

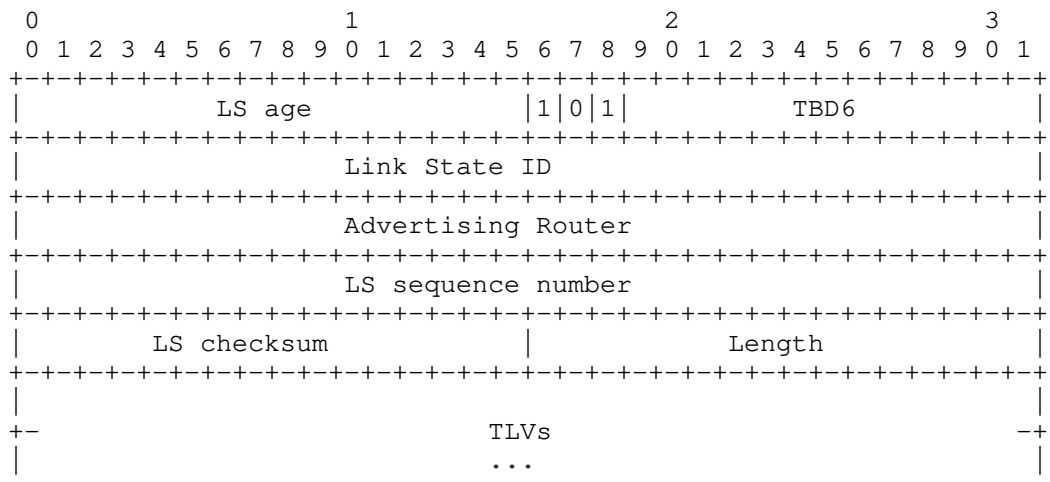
5.2.4. OSPFv3 Dynamic Flooding LSA

The OSPFv3 Dynamic Flooding Opaque LSA is only used in centralized mode.

The OSPFv3 Dynamic Flooding LSA is used to advertise additional data related to the dynamic flooding in OSPFv3.

The OSPFv3 Dynamic Flooding LSA has a function code of TBD. The flooding scope of the OSPFv3 Dynamic Flooding LSA is area-local. The U bit will be set indicating that the OSPFv3 Dynamic Flooding LSA should be flooded even if it is not understood. The Link State ID (LSID) value for this LSA is the Instance ID. OSPFv3 routers MAY advertise multiple Dynamic Flooding Opaque LSAs in each area.

The format of the OSPFv3 Dynamic Flooding LSA is as follows:



OSPFv3 Dynamic Flooding LSA

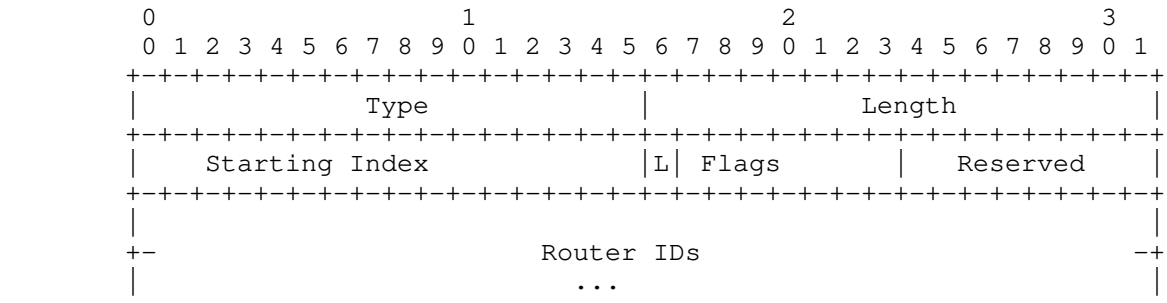
5.2.5. OSPF Area Router IDs TLV

The OSPF Area Router IDs TLV is a top level TLV of the OSPFv2 Dynamic Flooding Opaque LSA and OSPFv3 Dynamic Flooding LSA.

The OSPF Area Router IDs TLV is used by the Area Leader to enumerate the Router IDs that it has used in computing the flooding topology. Conceptually, the Area Leader creates a list of Router IDs for all routers in the area, assigning indices to each router, starting with index 0.

Because the space in a single OSPF Area Router IDs TLV is limited, more than one TLV may be required to encode all of the Router IDs in the area. This TLV may also recur in multiple OSPFv2 Dynamic Flooding Opaque LSAs or OSPFv3 Dynamic Flooding LSA, so that all Router IDs can be advertised.

The format of the Area Router IDs TLV is:



OSPF Area Router IDs TLV

TLV Type: 1

TLV Length: 4 + (Router ID length * (number of Router IDs))

Starting index: The index of the first Router ID that appears in this TLV.

L (Last): This bit is set if the index of the last system ID that appears in this TLV is equal to the last index in the full list of Rourer IDs for the area.

Router IDs: A concatenated list of Router IDs for the area.

If there are multiple OSPF Area Router IDs TLVs with the L bit set advertised by the same router, the TLV which specifies the smaller maximum index is used and the other TLV(s) with L bit set are ignored. TLVs which specify Router IDs with indices greater than that specified by the TLV with the L bit set are also ignored.

5.2.6. OSPF Flooding Path TLV

The OSPF Flooding Path TLV is a top level TLV of the OSPFv2 Dynamic Flooding Opaque LSAs and OSPFv3 Dynamic Flooding LSA.

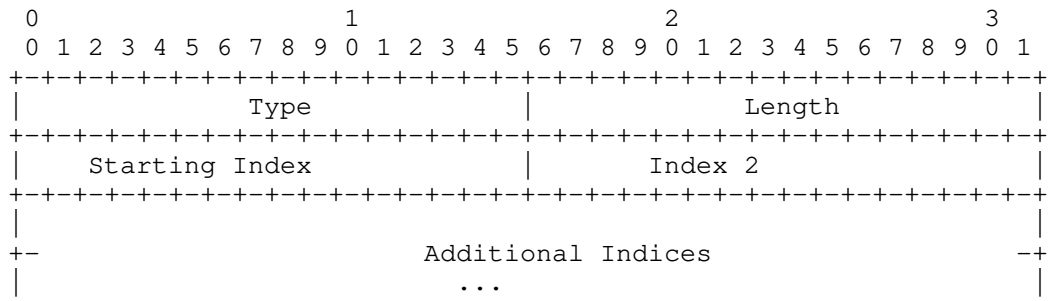
The usage of the OSPF Flooding Path TLV is identical to IS-IS and is described in Section 5.1.4.

The OSPF Flooding Path TLV contains a list of Router ID indices relative to the Router IDs advertised through the OSPF Area Router IDs TLV. At least 2 indices must be included in the TLV.

Multiple OSPF Flooding Path TLVs can be advertised in a single OSPFv2 Dynamic Flooding Opaque LSA or OSPFv3 Dynamic Flooding LSA. OSPF Flooding Path TLVs can also be advertised in multiple OSPFv2 Dynamic

Flooding Opaque LSAs or OSPFv3 Dynamic Flooding LSA, if they all can not fit in a single LSA.

The Flooding Path TLV has the format:



OSPF Flooding Path TLV

TLV Type: 2

TLV Length: 2 * (number of indices in the path)

Starting index: The index of the first Router ID in the path.

Index 2: The index of the next Router ID in the path.

Additional indices (optional): A sequence of additional indices to Router IDs along the path.

5.2.7. OSPF Flooding Request Bit

A single new option bit, the Flooding-Request (FR-bit), is defined in the LLS Type 1 Extended Options and Flags field [RFC2328]. The FR-bit allows a router to request an adjacent node to enable flooding towards it on a specific link in the case where the connection to adjacent node is not part of the current flooding topology.

Nodes that support Dynamic Flooding MAY include FR-bit in its OSPF LLS Extended Options and Flags TLV.

If FR-bit is signalled for an area for which the flooding on the link was disabled due to Dynamic Flooding, the flooding MUST be temporarily enabled over such link and area. Flooding MUST be enabled until FR-bit is no longer advertised in the OSPF LLS Extended

Options and Flags TLV or the OSPF LLS Extended Options and Flags TLV no longer appears in the OSPF Hellos.

When the flooding is temporarily enabled on the link for any area due to received FR-bit in OSPF LLS Extended Options and Flags TLV, the receiver MUST perform standard database synchronization for the corresponding area(s) on the link. If the adjacency is already in the FULL state, mechanism specified in [RFC4811] MUST be used for database resynchronization.

So long as the FR-bit is being received in the OSPF LLS Extended Options and Flags TLV for an area, flooding MUST not be disabled in such area even if the connection between the neighbors is removed from the flooding topology. Flooding for such area MUST continue on the link and be considered as temporarily enabled.

6. Behavioral Specification

In this section, we specify the detailed behaviors of the nodes participating in the IGP.

6.1. Terminology

We define some terminology here that is used in the following sections:

A node is considered reachable if it is part of the connected network graph. Note that this is independent of any constraints which may be considered when performing IGP SPT calculation (e.g., link metrics, OL bit state, etc.). Two-way-connectivity check MUST be performed before including an edge in the connected network graph.

Node is connected to the flooding topology, if it has at least one local link, which is part of the flooding topology.

Node is disconnected from the flooding topology when it is not connected to the flooding topology.

Current flooding topology - latest version of the flooding topology received (in case of the centralized mode) or calculated locally (in case of the distributed mode).

6.2. Flooding Topology

The flooding topology MUST include all reachable nodes in the area.

If a node's reachability changes, the flooding topology MUST be recalculated. In centralized mode, the Area Leader MUST advertise a new flooding topology.

If a node becomes disconnected from the current flooding topology but is still reachable then a new flooding topology MUST be calculated. In centralized mode the Area Leader MUST advertise the new flooding topology.

The flooding topology SHOULD be bi-connected.

6.3. Leader Election

Any node that is capable MAY advertise its eligibility to become Area Leader.

Nodes that are not reachable are not eligible as Area Leader. Nodes that do not advertise their eligibility to become Area Leader are not eligible. Amongst the eligible nodes, the node with the numerically highest priority is the Area Leader. If multiple nodes all have the highest priority, then the node with the numerically highest system identifier in the case of IS-IS, or Router-ID in the case of OSPFv2 and OSPFv3 is the Area Leader.

6.4. Area Leader Responsibilities

If the Area Leader operates in centralized mode, it MUST advertise algorithm 0 in its Area Leader Sub-TLV. It also MUST compute and advertise a flooding topology for the area. The Area Leader may update the flooding topology at any time, however, it should not destabilize the network with undue or overly frequent topology changes.

If the Area Leader operates in centralized mode and needs to advertises a new flooding topology, it floods a new flooding topology on both the new and old flooding topologies.

6.5. Distributed Flooding Topology Calculation

If the Area Leader advertises a non-zero algorithm in its Area Leader Sub-TLV, all nodes in the area that support Dynamic Flooding and the value of algorithm advertised by the Area Leader MUST compute the flooding topology based on the Area Leader's advertised algorithm.

Nodes that do not support the value of algorithm advertised by the Area Leader MUST continue to use standard flooding mechanism as defined by the protocol.

Nodes that do not support the value of algorithm advertised by the Area Leader MUST be considered as Dynamic Flooding incapable nodes by the Area Leader.

If the value of the algorithm advertised by the Area Leader is from the range 128-254 (private distributed algorithms), it is the responsibility of the network operator to guarantee that all nodes in the area have a common understanding of what the given algorithm value represents.

6.6. Flooding Behavior

Nodes that support Dynamic Flooding MUST use the flooding topology for flooding when possible, and MUST NOT revert to standard flooding when a valid flooding topology is available.

In some cases a node that supports Dynamic Flooding may need to add a local link(s) to the flooding topology temporarily, even though the link(s) is not part of the calculated flooding topology. This is termed "temporary flooding" and is discussed in Section 6.7.1.

The flooding topology is calculated locally in the case of distributed mode. In centralized mode the flooding topology is advertised in the area link state database. Received link state updates, whether received on a link that is in the flooding topology or on a link that is not in the flooding topology, MUST be flooded on all links that are in the flooding topology, except for the link on which the update was received.

In centralized mode, if multiple flooding topologies are present in the area link state database, the node SHOULD flood on the on each of these topologies.

When the flooding topology changes on a node, either as a result of the local computation in distributed mode or as a result of the advertisement from the Area Leader in centralized mode, the node MUST continue to flood on both the old and new flooding topology for a limited amount of time. This is required to provide all nodes sufficient time to migrate to the new flooding topology.

6.7. Treatment of Topology Events

In this section, we explicitly consider a variety of different topological events in the network and how Dynamic Flooding should address them.

6.7.1. Temporary Addition of Link to Flooding Topology

In some cases a node that supports Dynamic Flooding may need to add a local link(s) to the flooding topology temporarily, even though the link(s) is not part of the calculated flooding topology. We refer to this as "temporary flooding" on the link.

When temporary flooding is enabled on the link, the flooding needs to be enabled from both directions on such link. To achieve that, the following steps MUST be performed:

Link State Database needs to be re-synchronised on the link. This is done using the standard protocol mechanisms. In the case of IS-IS, this results in setting SRM bit for all LSPs on the circuit and sending compete set of CSNPs on it. In OSPF, the mechanism specified in [RFC4811] is used.

Flooding is enabled locally on the link.

Flooding is requested from the neighbor using the mechanism specified in section Section 5.1.5 or Section 5.2.7.

The request for temporary flooding is withdrawn on the link when all of the following conditions are met:

Node itself is connected to the current flooding topology.

Adjacent node is connected to the current flooding topology.

Any change in the flooding topology MUST result in evaluation of the above conditions for any link on which the temporary flooding was enabled.

Temporary flooding is stopped on the link when both adjacent nodes stop requesting temporary flooding on the link.

6.7.2. Local Link Addition

If a local link is added to the topology, the protocol will form a normal adjacency on the link and update the appropriate link state advertisements for the nodes on either end of the link. These link state updates will be flooded on the flooding topology.

In centralized mode, the Area Leader, upon receiving these updates, may choose to retain the existing flooding topology or may choose to modify the flooding topology. If it elects to change the flooding topology, it will update the flooding topology in the link state database and flood it using the new flooding topology.

In distributed mode, any change in the topology, including the link addition, MUST trigger the flooding topology recalculation. This is done to ensure that all nodes converge to the same flooding topology, regardless of the time of the calculation.

Temporary flooding MUST be enabled on the newly added local link, if at least one of the following conditions are met:

The node on which the local link was added is not connected to the current flooding topology.

The new adjacent node is not connected to the current flooding topology.

Note that in this case there is no need to perform a database synchronization as part of the enablement of the temporary flooding, because it has been part of the adjacency bring-up itself.

If multiple local links are added to the topology before the flooding topology is updated, temporary flooding MUST be enabled on a subset of these links.

6.7.3. Node Addition

If a node is added to the topology, then at least one link is also added to the topology. Section 6.7.2 applies.

6.7.4. Failures of Link Not on Flooding Topology

If a link that is not part of the flooding topology fails, then the adjacent nodes will update their link state advertisements and flood them on the flooding topology.

In centralized mode, the Area Leader, upon receiving these updates, may choose to retain the existing flooding topology or may choose to modify the flooding topology. If it elects to change the flooding topology, it will update the flooding topology in the link state database and flood it using the new flooding topology.

In distributed mode, any change in the topology, including the failure of the link that is not part of the flooding topology MUST trigger the flooding topology recalculation. This is done to ensure that all nodes converge to the same flooding topology, regardless of the time of the calculation.

6.7.5. Failures of Link On the Flooding Topology

If there is a failure on the flooding topology, the adjacent nodes will update their link state advertisements and flood them. If the original flooding topology is bi-connected, the flooding topology should still be connected despite a single failure.

If the failed local link represented the only connection to the flooding topology on the node where the link failed, the node MUST enable temporary flooding on a subset of its local links. This allows the node to send its updated link state advertisement(s) and also keep receiving link state updates from other nodes in the network before the new flooding topology is calculated and distributed (in the case of centralized mode).

In centralized mode, the Area Leader will notice the change in the flooding topology, recompute the flooding topology, and flood it using the new flooding topology.

In distributed mode, all nodes supporting dynamic flooding will notice the change in the topology and recompute the new flooding topology.

6.7.6. Node Deletion

If a node is deleted from the topology, then at least one link is also removed from the topology. The two sections above apply.

6.7.7. Local Link Addition to the Flooding Topology

If the new flooding topology is received in the case of centralized mode, or calculated locally in the case of distributed mode and the local link on the node that was not part of the flooding topology has been added to the flooding topology, the node MUST:

Re-synchronize the Link State Database over the link. This is done using the standard protocol mechanisms. In the case of IS-IS, this results in setting SRM bit for all LSPs on the circuit and sending a complete set of CSNPs. In OSPF, the mechanism specified in [RFC4811] is used.

Make the link part of the flooding topology and start flooding over it

6.7.8. Local Link Deletion from the Flooding Topology

If the new flooding topology is received in the case of centralized mode, or calculated locally in the case of distributed mode and the local link on the node that was part of the flooding topology has been removed from the flooding topology, the node MUST remove the link from the flooding topology.

The node MUST keep flooding on such link for a limited amount of time to allow other nodes to migrate to the new flooding topology.

If the removed local link represented the only connection to the flooding topology on the node, the node MUST enable temporary flooding on a subset of its local links. This allows the node to send its updated link state advertisement(s) and also keep receiving link state updates from other nodes in the network before the new flooding topology is calculated and distributed (in the case of centralized mode).

6.7.9. Treatment of Disconnected Adjacent Nodes

Every time there is a change in the flooding topology a node MUST check if there are any adjacent nodes that are disconnected from the current flooding topology. Temporary flooding MUST be enabled towards a subset of the disconnected nodes.

6.7.10. Failure of the Area Leader

The failure of the Area Leader can be detected by observing that it is no longer reachable. In this case, the Area Leader election process is repeated and a new Area Leader is elected.

In the centralized mode, the new Area Leader will compute a new flooding topology and flood it using the new flooding topology.

As an optimization, applicable to centralized mode, the new Area Leader MAY compute a new flooding topology that has as much in common as possible with the old flooding topology. This will minimize the risk of over-flooding.

In the distributed mode, the new flooding topology will be calculated on all nodes that support the algorithm that is advertised by the new Area Leader. Nodes that do not support the algorithm advertised by the new Area Leader will no longer participate in Dynamic Flooding and will revert to standard flooding.

6.7.11. Recovery from Multiple Failures

In the unlikely event of multiple failures on the flooding topology, it may become partitioned. The nodes that remain active on the edges of the flooding topology partitions will recognize this and will try to repair the flooding topology locally by enabling temporary flooding towards the nodes that they consider disconnected from the flooding topology until a new flooding topology becomes connected again.

Nodes where local failure was detected update their own link state advertisements and flood them on the remainder of the flooding topology.

In centralized mode, the Area Leader will notice the change in the flooding topology, recompute the flooding topology, and flood it using the new flooding topology.

In distributed mode, all nodes that actively participate in Dynamic Flooding will compute the new flooding topology.

Note that this is very different from the area partition because there is still a connected network graph between the nodes in the area. The area may remain connected and forwarding may still be effective.

7. IANA Considerations

7.1. IS-IS

This document requests the following code point from the "sub-TLVs for TLV 242" registry (IS-IS Router CAPABILITY TLV).

Type: TBD1

Description: IS-IS Area Leader Sub-TLV

Reference: This document (Section 5.1.1)

Type: TBD7

Description: IS-IS Dynamic Flooding Sub-TLV

Reference: This document (Section 5.1.2)

This document requests that IANA allocate and assign two code points from the "IS-IS TLV Codepoints" registry. One for each of the following TLVs:

Type: TBD2

Description: IS-IS Area System IDs TLV

Reference: This document (Section 5.1.3)

Type: TBD3

Description: IS-IS Flooding Path TLV

Reference: This document (Section 5.1.4)

Type: TBD9

Description: IS-IS Flooding Request TLV

Reference: This document (Section 5.1.5)

7.2. OSPF

This document requests the following code points from the "OSPF Router Information (RI) TLVs" registry:

Type: TBD4

Description: OSPF Area Leader Sub-TLV

Reference: This document (Section 5.2.1)

Type: TBD8

Description: OSPF Dynamic Flooding Sub-TLV

Reference: This document (Section 5.2.2)

This document requests the following code point from the "Opaque Link-State Advertisements (LSA) Option Types" registry:

Type: TBD5

Description: OSPFv2 Dynamic Flooding Opaque LSA

Reference: This document (Section 5.2.3)

This document requests the following code point from the "OSPFv3 LSA Function Codes" registry:

Type: TBD6

Description: OSPFv3 Dynamic Flooding LSA

Reference: This document (Section 5.2.4)

This document requests a new bit in LLS Type 1 Extended Options and Flags registry:

Bit Position: TBD10

Description: Flooding Request bit

Reference: This document (Section 5.2.7)

7.2.1. OSPF Dynamic Flooding LSA TLVs Registry

This specification also requests one new registry - "OSPF Dynamic Flooding LSA TLVs". New values can be allocated via IETF Review or IESG Approval

The "OSPF Dynamic Flooding LSA TLVs" registry will define top-level TLVs for the OSPFv2 Dynamic Flooding Opaque LSA and OSPFv3 Dynamic Flooding LSAs. It should be added to the "Open Shortest Path First (OSPF) Parameters" registries group.

The following initial values are allocated:

Type: 0

Description: Reserved

Reference: This document

Type: 1

Description: OSPF Area Router IDs TLV

Reference: This document (Section 5.2.5)

Type: 2

Description: OSPF Flooding Path TLV

Reference: This document (Section 5.2.6)

Types in the range 32768-33023 are for experimental use; these will not be registered with IANA, and MUST NOT be mentioned by RFCs.

Types in the range 33024-65535 are not to be assigned at this time. Before any assignments can be made in the 33024-65535 range, there MUST be an IETF specification that specifies IANA Considerations that covers the range being assigned.

7.3. IGP

IANA is requested to set up a registry called "IGP Algorithm Type For Computing Flooding Topology" under an existing "Interior Gateway Protocol (IGP) Parameters" IANA registries.

Values in this registry come from the range 0-255.

The initial values in the IGP Algorithm Type For Computing Flooding Topology registry are:

0: Reserved for centralized mode. Individual values are are to be assigned according to the "Specification Required" policy defined in [RFC8126]

1-127: Available for standards action.Individual values are are to be assigned according to the "Private Use" policy defined in [RFC8126]

128-254: Reserved for private use.

255: Reserved.

8. Security Considerations

This document introduces no new security issues. Security of routing within a domain is already addressed as part of the routing protocols themselves. This document proposes no changes to those security architectures.

It is possible that an attacker could become Area Leader and introduce a flawed flooding algorithm into the network thus compromising the operation of the protocol. Authentication methods as describe in [RFC5304] and [RFC5310] for IS-IS, [RFC2328] and [RFC7474] for OSPFv2 and [RFC5340] and [RFC4552] for OSPFv3 SHOULD be used to prevent such attack.

9. Acknowledgements

The authors would like to thank Sarah Chen for her contribution to this work.

The authors would like to thank Zeqing (Fred) Xia, Naiming Shen, Adam Sweeney and Olufemi Komolafe for their helpful comments.

The authors would like to thank Tom Edsall for initially introducing them to the problem.

10. References

10.1. Normative References

[ISO10589] International Organization for Standardization, "Intermediate System to Intermediate System Intra-Domain Routing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, Nov. 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC4552] Gupta, M. and N. Melam, "Authentication/Confidentiality for OSPFv3", RFC 4552, DOI 10.17487/RFC4552, June 2006, <https://www.rfc-editor.org/info/rfc4552>.

[RFC5250] Berger, L., Bryskin, I., Zinin, A., and R. Coltun, "The OSPF Opaque LSA Option", RFC 5250, DOI 10.17487/RFC5250, July 2008, <https://www.rfc-editor.org/info/rfc5250>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC5613] Zinin, A., Roy, A., Nguyen, L., Friedman, B., and D. Yeung, "OSPF Link-Local Signaling", RFC 5613, DOI 10.17487/RFC5613, August 2009, <https://www.rfc-editor.org/info/rfc5613>.

[RFC7120] Cotton, M., "Early IANA Allocation of Standards Track Code Points", BCP 100, RFC 7120, DOI 10.17487/RFC7120, January 2014, <https://www.rfc-editor.org/info/rfc7120>.

[RFC7356] Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, DOI 10.17487/RFC7356, September 2014, <https://www.rfc-editor.org/info/rfc7356>.

[RFC7474] Bhatia, M., Hartman, S., Zhang, D., and A. Lindem, Ed., "Security Extension for OSPFv2 When Using Manual Key Management", RFC 7474, DOI 10.17487/RFC7474, April 2015, <https://www.rfc-editor.org/info/rfc7474>.

[RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 7770, DOI 10.17487/RFC7770, February 2016, <https://www.rfc-editor.org/info/rfc7770>.

[RFC7981] Ginsberg, L., Previdi, S., and M. Chen, "IS-IS Extensions for Advertising Router Information", RFC 7981, DOI 10.17487/RFC7981, October 2016, <https://www.rfc-editor.org/info/rfc7981>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

10.2. Informative References

[Clos] Clos, C., "A Study of Non-Blocking Switching Networks", The Bell System Technical Journal Vol. 32(2), DOI 10.1002/j.1538-7305.1953.tb01433.x, March 1953, <http://dx.doi.org/10.1002/j.1538-7305.1953.tb01433.x>.

[Leiserson] Leiserson, C., "Fat-Trees: Universal Networks for Hardware-Efficient Supercomputing", IEEE Transactions on Computers 34(10):892-901, 1985.

[RFC2973] Balay, R., Katz, D., and J. Parker, "IS-IS Mesh Groups", RFC 2973, DOI 10.17487/RFC2973, October 2000, <https://www.rfc-editor.org/info/rfc2973>.

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <https://www.rfc-editor.org/info/rfc3630>.

[RFC4811] Nguyen, L., Roy, A., and A. Zinin, "OSPF Out-of-Band Link State Database (LSDB) Resynchronization", RFC 4811, DOI 10.17487/RFC4811, March 2007, <https://www.rfc-editor.org/info/rfc4811>.

[RFC7938] Lapukhov, P., Premji, A., and J. Mitchell, Ed., "Use of BGP for Routing in Large-Scale Data Centers", RFC 7938, DOI 10.17487/RFC7938, August 2016, <https://www.rfc-editor.org/info/rfc7938>.

Authors' Addresses

Tony Li (editor)
Arista Networks
5453 Great America Parkway
Santa Clara, California 95054
USA

Email: tony.li@tony.li

Peter Psenak (editor)
Cisco Systems, Inc.
Eurovea Centre, Central 3
Pribinova Street 10
Bratislava 81109
Slovakia

Email: ppsenak@cisco.com

Les Ginsberg
Cisco Systems, Inc.
510 McCarthy Blvd.
Milpitas, California 95035
USA

Email: ginsberg@cisco.com

Tony Przygienda
Juniper Networks, Inc.
1194 N. Mathilda Ave
Sunnyvale, California 94089
USA

Email: prz@juniper.net

Dave Cooper
CenturyLink
1025 Eldorado Blvd
Broomfield, Colorado 80021
USA

Email: Dave.Cooper@centurylink.com

Luay Jalil
Verizon
Richardson, Texas 75081
USA

Email: luay.jalil@verizon.com

Srinath Dontula
ATT
200 S Laurel Ave
Middletown, New Jersey 07748
USA

Email: sd947e@att.com

Networking Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 19, 2019

N. Shen
L. Ginsberg
Cisco Systems
S. Thyamagundalu

October 16, 2018

IS-IS Routing for Spine-Leaf Topology
draft-shen-isis-spine-leaf-ext-07

Abstract

This document describes a mechanism for routers and switches in a Spine-Leaf type topology to have non-reciprocal Intermediate System to Intermediate System (IS-IS) routing relationships between the leafs and spines. The leaf nodes do not need to have the topology information of other nodes and exact prefixes in the network. This extension also has application in the Internet of Things (IoT).

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 19, 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 (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

The IS-IS routing protocol defined by [ISO10589] has been widely deployed in provider networks, data centers and enterprise campus environments. In the data center and enterprise switching networks, a Spine-Leaf topology is commonly used. This document describes a mechanism where IS-IS routing can be optimized for a Spine-Leaf topology.

In a Spine-Leaf topology, normally a leaf node connects to a number of spine nodes. Data traffic going from one leaf node to another leaf node needs to pass through one of the spine nodes. Also, the decision to choose one of the spine nodes is usually part of equal cost multi-path (ECMP) load sharing. The spine nodes can be considered as gateway devices to reach destinations on other leaf nodes. In this type of topology, the spine nodes have to know the topology and routing information of the entire network, but the leaf nodes only need to know how to reach the gateway devices to which are the spine nodes they are uplinked.

This document describes the IS-IS Spine-Leaf extension that allows the spine nodes to have all the topology and routing information, while keeping the leaf nodes free of topology information other than the default gateway routing information. The leaf nodes do not even need to run a Shortest Path First (SPF) calculation since they have no topology information.

1.1. 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 RFC 2119 [RFC2119].

2. Motivations

- The leaf nodes in a Spine-Leaf topology do not require complete topology and routing information of the entire domain since their forwarding decision is to use ECMP with spine nodes as default gateways

- The spine nodes in a Spine-Leaf topology are richly connected to leaf nodes, which introduces significant flooding duplication if they flood all Link State PDUs (LSPs) to all the leaf nodes. It saves both spine and leaf nodes' CPU and link bandwidth resources if flooding is blocked to leaf nodes. For small Top of the Rack (ToR) leaf switches in data centers, it is meaningful to prevent full topology routing information and massive database flooding through those devices.

- When a spine node advertises a topology change, every leaf node connected to it will flood the update to all the other spine nodes, and those spine nodes will further flood them to all the leaf nodes, causing a O(n^2) flooding storm which is largely redundant.

- Similar to some of the overlay technologies which are popular in data centers, the edge devices (leaf nodes) may not need to contain all the routing and forwarding information on the device's control and forwarding planes. "Conversational Learning" can be utilized to get the specific routing and forwarding information in the case of pure CLOS topology and in the events of link and node down.

- Small devices and appliances of Internet of Things (IoT) can be considered as leafs in the routing topology sense. They have CPU and memory constrains in design, and those IoT devices do not have to know the exact network topology and prefixes as long as there are ways to reach the cloud servers or other devices.

3. Spine-Leaf (SL) Extension

3.1. Topology Examples

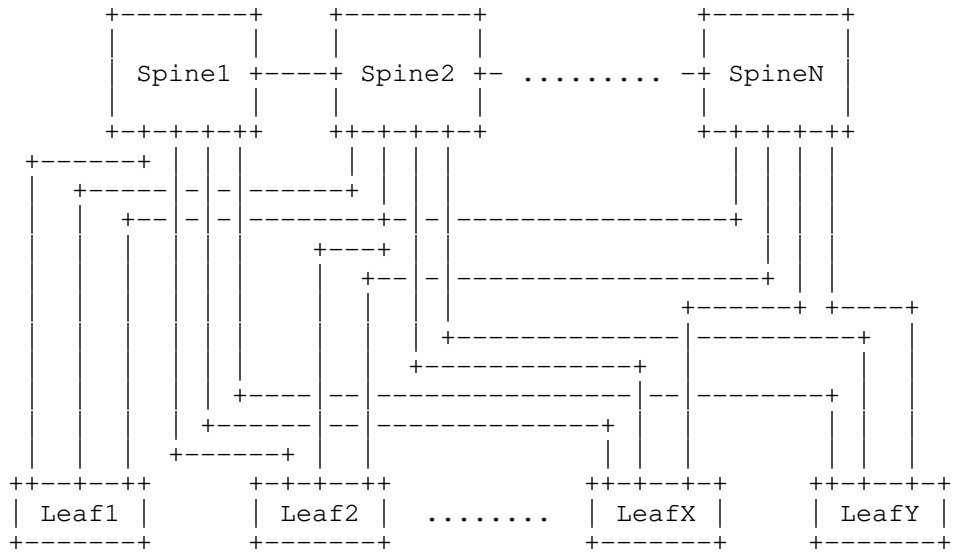


Figure 1: A Spine-Leaf Topology

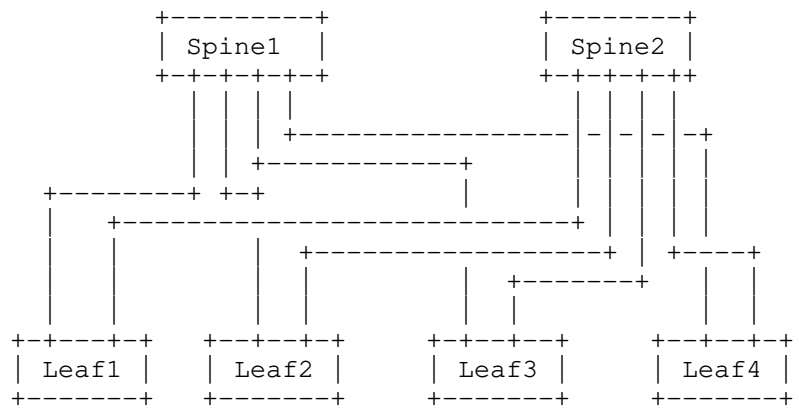


Figure 2: A CLOS Topology

3.2. Applicability Statement

This extension assumes the network is a Spine-Leaf topology, and it should not be applied in an arbitrary network setup. The spine nodes can be viewed as the aggregation layer of the network, and the leaf nodes as the access layer of the network. The leaf nodes use a load sharing algorithm with spine nodes as nexthops in routing and forwarding.

This extension works when the spine nodes are inter-connected, and it works with a pure CLOS or Fat Tree topology based network where the spines are NOT horizontally interconnected.

Although the example diagram in Figure 1 shows a fully meshed Spine-Leaf topology, this extension also works in the case where they are partially meshed. For instance, leaf1 through leaf10 may be fully meshed with spine1 through spine5 while leaf11 through leaf20 is fully meshed with spine4 through spine8, and all the spines are inter-connected in a redundant fashion.

This extension can also work in multi-level spine-leaf topology. The lower level spine node can be a 'leaf' node to the upper level spine node. A spine-leaf 'Tier' can be exchanged with IS-IS hello packets to allow tier X to be connected with tier X+1 using this extension. Normally tier-0 will be the TOR routers and switches if provisioned.

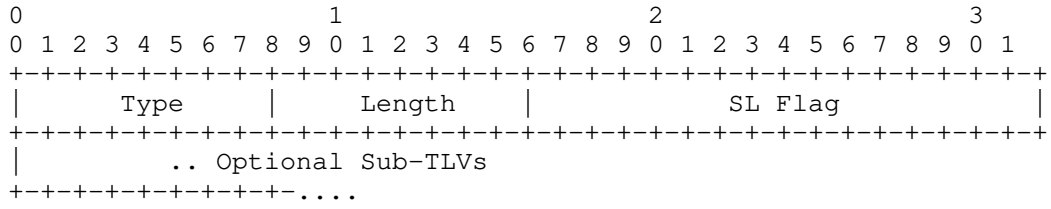
This extension also works with normal IS-IS routing in a topology with more than two layers of spine and leaf. For instance, in example diagrams Figure 1 and Figure 2, there can be another Core layer of routers/switches on top of the aggregation layer. From an IS-IS routing point of view, the Core nodes are not affected by this

extension and will have the complete topology and routing information just like the spine nodes. To make the network even more scalable, the Core layer can operate as a level-2 IS-IS sub-domain while the Spine and Leaf layers operate as stays at the level-1 IS-IS domain.

This extension assumes the link between the spine and leaf nodes are point-to-point, or point-to-point over LAN [RFC5309]. The links connecting among the spine nodes or the links between the leaf nodes can be any type.

3.3. Spine-Leaf TLV

This extension introduces a new TLV, the Spine-Leaf TLV, which may be advertised in IS-IS Hello (IIH) PDUs, LSPs, or in Circuit Scoped Link State PDUs (CS-LSP) [RFC7356]. It is used by both spine and leaf nodes in this Spine-Leaf mechanism.

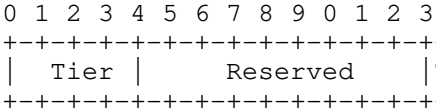


The fields of this TLV are defined as follows:

Type: 1 octet Suggested value 150 (to be assigned by IANA)

Length: 1 octet (2 + length of sub-TLVs).

SL Flags: 16 bits



Tier: A value from 0 to 15. It represents the spine-leaf tier level. The value 15 is reserved to indicate the tier level is unknown. This value is only valid when the 'T' bit (see below) is set. If the 'T' bit is clear, this value MUST be set to zero on transmission, and it MUST be ignored on receipt.

L bit (0x01): Only leaf node sets this bit. If the L bit is set in the SL flag, the node indicates it is in 'Leaf-Mode'.

R bit (0x02): Only Spine node sets this bit. If the R bit is set, the node indicates to the leaf neighbor that it can be used as the default route gateway.

T bit (0x04): If set, the value in the "Tier" field (see above) is valid.

Optional Sub-TLV: Not defined in this document, for future extension

sub-TLVs MAY be included when the TLV is in a CS-LSP. sub-TLVs MUST NOT be included when the TLV is in an IIH

3.3.1. Spine-Leaf Sub-TLVs

If the data center topology is a pure CLOS or Fat Tree, there are no link connections among the spine nodes. If we also assume there is not another Core layer on top of the aggregation layer, then the traffic from one leaf node to another may have a problem if there is a link outage between a spine node and a leaf node. For instance, in the diagram of Figure 2, if Leaf1 sends data traffic to Leaf3 through Spine1 node, and the Spine1-Leaf3 link is down, the data traffic will be dropped on the Spine1 node.

To address this issue spine and leaf nodes may send/request specific reachability information via the sub-TLVs defined below.

Two Spine-Leaf sub-TLVs are defined. The Leaf-Set sub-TLV and the Info-Req sub-TLV.

3.3.1.1. Leaf-Set Sub-TLV

This sub-TLV is used by spine nodes to optionally advertise Leaf neighbors to other Leaf nodes. The fields of this sub-TLV are defined as follows:

Type: 1 octet Suggested value 1 (to be assigned by IANA)

Length: 1 octet MUST be a multiple of 6 octets.

Leaf-Set: A list of IS-IS System-ID of the leaf node neighbors of this spine node.

3.3.1.2. Info-Req Sub-TLV

This sub-TLV is used by leaf nodes to request the advertisement of more specific prefix information from a selected spine node. The list of leaf nodes in this sub-TLV reflects the current set of leaf-nodes for which not all spine node neighbors have indicated the presence of connectivity in the Leaf-Set sub-TLV (See Section 3.3.1.1). The fields of this sub-TLV are defined as follows:

Type: 1 octet Suggested value 2 (to be assigned by IANA)

Length: 1 octet. It MUST be a multiple of 6 octets.

Info-Req: List of IS-IS System-IDs of leaf nodes for which connectivity information is being requested.

3.3.2. Advertising IPv4/IPv6 Reachability

In cases where connectivity between a leaf node and a spine node is down, the leaf node MAY request reachability information from a spine node as described in Section 3.3.1.2. The spine node utilizes TLVs 135 [RFC5305] and TLVs 236 [RFC5308] to advertise this information. These TLVs MAY be included either in IIHs or CS-LSPs [RFC7356] sent from the spine to the requesting leaf node. Sending such information in IIHs has limited scale - all reachability information MUST fit within a single IIH. It is therefore recommended that CS-LSPs be used.

3.3.3. Advertising Connection to RF-Leaf Node

For links between Spine and Leaf Nodes on which the Spine Node has set the R-bit and the Leaf node has set the L-bit in their respective Spine-Leaf TLVs, spine nodes may advertise the link with a bit in the "link-attribute" sub-TLV [RFC5029] to express this link is not used for LSP flooding. This information can be used by nodes computing a flooding topology e.g., [DYNAMIC-FLOODING], to exclude the RF-Leaf nodes from the computed flooding topology.

3.4. Mechanism

Leaf nodes in a spine-leaf application using this extension are provisioned with two attributes:

1)Tier level of 0. This indicates the node is a Leaf Node. The value 0 is advertised in the Tier field of Spine-Leaf TLV defined above.

2)Flooding reduction enabled/disabled. If flooding reduction is enabled the L-bit is set to one in the Spine-Leaf TLV defined above

A spine node does not need explicit configuration. Spine nodes can dynamically discover their tier level by computing the number of hops to a leaf node. Until a spine node determines its tier level it MUST advertise level 15 (unknown tier level) in the Spine-Leaf TLV defined above. Each tier level can also be statically provisioned on the node.

When a spine node receives an IIH which includes the Spine-Leaf TLV with Tier level 0 and 'L' bit set, it labels the point-to-point interface and adjacency to be a 'Reduced Flooding Leaf-Peer (RF-Leaf)'. IIHs sent by a spine node on a link to an RF-Leaf include the Spine-Leaf TLV with the 'R' bit set in the flags field. The 'R' bit indicates to the RF-Leaf neighbor that the spine node can be used as a default routing nexthop.

There is no change to the IS-IS adjacency bring-up mechanism for Spine-Leaf peers.

A spine node blocks LSP flooding to RF-Leaf adjacencies, except for the LSP PDUs in which the IS-IS System-ID matches the System-ID of the RF-Leaf neighbor. This exception is needed since when the leaf node reboots, the spine node needs to forward to the leaf node non-purged LSPs from the RF-Leaf's previous incarnation.

Leaf nodes will perform IS-IS LSP flooding as normal over all of its IS-IS adjacencies, but in the case of RF-Leafs only self-originated LSPs will exist in its LSP database.

Spine nodes will receive all the LSP PDUs in the network, including all the spine nodes and leaf nodes. It will perform Shortest Path First (SPF) as a normal IS-IS node does. There is no change to the route calculation and forwarding on the spine nodes.

The LSPs of a node only floods north bound towards the upper layer spine nodes. The default route is generated with loadsharing also towards the upper layer spine nodes.

RF-Leaf nodes do not have any LSP in the network except for its own. Therefore there is no need to perform SPF calculation on the RF-Leaf node. It only needs to download the default route with the nexthops of those Spine Neighbors which have the 'R' bit set in the Spine-Leaf TLV in IIH PDUs. IS-IS can perform equal cost or unequal cost load sharing while using the spine nodes as nexthops. The aggregated metric of the outbound interface and the 'Reverse Metric' [REVERSE-METRIC] can be used for this purpose.

3.4.1. Pure CLOS Topology

In a data center where the topology is pure CLOS or Fat Tree, there is no interconnection among the spine nodes, and there is not another Core layer above the aggregation layer with reachability to the leaf nodes. When flooding reduction to RF-Leafs is in use, if the link between a spine and a leaf goes down, there is then a possibility of black holing the data traffic in the network.

As in the diagram Figure 2, if the link Spine1-Leaf3 goes down, there needs to be a way for Leaf1, Leaf2 and Leaf4 to avoid the Spine1 if the destination of data traffic is to Leaf3 node.

In the above example, the Spine1 and Spine2 are provisioned to advertise the Leaf-Set sub-TLV of the Spine-Leaf TLV. Originally both Spines will advertise Leaf1 through Leaf4 as their Leaf-Set. When the Spine1-Leaf3 link is down, Spine1 will only have Leaf1, Leaf2 and Leaf4 in its Leaf-Set. This allows the other leaf nodes to know that Spine1 has lost connectivity to the leaf node of Leaf3.

Each RF-Leaf node can select another spine node to request for some prefix information associated with the lost leaf node. In this diagram of Figure 2, there are only two spine nodes (Spine-Leaf topology can have more than two spine nodes in general). Each RF-Leaf node can independently select a spine node for the leaf information. The RF-Leaf nodes will include the Info-Req sub-TLV in the Spine-Leaf TLV in hellos sent to the selected spine node, Spine2 in this case.

The spine node, upon receiving the request from one or more leaf nodes, will find the IPv6/IPv4 prefixes advertised by the leaf nodes listed in the Info-Req sub-TLV. The spine node will use the mechanism defined in Section 3.3.2 to advertise these prefixes to the RF-Leaf node. For instance, it will include the IPv4 loopback prefix of leaf3 based on the policy configured or administrative tag attached to the prefixes. When the leaf nodes receive the more specific prefixes, they will install the advertised prefixes towards the other spine nodes (Spine2 in this example).

For instance in the data center overlay scenario, when any IP destination or MAC destination uses the leaf3’s loopback as the tunnel nexthop, the overlay tunnel from leaf nodes will only select Spine2 as the gateway to reach leaf3 as long as the Spine1-Leaf3 link is still down.

In cases where multiple links or nodes fail at the same time, the RF-leaf node may need to send the Info-Req to multiple upper layer spine

nodes in order to obtain reachability information for all the partially connected nodes.

This negative routing is more useful between tier 0 and tier 1 spine-leaf levels in a multi-level spine-leaf topology when the reduced flooding extension is in use. Nodes in tiers 1 or greater may have much richer topology information and alternative paths.

3.5. Implementation and Operation

3.5.1. CSNP PDU

In Spine-Leaf extension, Complete Sequence Number PDU (CSNP) does not need to be transmitted over the Spine-Leaf link to an RF-Leaf. Some IS-IS implementations send periodic CSNPs after the initial adjacency bring-up over a point-to-point interface. There is no need for this optimization here since the RF-Leaf does not need to receive any other LSPs from the network, and the only LSPs transmitted across the Spine-Leaf link is the leaf node LSP.

Also in the graceful restart case[RFC5306], for the same reason, there is no need to send the CSNPs over the Spine-Leaf interface to an RF-Leaf. Spine nodes only need to set the SRMflag on the LSPs belonging to the RF-Leaf.

3.5.2. Overload Bit

The leaf node SHOULD set the 'overload' bit on its LSP PDU, since if the spine nodes were to forward traffic not meant for the local node, the leaf node does not have the topology information to prevent a routing/forwarding loop.

3.5.3. Spine Node Hostname

This extension creates a non-reciprocal relationship between the spine node and leaf node. The spine node will receive leaf's LSP and will know the leaf's hostname, but the leaf does not have spine's LSP. This extension allows the Dynamic Hostname TLV [RFC5301] to be optionally included in spine's IIH PDU when sending to a 'Leaf-Peer'. This is useful in troubleshooting cases.

3.5.4. IS-IS Reverse Metric

This metric is part of the aggregated metric for leaf's default route installation with load sharing among the spine nodes. When a spine node is in 'overload' condition, it should use the IS-IS Reverse Metric TLV in IIH [REVERSE-METRIC] to set this metric to maximum to discourage the leaf using it as part of the loadsharing.

In some cases, certain spine nodes may have less bandwidth in link provisioning or in real-time condition, and it can use this metric to signal to the leaf nodes dynamically.

In other cases, such as when the spine node loses a link to a particular leaf node, although it can redirect the traffic to other spine nodes to reach that destination leaf node, but it MAY want to increase this metric value if the inter-spine connection becomes over utilized, or the latency becomes an issue.

In the leaf-leaf link as a backup gateway use case, the 'Reverse Metric' SHOULD always be set to very high value.

3.5.5. Spine-Leaf Traffic Engineering

Besides using the IS-IS Reverse Metric by the spine nodes to affect the traffic pattern for leaf default gateway towards multiple spine nodes, the IPv6/IPv4 Info-Advertise sub-TLVs can be selectively used by traffic engineering controllers to move data traffic around the data center fabric to alleviate congestion and to reduce the latency of a certain class of traffic pairs. By injecting more specific leaf node prefixes, it will allow the spine nodes to attract more traffic on some underutilized links.

3.5.6. Other End-to-End Services

Losing the topology information will have an impact on some of the end-to-end network services, for instance, MPLS TE or end-to-end segment routing. Some other mechanisms such as those described in PCE [RFC4655] based solution may be used. In this Spine-Leaf extension, the role of the leaf node is not too much different from the multi-level IS-IS routing while the level-1 IS-IS nodes only have the default route information towards the node which has the Attach Bit (ATT) set, and the level-2 backbone does not have any topology information of the level-1 areas. The exact mechanism to enable certain end-to-end network services in Spine-Leaf network is outside the scope of this document.

3.5.7. Address Family and Topology

IPv6 Address families[RFC5308], Multi-Topology (MT)[RFC5120] and Multi-Instance (MI)[RFC8202] information is carried over the IIH PDU. Since the goal is to simplify the operation of IS-IS network, for the simplicity of this extension, the Spine-Leaf mechanism is applied the same way to all the address families, MTs and MIs.

3.5.8. Migration

For this extension to be deployed in existing networks, a simple migration scheme is needed. To support any leaf node in the network, all the involved spine nodes have to be upgraded first. So the first step is to migrate all the involved spine nodes to support this extension, then the leaf nodes can be enabled with 'Leaf-Mode' one by one. No flag day is needed for the extension migration.

4. IANA Considerations

A new TLV codepoint is defined in this document and needs to be assigned by IANA from the "IS-IS TLV Codepoints" registry. It is referred to as the Spine-Leaf TLV and the suggested value is 150. This TLV is only to be optionally inserted either in the IIH PDU or in the Circuit Flooding Scoped LSP PDU. IANA is also requested to maintain the SL-flag bit values in this TLV, and 0x01, 0x02 and 0x04 bits are defined in this document.

Value	Name	IIH	LSP	SNP	Purge	CS-LSP
150	Spine-Leaf	y	y	n	n	y

This extension also proposes to have the Dynamic Hostname TLV, already assigned as code 137, to be allowed in IIH PDU.

Value	Name	IIH	LSP	SNP	Purge
137	Dynamic Name	y	y	n	y

Two new sub-TLVs are defined in this document and needs to be added assigned by IANA from the "IS-IS TLV Codepoints". They are referred to in this document as the Leaf-Set sub-TLV and the Info-Req sub-TLV. It is suggested to have the values 1 and 2 respectively.

This document also requests that IANA allocate from the registry of link-attribute bit values for sub-TLV 19 of TLV 22 (Extended IS reachability TLV). This new bit is referred to as the "Connect to RF-Leaf Node" bit.

Value	Name	Reference
0x3	Connect to RF-Leaf Node	This document

5. Security Considerations

Security concerns for IS-IS are addressed in [ISO10589], [RFC5304], [RFC5310], and [RFC7602]. This extension does not raise additional security issues.

6. Acknowledgments

The authors would like to thank Tony Przygienda for his discussion and contributions. The authors also would like to thank Acee Lindem, Russ White and Christian Hopps for their review and comments of this document.

7. Document Change Log

7.1. Changes to draft-shen-isis-spine-leaf-ext-05.txt

- Submitted January 2018.
- Just a refresh.

7.2. Changes to draft-shen-isis-spine-leaf-ext-04.txt

- Submitted June 2017.
- Added the Tier level information to handle the multi-level spine-leaf topology using this extension.

7.3. Changes to draft-shen-isis-spine-leaf-ext-03.txt

- Submitted March 2017.
- Added the Spine-Leaf sub-TLVs to handle the case of data center pure CLOS topology and mechanism.
- Added the Spine-Leaf TLV and sub-TLVs can be optionally inserted in either IIH PDU or CS-LSP PDU.
- Allow use of prefix Reachability TLVs 135 and 236 in IIHs/CS-LSPs sent from spine to leaf.

7.4. Changes to draft-shen-isis-spine-leaf-ext-02.txt

- Submitted October 2016.
- Removed the 'Default Route Metric' field in the Spine-Leaf TLV and changed to using the IS-IS Reverse Metric in IIH.

7.5. Changes to draft-shen-isis-spine-leaf-ext-01.txt

- Submitted April 2016.
- No change. Refresh the draft version.

7.6. Changes to draft-shen-isis-spine-leaf-ext-00.txt

- Initial version of the draft is published in November 2015.

8. References

8.1. Normative References

[ISO10589] ISO "International Organization for Standardization", "Intermediate system to Intermediate system intra-domain routeing 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.

[REVERSE-METRIC] Shen, N., Amante, S., and M. Abrahamsson, "IS-IS Routing with Reverse Metric", draft-ietf-isis-reverse-metric-07 (work in progress), 2017.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5029] Vasseur, JP. and S. Previdi, "Definition of an IS-IS Link Attribute Sub-TLV", RFC 5029, DOI 10.17487/RFC5029, September 2007, <https://www.rfc-editor.org/info/rfc5029>.

[RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <https://www.rfc-editor.org/info/rfc5120>.

[RFC5301] McPherson, D. and N. Shen, "Dynamic Hostname Exchange Mechanism for IS-IS", RFC 5301, DOI 10.17487/RFC5301, October 2008, <https://www.rfc-editor.org/info/rfc5301>.

[RFC5304] Li, T. and R. Atkinson, "IS-IS Cryptographic Authentication", RFC 5304, DOI 10.17487/RFC5304, October 2008, <https://www.rfc-editor.org/info/rfc5304>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5306] Shand, M. and L. Ginsberg, "Restart Signaling for IS-IS", RFC 5306, DOI 10.17487/RFC5306, October 2008, <https://www.rfc-editor.org/info/rfc5306>.

[RFC5308] Hopps, C., "Routing IPv6 with IS-IS", RFC 5308, DOI 10.17487/RFC5308, October 2008, <https://www.rfc-editor.org/info/rfc5308>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, DOI 10.17487/RFC5310, February 2009, <https://www.rfc-editor.org/info/rfc5310>.

[RFC7356] Ginsberg, L., Previdi, S., and Y. Yang, "IS-IS Flooding Scope Link State PDUs (LSPs)", RFC 7356, DOI 10.17487/RFC7356, September 2014, <https://www.rfc-editor.org/info/rfc7356>.

[RFC7602] Chunduri, U., Lu, W., Tian, A., and N. Shen, "IS-IS Extended Sequence Number TLV", RFC 7602, DOI 10.17487/RFC7602, July 2015, <https://www.rfc-editor.org/info/rfc7602>.

[RFC8202] Ginsberg, L., Previdi, S., and W. Henderickx, "IS-IS Multi-Instance", RFC 8202, DOI 10.17487/RFC8202, June 2017, <https://www.rfc-editor.org/info/rfc8202>.

8.2. Informative References

[DYNAMIC-FLOODING] Li, T., "Dynamic Flooding on Dense Graphs", draft-li-dynamic-flooding (work in progress), 2018.

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

[RFC5309] Shen, N., Ed. and A. Zinin, Ed., "Point-to-Point Operation over LAN in Link State Routing Protocols", RFC 5309, DOI 10.17487/RFC5309, October 2008, <https://www.rfc-editor.org/info/rfc5309>.

Authors' Addresses

Naiming Shen
Cisco Systems
560 McCarthy Blvd.
Milpitas, CA 95035
US

Email: naiming@cisco.com

Les Ginsberg
Cisco Systems
821 Alder Drive
Milpitas, CA 95035
US

Email: ginsberg@cisco.com

Sanjay Thyamagundalu

Email: tsanjay@gmail.com

LSR Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 18, 2019

A. Wang
China Telecom
A. Lindem
Cisco Systems
J. Dong
Huawei Technologies
K. Talaulikar
P. Psenak
Cisco Systems

October 15, 2018

OSPF Extension for Prefix Originator
draft-wang-lsr-ospf-prefix-originator-ext-00

Abstract

This document describes method to transfer the source router id of inter-area prefixes for OSPFv2 [RFC7684]and OSPFv3 [RFC8362], which is needed in several use cases in OSPF inter-area scenario.

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 April 18, 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

Draft [I-D.ietf-ospf-mpls-elc] defines mechanisms to signal Entropy Label Capability(ELC) and Entropy Readable Label Depth (ERLD) for the ingress LSR to know each LSR's capability of reading the maximum label stack depth and performing EL-based load-balancing in MPLS network. After knowing this information, the ingress LSR can push the appropriate label stack for specific FEC trafffic, especially in segment routing environment or in other stacked LSPs scenarios.

But in OSPF inter-area scenario, the prefixes originator information in another area is omitted by ABR router, all prefixes are attached behind the ABR. Router in one area doesn't know where the prefixes really come from, can't decide the associated router of the inter-area prefixes and then can't judge the ELC and ERLD capabilities of the destination. It is necessary to transfer the originator information of these inter-area prefixes to assist the ingress LSR does the right Label stack action.

More generally, draft [I-D.ietf-ospf-segment-routing-msd] defines the mechanism to advertise multiple types of supported Maximum SID Depths(MSD) at node and/or link granularity. These information will be referred when the head end router starts to send the traffic to destination prefixes. In inter-area scenario, it is also necessary

for the sender to knows the capabilities of the receivers that associated with the inter-area prefixes.

There is also other scenario that the originator of inter-area prefixes are useful. For example, BGP-LS [RFC7752] describes the methodology that using BGP protocol to transfer the Link-State information. Such method can enable SDN controller to collect the underlay network topology automatically.

But if the underlay network is divided into multi area and running OSPF protocol, it is not easy for the SDN controller to rebuild the multi-area topology, because normally the ABR that locates on the boundary of different area will hide the detail topology information in non-backbone area, and the router in backbone area that runs BGP-LS protocol can only get and report the summary network information in non-backbone area. If the SDN controller can get the originator of the inter-area prefixes, it is easy for them to rebuild the inter-area topology automatically.

[RFC7794] introduces "IPv4/IPv6 Source Router IDs" TLV to label the source of the prefixes redistributed from different Level, this TLV can be used in the above scenarios. Such solution can also be applied into network that run OSPF protocol, but the related LSP messages must be extended.

This draft gives such solution for the OSPF v2 and OSPF v3 protocol.

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 [RFC2119] .

3. Scenario Description

Fig.1 illustrates the topology scenario when OSPF is running in multi-area. R0-R4 are routers in backbone area, S1-S4,T1-T4 are internal router in area 1 and area 2 respectively. R1 and R3 are border routers between area 0 and area 1; R2 and R4 are border routers between area 0 and area 2. N1 is the network between router S1 and S2, N2 is the network between router T1 and T2. Ls1 is the loopback address of Node S1, Lt1 is the loopback address of Node T1.

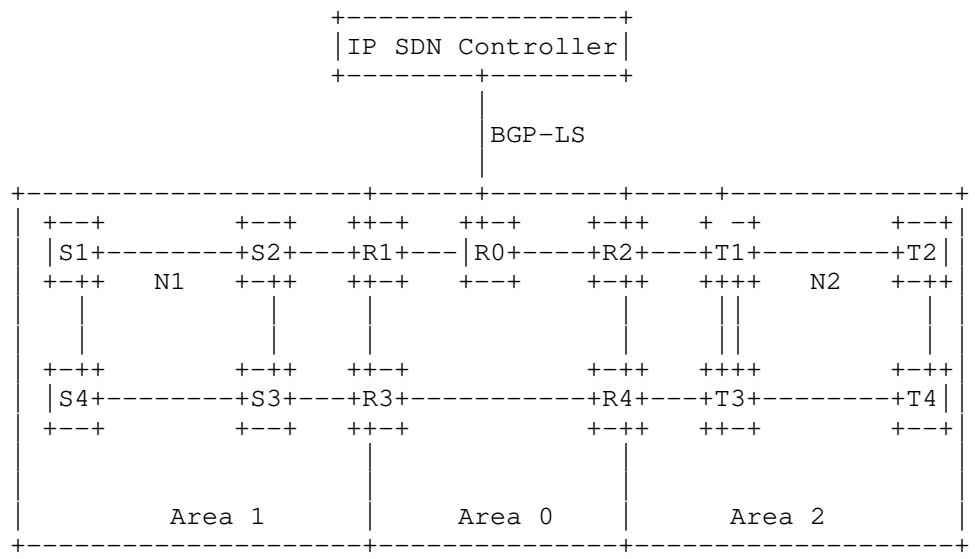


Fig.1 OSPF Inter-Area Prefix Originator Scenario

If S1 want to send traffic to prefix Lt1 that is connected T1 in another area, it should know the ELC, ERLD and MSD values that are associated with the node T1, and then select the right label action at the ingress node for the target traffic.

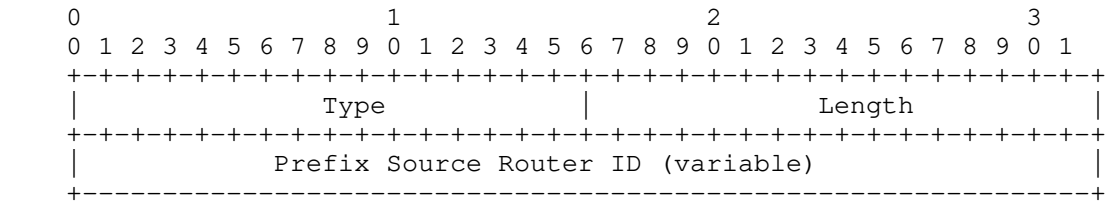
On the other hand, If R0 has some methods to know the originator of network N1 and reports such information to IP SDN controller, then it is possible for the controller to retrieval the topology in non-backbone area. The topology retrieval process and its usage limitation are described in the Appendix A and Appendix B.

From the above scenarios we can conclude it is useful to introduce and define the prefix originator sub TLV within OSPF.

4. Prefix Source Router ID sub TLV

[RFC7684] and [RFC8362] define the TLV format extension for OSPFv2 and OSPFv3 respectively. These documents give the flexibility to add new attributes for the prefixes and links. Based on these formats, we can define new sub TLV to transfer the "Prefix Source Router ID", as that defined in [RFC7794].

The proposed "Prefix Source Router ID" format is the following:



For IPv4 network, it is the following:

- IPv4 Source Router ID Type: TBD
- Length: 4
- Value: IPv4 Router ID of the source of the advertisement

This sub TLV should be included in the "OSPFv2 Extended Prefix Opaque LSA" that defined in [RFC7684]

For IPv6 network, it is the following:

- IPv6 Source Router ID Type: TBD
- Length: 16
- Value: IPv6 Router ID of the source of the advertisement

This sub TLV should be included in "E-Inter-Area-Prefix-LSA" that defined in [RFC8362]

5. Extend LSA Operation Procedure

When ABR(for example R2 in Fig.1)receives the "Router LSA" announcement in area 2, it should generate the corresponding "OSPFv2 Extended Prefix Opaque LSA" for OSPFv2 or "E-Inter-Area-Prefix-LSA" for OSPFv3 that includes the sub TLV "Source Router ID" of the network prefixes(for example, for prefix Lt1, N2 etc.), which labels the source router of the corresponding link.

When S1 in another area receives such LSA, it then can know the prefix Lt1 is associated with node T1, check the ELC, ERLD or MSD value according to its necessary, and select the right label action at the ingress node S1 for the traffic target to Lt1.

When R0 receives such LSA, it then strips this additional information, put it into the corresponding part in BGP-LS protocol as described in[I-D.ietf-idr-bgp-ls-segment-routing-ext] and reports them to the IP SDN Controller, the SDN controller can then use such

information to build the inter-area topology according to the process described in the Appendix A. The topology retrieval process may not suitable for some special environment as that stated in Appendix B

6. Security Considerations

TBD.

7. IANA Considerations

TBD.

8. Acknowledgement

Very thanks Les Ginsberg for their valuable suggestions on the contents of this draft. And also thanks Jeff Tantsura, Rob Shakir for their valuable comments on this draft.

9. References

9.1. Normative References

[I-D.ietf-idr-bgp-ls-segment-routing-ext]
Previdi, S., Talaulikar, K., Filsfils, C., Gredler, H., and M. Chen, "BGP Link-State extensions for Segment Routing", draft-ietf-idr-bgp-ls-segment-routing-ext-08 (work in progress), May 2018.

[I-D.ietf-ospf-mpls-elc]
Xu, X., Kini, S., Sivabalan, S., Filsfils, C., and S. Litkowski, "Signaling Entropy Label Capability and Entropy Readable Label-stack Depth Using OSPF", draft-ietf-ospf-mpls-elc-07 (work in progress), September 2018.

[I-D.ietf-ospf-segment-routing-msd]
Tantsura, J., Chunduri, U., Aldrin, S., and P. Psenak, "Signaling MSD (Maximum SID Depth) using OSPF", draft-ietf-ospf-segment-routing-msd-23 (work in progress), October 2018.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC7684] Psenak, P., Gredler, H., Shakir, R., Henderickx, W., Tantsura, J., and A. Lindem, "OSPFv2 Prefix/Link Attribute Advertisement", RFC 7684, DOI 10.17487/RFC7684, November 2015, <https://www.rfc-editor.org/info/rfc7684>.

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

[RFC7794] Ginsberg, L., Ed., Decraene, B., Previdi, S., Xu, X., and U. Chunduri, "IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability", RFC 7794, DOI 10.17487/RFC7794, March 2016, <https://www.rfc-editor.org/info/rfc7794>.

[RFC8362] Lindem, A., Roy, A., Goethals, D., Reddy Vallem, V., and F. Baker, "OSPFv3 Link State Advertisement (LSA) Extensibility", RFC 8362, DOI 10.17487/RFC8362, April 2018, <https://www.rfc-editor.org/info/rfc8362>.

9.2. Informative References

[I-D.wang-idr-bgpls-inter-as-topology-ext] Wang, A. and H. Chen, "BGP-LS Extension for Inter-AS Topology Retrieval", draft-wang-idr-bgpls-inter-as-topology-ext-02 (work in progress), August 2018.

Appendix A. Inter-Area Topology Retrieval Process

When IP SDN Controller receives this information, it should compare the prefix NLRI that included in the BGP-LS packet. When it encounters the same prefix but with different source router ID, it should extract the corresponding area ID, rebuild the link between these two different source router in non-backbone area. Belows is one example that based on the Fig.1:

Assuming we want to rebuild the connection between router S1 and router S2 that locates in area 1:

a. Normally, router S1 will advertise prefix N1 within its router LSA

b. When this router LSA reaches the ABR router R1, it will convert it into summary LSA, add the "Source Router Information", which is router id of S1 in this example, as proposed in this draft.

c. R1 then floods this extension summary LSA to R0, which is running BGP-LS protocol with IP SDN Controller. The controller then knows the prefixes of N1 is from S1.

d. Router S2 will do the similar process, and the controller will also knows the prefixes N1 is also from S2

e. Then it can reconstruct the connection between S1 and S2, which prefix is N1. The topology within Area 1 can then be recovered accordingly.

Iterating the above process continuously, the IP SDN controller can then retrieve the detail topology that span multi-area.

Appendix B. Special Considerations on Inter-Area Topology Retrieval

The above topology retrieval process can be applied in general situation, where each prefix of the link between two nodes is planned and allocated in normal space. But there are some situations not belong to this and needs to be considered specially, for example when the link is unnumbered and there are anycast prefixes deployed within the network etc.

When ABR receives the unnumbered LSA, it will not advertise such LSA into another area in the current OSPF specification. Considering such situation is seldom exist in real network, here we will only state explicitly that the above retrieval process is not suitable for the network that deploys unnumbered links.

When there are anycast prefixes deployment within the network, if the anycast prefix length is equal to 32, the controller can bypass them easily because no prefix of the link will use such prefix length. If the ansycast prefixes length is less than 32, it is acceptable that connects the nodes advertising these anycast prefixes logically. Or, if these anycast prefixes come from more than two nodes, the controller can also detect such situation and label it explicitly.

Authors' Addresses

Aijun Wang
China Telecom
Beiqijia Town, Changping District
Beijing 102209
China

Email: wangaj.bri@chinatelecom.cn

Acee Lindem
Cisco Systems
301 Midenhall Way
Cary, NC 27513
USA

Email: acee@cisco.com

Jie Dong
Huawei Technologies
Beijing
China

Email: jie.dong@huawei.com

Ketan Talaulikar
Cisco Systems
S.No. 154/6, Phase I, Hinjawadi
Pune 411 057
India

Email: ketant@cisco.com

Peter Psenak
Cisco Systems
Pribinova Street 10
Bratislava, Eurovea Centre, Central 3 81109
Slovakia

Email: ppsenak@cisco.com

PCE working group
Internet-Draft
Intended status: Standards Track
Expires: February 24, 2019

D. Lopez
Telefonica I+D
Q. Wu
D. Dhody
Z. Wang
Huawei
D. King
Old Dog Consulting

August 23, 2018

IGP extension for PCEP security capability support in the PCE discovery
draft-wu-lsr-pce-discovery-security-support-00

Abstract

When a Path Computation Element (PCE) is a Label Switching Router (LSR) participating in the Interior Gateway Protocol (IGP), or even a server participating in IGP, its presence and path computation capabilities can be advertised using IGP flooding. The IGP extensions for PCE discovery (RFC 5088 and RFC 5089) define a method to advertise path computation capabilities using IGP flooding for OSPF and IS-IS respectively. However these specifications lack a method to advertise PCEP security (e.g., Transport Layer Security(TLS),TCP Authentication Option (TCP-AO)) support capability.

This document proposes new capability flag bits for PCE-CAP-FLAGS sub-TLV that can be announced as attribute in the IGP advertisement to distribute PCEP security support information.

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 February 24, 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.

1. Introduction

As described in [RFC5440], PCEP communication privacy is one importance issue, as an attacker that intercepts a Path Computation Element (PCE) message could obtain sensitive information related to computed paths and resources.

Among the possible solutions mentioned in these documents, Transport Layer Security (TLS) [RFC5246] provides support for peer authentication, and message encryption and integrity while TCP Authentication Option (TCP-AO) [RFC5925] and Cryptographic Algorithms for TCP-AO [RFC5926] offer significantly improved security for applications using TCP. As specified in section 4 of [RFC8253], in order for a Path Computation Client(PCC) to begin a connection with a PCE server using TLS or TCP-AO, PCC SHOULD know whether PCE server supports TLS or TCP-AO as a secure transport.

[RFC5088] and [RFC5089] define a method to advertise path computation capabilities using IGP flooding for OSPF and IS-IS respectively. However [RFC5088] and [RFC5089] lacks a method to advertise PCEP security (e.g., TLS) support capability.

This document proposes new capability flag bits for PCE-CAP-FLAGS sub-TLV that can be announced as attributes in the IGP advertisement (defined in [RFC5088] and [RFC5089]) to distribute PCEP security support information.

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 RFC2119 [RFC2119].

3. IGP extension for PCEP security capability support

The PCE-CAP-FLAGS sub-TLV is defined in section 4.5 of [RFC5088] and [RFC5089] as an optional sub-TLV used to advertise PCE capabilities. In this section, we extend the PCE-CAP-FLAGS sub-TLV to include the capability and indications that are described for PCEP security (e.g., TLS) support in the current document.

In the PCE-CAP-FLAGS sub-TLV defined in [RFC5088] and [RFC5089], nine capability flags defined in [RFC5088] (as per [RFC4657]) and two capability flags defined [RFC5557], [RFC6006] are included and follows the following format:

- TYPE: 5
- LENGTH: Multiple of 4
- VALUE: This contains an array of units of 32 bit flags with the most significant bit as 0. Each bit represents one PCE capability.

and the processing rule of these flag bits are defined in [RFC5088] and [RFC5089]. In this document, we define two new capability flag bits that indicate TCP Authentication Option (TCP-AO) support, PCEP over TLS support respectively as follows:

Bit	Capability Description
xx	TCP AO Support
xx	PCEP over TLS support

3.1. Use of PCEP security capability support for PCE discovery

TCP-AO, PCEP over TLS support flag bits are advertised using IGP flooding.

- PCE supports TCP-AO: IGP advertisement SHOULD include TCP-AO support flag bit.

- PCE supports TLS: IGP advertisement SHOULD include PCEP over TLS support flag bit.

If PCE supports multiple security mechanisms, it SHOULD include all corresponding flag bits in IGP advertisement.

If the client is looking for connecting with PCE server with TCP-AO support, the client MUST check if TCP-AO support flag bit in the PCE-CAP-FLAGS sub-TLV is set. If not, the client SHOULD NOT consider this PCE. If the client is looking for connecting with PCE server using TLS, the client MUST check if PCEP over TLS support flag bit in

the PCE-CAP-FLAGS sub-TLV is set. If not, the client SHOULD NOT consider this PCE.

4. Backward Compatibility Consideration

An LSR that does not support the new IGP PCE capability bits specified in this document silently ignores those bits.

IGP extensions defined in this document do not introduce any new interoperability issues.

5. Management Considerations

A configuration option may be provided for advertising and withdrawing PCE security capability via IGP.

6. Security Considerations

This document raises no new security issues beyond those described in [RFC5088] and [RFC5089].

7. IANA Considerations

IANA is requested to allocate a new bit in "PCE Security Capability Flags" registry for PCEP Security support capability.

Bit	Meaning	Reference
xx	TCP-AO Support	[This.I.D]
xx	PCEP over TLS support	[This.I.D]

8. References

8.1. Normative References

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

[RFC5088] Le Roux, JL., "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, January 2008.

[RFC5089] Le Roux, JL., "IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5089, January 2008.

[RFC5925] Touch , J., "The TCP Authentication Option", RFC 5925, June 2010.

[RFC5926] Gregory Lebovitz, G., "Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)", RFC 5926, June 2010.

[RFC8253] R. Lopez, D., "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, October 2017.

8.2. Informative References

[RFC4657] Ash, J. and J. Le Roux, "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, September 2006.

[RFC5246] Dierks, T., "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5440] Le Roux, JL., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5557] Lee, Y., Le Roux, JL., King, D., and E. Oki, "Path Computation Element Communication Protocol (PCEP) Requirements and Protocol Extensions in Support of Global Concurrent Optimization", RFC 5557, July 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

Appendix A. Appendix A: No MD5 Capability Support

To be compliant with Section 10.2 of RFC5440, this document doesn't consider to add capability for TCP-MD5. Therefore by default, PCEP Speaker in communication supports capability for TCP-MD5 (See section 10.2,[RFC5440]). A method to advertise TCP-MD5 Capability support using IGP flooding is not required. If the client is looking for connecting with PCE server with other Security capability support (e.g., TLS support) than TCP-MD5, the client MUST check if flag bit in the PCE- CAP-FLAGS sub-TLV for specific capability is set (See section 3.1).

Authors' Addresses

Diego R. Lopez
Telefonica I+D
Spain

Email: diego.r.lopez@telefonica.com

Qin Wu
Huawei Technologies
12 Mozhou East Road, Jiangning District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

Email: dhruv.ietf@gmail.com

Michael Wang
Huawei
12 Mozhou East Road, Jiangning District
Nanjing, Jiangsu 210012
China

Email: wangzitao@huawei.com

Daniel King
Old Dog Consulting
UK

Email: daniel@olddog.co.uk