LSVR
Internet-Draft
Intended status: Informational
Expires: April 25, 2019

K. Patel
Arrcus, Inc.
A. Lindem
Cisco Systems
S. Zandi
G. Dawra
Linkedin

October 22, 2018

Usage and Applicability of Link State Vector Routing in Data Centers
draft-ietf-lsvr-applicability-01.txt

Abstract

This document discusses the usage and applicability of Link State Vector Routing (LSVR) extensions in the CLOS architecture of Data Center Networks. The document is intended to provide a simplified guide for the deployment of LSVR extensions.

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 25, 2019.

Copyright Notice

Table of Contents

1. Introduction

This document complements [I-D.ietf-lsvr-bgp-spf] by discussing the applicability of the technology in a simple and fairly common deployment scenario, which is described in Section 4.

After describing the deployment scenario, Section 5 will describe the reasons for BGP modifications for such deployments.

Once the control plane routing protocol requirements are described, Section 6 will cover the LSVR protocol enhancements to BGP to meet these requirements and their applicability to Data Center CLOS networks.

2. 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.

3. Recommended Reading

This document assumes knowledge of existing data center networks and data center network topologies [CLOS]. This document also assumes knowledge of data center routing protocols like BGP [RFC4271], BGP-SPF [I-D.ietf-lsvr-bgp-spf], OSPF [RFC2328], as well as, data center OAM protocols like LLDP [RFC4957] and BFD [RFC5580].

4. Common Deployment Scenario

Within a Data Center, a common network design to interconnect servers is done using the CLOS topology [CLOS]. The CLOS topology is fully non-blocking and the topology is realized using Equal Cost Multipath (ECMP). In a CLOS topology, the minimum number of parallel paths between two servers is determined by the width of a tier-1 stage as shown in the figure 1.

The following example illustrates multistage CLOS topology.

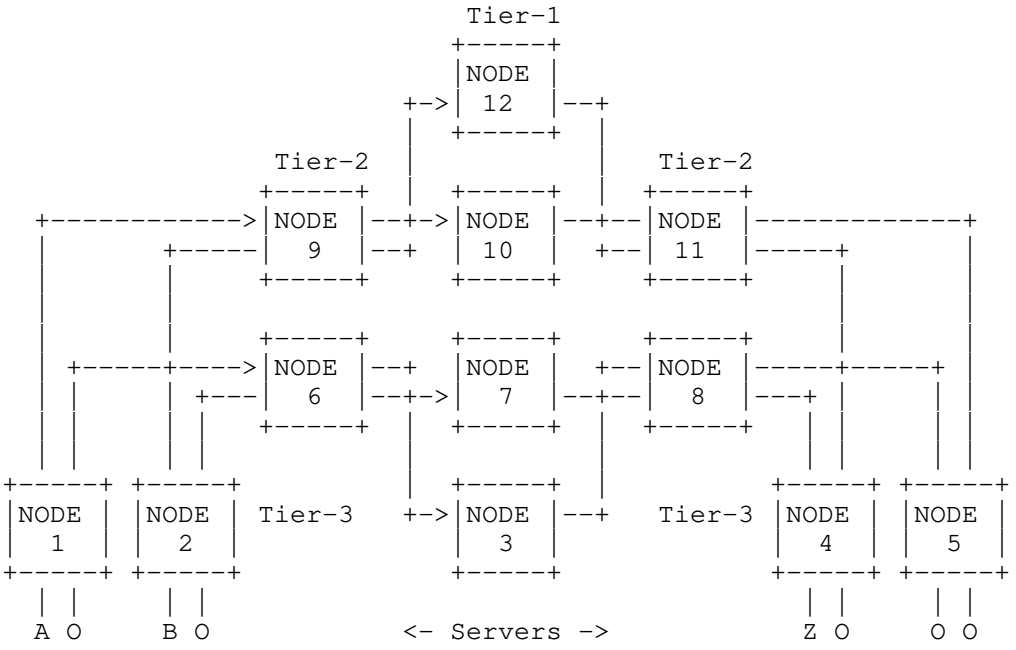


Figure 1: Illustration of the basic CLOS

5. Justification for BGP SPF Extension

In order to simplify layer-3 routing and operations [RFC7938], many data centers use BGP as a routing protocol to create an overlay as well as an underlay network for their CLOS Topologies. However, BGP is a path-vector routing protocol. Since it does not create a fabric topology, it uses hop-by-hop EBGP peering to facilitate hop-by-hop routing to create the underlay network and to resolve any overlay next hops. The hop-by-hop BGP peering paradigm imposes several restrictions within a CLOS. It severely prohibits a deployment of Route Reflectors/Route Controllers as the EBGP sessions are congruent with the data path. The BGP best path algorithm is prefix-based and it prevents announcements of prefixes to other BGP speakers until the best path decision process is performed for the prefix at each intermediate hop. These restrictions significantly delay the overall convergence of the underlay network within a CLOS.

The LSVR SPF modifications allow BGP to overcome these limitations. Furthermore, using the BGP-LS NLRI format [RFC7752] allows the LSVR data to be advertised for nodes, links, and prefixes in the BGP routing domain and used for SPF computations.

6. LSVR Applicability to CLOS Networks

With the BGP SPF extensions [I-D.ietf-lsvr-bgp-spf], the BGP best path computation and route computation are replaced with OSPF-like algorithms [RFC2328] both to determine whether an BGP-LS NLRI has changed and needs to be re-advertised and to compute the routing table. These modifications will significantly improve convergence of the underlay while affording the operational benefits of a single routing protocol [RFC7938].

Data center controllers typically require visibility to the BGP topology to compute traffic-engineered paths. These controllers learn the topology and other relevant information via the BGP-LS address family [RFC7752] which is totally independent of the underlay address families (usually IPv4/IPv6 unicast). Furthermore, in traditional BGP underlays, all the BGP routers will need to advertise their BGP-LS information independently. With the BGP SPF extensions, controllers can learn the topology using the same BGP advertisements used to compute the underlay routes. Furthermore, these data center controllers can avail the convergence advantages of the BGP SPF extensions. The placement of controllers can be outside of the forwarding path or within the forwarding path.

Alternatively, as each and every router in the BGP SPF domain will have a complete view of the topology, the operator can also choose to configure BGP sessions in hop-by-hop peering model described in [RFC7938] along with BFD [RFC5580]. In doing so, while the hop-by-hop peering model lacks inherent benefits of the controller-based model, BGP updates need not be serialized by BGP best path algorithm in either of these models. This helps overall network convergence.

6.1. Usage of BGP-LS SAFI

The BGP SPF extensions [I-D.ietf-lsvr-bgp-spf] define a new BGP-LS SAFI for announcement of BGP SPF link-state. The NLRI format and its associated attributes follow the format of BGP-LS for node, link, and prefix announcements. Whether the peering model within a CLOS follows hop-by-hop peering described in [RFC7938] or any controller-based or route-reflector peering, an operator can exchange BGP SPF SAFI routes over the BGP peering by simply configuring BGP SPF SAFI between the necessary BGP speakers.

The BGP-LS SPF SAFI can also co-exist with BGP IP Unicast SAFI which could exchange overlapping IP routes. The routes received by these SAFIs are evaluated, stored, and announced separately according to the rules of [RFC4760]. The tie-breaking of route installation is a matter of the local policies and preferences of the network operator.

Finally, as the BGP SPF peering is done following the procedures described in [RFC4271], all the existing transport security mechanisms including [RFC5925] are available for the BGP-LS SPF SAFI.

6.1.1. Relationship to Other BGP AFI/SAFI Tuples

Normally, the BGP-LS AFI/SAFI is used solely to compute the underlay and is given preference over other AFI/SAFIs. Other BGP SAFIs, e.g., IPv6/IPv6 Unicast VPN would use the BGP-SPF computed routes for next hop resolution. However, if BGP-LS NLRI is also being advertised for controller consumption, there is no need to replicate the Node, Link, and Prefix NLRI in BGP-NLRI. Rather, additional NLRI attributes can be advertised in the BGP-LS SPF AFI/SAFI as required.

6.2. Peering Models

As previously stated, BGP SPF can be deployed using the existing peering model where there is a single hop BGP session on each and every link in the data center fabric [RFC7938]. This provides for both the advertisement of routes and the determination of link and neighboring switch availability. With BGP SPF, the underlay will converge faster due to changes in the decision process which will allow NLRI changes to be advertised faster after detecting a change.

6.2.1. Sparse Peering Model

Alternately, BFD [RFC5580] can be used to swiftly determine the availability of links and the BGP peering model can be significantly sparser than the data center fabric. BGP SPF sessions then only be established with enough peers to provide a bi-connected graph. If IEBGP is used, then the BGP routers at tier N-1 will act as route-reflectors for the routers at tier N.

The obvious usage of sparse peering is to avoid parallel sessions between the same two BGP speakers in the data center fabric. However, this use case is not very useful since parallel layer-3 links between the same two BGP routers are rare in CLOS or Fat-Tree topologies. Two more interesting scenarios are described below.

In current Data Center topologies, there is often a very dense mesh of links between levels, e.g., leaf and spine, providing 32-way, 64-way, or more Equal-Cost Multi-Path (ECMP) paths. In these topologies, it is desirable not to have a BGP session on every link and techniques such as the one described below Section 6.2.2 can be used establish sessions on some subset of northbound links.

Alternately, controller-based data center topologies are envisioned where BGP speakers within the data center only establish BGP sessions

with two or more controllers. In these topologies, fabric nodes below the first tier (using [RFC7938] hierarchy) will establish BGP multi-hop sessions with the controllers. For the multi-hop sessions, determining the route to the controllers without depending on BGP would need to be through some other means beyond the scope of this document. However, the BGP discovery mechanisms Section 6.3 would be one possibility.

6.2.2. Bi-Connected Graph Heuristic

With this heuristic, discovery of BGP peers is assumed Section 6.3. Additionally, it assumed that the direction of the peering can be ascertained. In the context of a data center fabric, direction is either northbound (toward the spine), southbound (toward the Top-Of-Rack (TOR) switches) or east-west (same level in hierarchy. The determination of the direction is beyond the scope of this document. However, it would be reasonable to assume a technique where the TOR switches can be identified and the number of hops to the TOR is used to determine the direction.

In this heuristic, BGP speakers allow passive session establishment for southbound BGP sessions. For northbound sessions, BGP speakers will attempt to maintain two northbound BGP sessions with different switches (in data center fabrics there is normally a single layer-3 connection anyway). For east-west sessions, passive BGP session establishment is allowed. However, BGP speaker will never actively establish an east-west BGP session unless it can’t establish two northbound BGP sessions.

6.3. BGP Peer Discovery

6.3.1. BGP Peer Discovery Requirements

The most basic requirement is to be able to discover the address of a single-hop peer without pre-configuration. This is being accomplished today with using IPv6 Router Advertisements (RA) [RFC4861] and assuming that a BGP sessions is desired with any discovered peer. Beyond the basic requirement, it is useful to have to following information relating to the BGP session:

- Autonomous System (AS) and BGP Identifier of a potential peer. The latter can be used for debugging and to decrease the likelihood of BGP session establishment collisions.

- Security capabilities supported and for cryptographic authentication, the security capabilities and possibly a key-chain [RFC8177] to be used.

- Session Policy Identifier - A group number or name used to associate common session parameters with the peer. For example, in a data center, BGP sessions with a Top of Rack (ToR) device could have parameters than BGP sessions between leaf and spine.

In a data center fabric, it is often useful to know whether a peer is southbound (towards the servers) or northbound (towards the spine or super-spine) Section 6.2.2. A potential requirement would also be to determine this dynamically. One mechanism, without specifying all the details, might be for the ToRs to be identified when installed and for the others switches in the fabric to determine their level based on the distance from the closest ToR.

If there are multiple links between BGP speakers or the links between BGP speakers are unnumbered, it is also useful to be able to establish multi-hop sessions using the loopback addresses. This will often require the discovery protocol to install route(s) toward the potential peer loopback addresses prior to BGP session establishment.

Finally, a simple BGP discovery protocol could also be used to establish a multi-hop session with one or more controllers by advertising connectivity to one or more controllers. However, once the multi-hop session actually traverses multiple nodes, it is bordering a distance-vector routing protocol and possibly this is not a good requirement for the discovery protocol.

6.3.2. BGP Peer Discovery Alternatives

While BGP peer discovery is not part of [I-D.ietf-lsvr-bgp-spf], there are, at least, three proposals for BGP peer discovery. At least one of these mechanisms will be adopted and will be applicable to deployments other than the data center. It is strongly RECOMMENDED that the accepted mechanism be used in conjunction with BGP SPF in data centers. The BGP discovery mechanism should discovery both peer addresses and endpoints for BFD discovery. Additionally, it would be great if there were a heuristic for determining whether the peer is at a tier above or below the discovering BGP speaker (refer to Section 6.2.2).

The BGP discovery mechanisms under consideration are [I-D.acee-idr-lldp-peer-discovery], [I-D.xu-idr-neighbor-autodiscovery], and [I-D.ymbk-lsvr-lsoe].

6.3.3. Data Center Interconnect (DCI) Applicability

Since BGP SPF is to be used for the routing underlay and DCI gateway boxes typically have direct or very simple connectivity, BGP external sessions would typically not include the BGP SPF SAFI.

6.4. Non-CLOS/FAT Tree Topology Applicability

The BGP SPF extensions [I-D.ietf-lsvr-bgp-spf] can be used in other topologies and avail the inherent convergence improvements. Additionally, sparse peering techniques may be utilized Section 6.2. However, determining whether or to establish a BGP session is more complex and the heuristic described in Section 6.2.2 cannot be used. In such topologies, other techniques such as those described in [I-D.li-lsr-dynamic-flooding] may be employed. One potential deployment would be the underlay for a Service Provider (SP) backbone where usage of a single protocol, i.e., BGP, is desired.

7. IANA Considerations

No IANA updates are requested by this document.

8. Security Considerations

This document introduces no new security considerations above and beyond those already specified in the [RFC4271] and [I-D.ietf-lsvr-bgp-spf].

9. Acknowledgements

The authors would like to thank Alvaro Retana and Yan Filyurin for the review and comments.

10. References

10.1. Normative References

[I-D.ietf-lsvr-bgp-spf]
Patel, K., Lindem, A., Zandi, S., and W. Henderickx, "Shortest Path Routing Extensions for BGP Protocol", draft-ietf-lsvr-bgp-spf-03 (work in progress), September 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>.

[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>.

10.2. Informative References

[CLOS] "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.

[I-D.acee-idr-lldp-peer-discovery] Lindem, A., Patel, K., Zandi, S., Haas, J., and X. Xu, "BGP Logical Link Discovery Protocol (LLDP) Peer Discovery", draft-acee-idr-lldp-peer-discovery-03 (work in progress), June 2018.

[I-D.li-lsr-dynamic-flooding] Li, T., Psenak, P., Ginsberg, L., Przygienda, T., and D. Cooper, "Dynamic Flooding on Dense Graphs", draft-li-lsr-dynamic-flooding-01 (work in progress), October 2018.

[I-D.xu-idr-neighbor-autodiscovery] Xu, X., Talaulikar, K., Bi, K., Tantsura, J., and N. Triantafillis, "BGP Neighbor Discovery", draft-xu-idr-neighbor-autodiscovery-10 (work in progress), October 2018.

[I-D.ymbk-lsvr-lsoe] Bush, R. and K. Patel, "Link State Over Ethernet", draft-ymbk-lsvr-lsoe-01 (work in progress), July 2018.

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

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC4957] Krishnan, S., Ed., Montavont, N., Njedjou, E., Veerepalli, S., and A. Yegin, Ed., "Link-Layer Event Notifications for Detecting Network Attachments", RFC 4957, DOI 10.17487/RFC4957, August 2007, <https://www.rfc-editor.org/info/rfc4957>.

[RFC5580] Tschofenig, H., Ed., Adrangi, F., Jones, M., Lior, A., and B. Aboba, "Carrying Location Objects in RADIUS and Diameter", RFC 5580, DOI 10.17487/RFC5580, August 2009, <https://www.rfc-editor.org/info/rfc5580>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[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>.

[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>.

[RFC8177] Lindem, A., Ed., Qu, Y., Yeung, D., Chen, I., and J. Zhang, "YANG Data Model for Key Chains", RFC 8177, DOI 10.17487/RFC8177, June 2017, <https://www.rfc-editor.org/info/rfc8177>.

Authors' Addresses

Keyur Patel
Arrcus, Inc.
2077 Gateway Pl
San Jose, CA 95110
USA

Email: keyur@arrcus.com

Acee Lindem
Cisco Systems
301 Midenhall Way
Cary, NC 95110
USA

Email: acee@cisco.com

Shawn Zandi
Linkedin
222 2nd Street
San Francisco, CA 94105
USA

Email: szandi@linkedin.com

Gaurav Dawra
Linkedin
222 2nd Street
San Francisco, CA 94105
USA

Email: gdawra@linkedin.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 31, 2019

K. Patel
Arrcus, Inc.
A. Lindem
Cisco Systems
S. Zandi
Linkedin
W. Henderickx
Nokia

September 27, 2018

Shortest Path Routing Extensions for BGP Protocol
draft-ietf-lsvr-bgp-spf-03.txt

Abstract

Many Massively Scaled Data Centers (MSDCs) have converged on simplified layer 3 routing. Furthermore, requirements for operational simplicity have lead many of these MSDCs to converge on BGP as their single routing protocol for both their fabric routing and their Data Center Interconnect (DCI) routing. This document describes a solution which leverages BGP Link-State distribution and the Shortest Path First (SPF) algorithm similar to Internal Gateway Protocols (IGPs) such as OSPF.

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 March 31, 2019.

Copyright Notice

Table of Contents

1. Introduction

Many Massively Scaled Data Centers (MSDCs) have converged on simplified layer 3 routing. Furthermore, requirements for operational simplicity have lead many of these MSDCs to converge on BGP [RFC4271] as their single routing protocol for both their fabric routing and their Data Center Interconnect (DCI) routing. Requirements and procedures for using BGP are described in [RFC7938]. This document describes an alternative solution which leverages BGP-LS [RFC7752] and the Shortest Path First algorithm similar to Internal Gateway Protocols (IGPs) such as OSPF [RFC2328].

[RFC4271] defines the Decision Process that is used to select routes for subsequent advertisement by applying the policies in the local Policy Information Base (PIB) to the routes stored in its Adj-RIBs-In. The output of the Decision Process is the set of routes that are announced by a BGP speaker to its peers. These selected routes are stored by a BGP speaker in the speaker's Adj-RIBs-Out according to policy.

[RFC7752] describes a mechanism by which link-state and TE information can be collected from networks and shared with external components using BGP. This is achieved by defining NLRI advertised within the BGP-LS/BGP-LS-SPF AFI/SAFI. The BGP-LS extensions defined in [RFC7752] makes use of the Decision Process defined in [RFC4271].

This document augments [RFC7752] by replacing its use of the existing Decision Process. Rather than reusing the BGP-LS SAFI, the BGP-LS-SPF SAFI is introduced to insure backward compatibility. The Phase 1 and 2 decision functions of the Decision Process are replaced with the Shortest Path First (SPF) algorithm also known as the Dijkstra algorithm. The Phase 3 decision function is also simplified since it is no longer dependent on the previous phases. This solution avails the benefits of both BGP and SPF-based IGPs. These include TCP based flow-control, no periodic link-state refresh, and completely incremental NLRI advertisement. These advantages can reduce the overhead in MSDCs where there is a high degree of Equal Cost Multi-Path (ECMPs) and the topology is very stable. Additionally, using a

SPF-based computation can support fast convergence and the computation of Loop-Free Alternatives (LFAs) [RFC5286] in the event of link failures. Furthermore, a BGP based solution lends itself to multiple peering models including those incorporating route-reflectors [RFC4456] or controllers.

Support for Multiple Topology Routing (MTR) as described in [RFC4915] is an area for further study dependent on deployment requirements.

1.1. BGP Shortest Path First (SPF) Motivation

Given that [RFC7938] already describes how BGP could be used as the sole routing protocol in an MSDC, one might question the motivation for defining an alternate BGP deployment model when a mature solution exists. For both alternatives, BGP offers the operational benefits of a single routing protocol. However, BGP SPF offers some unique advantages above and beyond standard BGP distance-vector routing.

A primary advantage is that all BGP speakers in the BGP SPF routing domain will have a complete view of the topology. This will allow support for ECMP, IP fast-reroute (e.g., Loop-Free Alternatives), Shared Risk Link Groups (SRLGs), and other routing enhancements without advertisement of addition BGP paths or other extensions. In short, the advantages of an IGP such as OSPF [RFC2328] are availed in BGP.

With the simplified BGP decision process as defined in Section 5.1, NLRI changes can be disseminated throughout the BGP routing domain much more rapidly (equivalent to IGPs with the proper implementation).

Another primary advantage is a potential reduction in NLRI advertisement. With standard BGP distance-vector routing, a single link failure may impact 100s or 1000s prefixes and result in the withdrawal or re-advertisement of the attendant NLRI. With BGP SPF, only the BGP speakers corresponding to the link NLRI need withdraw the corresponding BGP-LS Link NLRI. This advantage will contribute to both faster convergence and better scaling.

With controller and route-reflector peering models, BGP SPF advertisement and distributed computation require a minimal number of sessions and copies of the NLRI since only the latest version of the NLRI from the originator is required. Given that verification of the adjacencies is done outside of BGP (see Section 2), each BGP speaker will only need as many sessions and copies of the NLRI as required for redundancy (e.g., one for the SPF computation and another for backup). Functions such as Optimized Route Reflection (ORR) are supported without extension by virtue of the primary advantages.

Additionally, a controller could inject topology that is learned outside the BGP routing domain.

Given that controllers are already consuming BGP-LS NLRI [RFC7752], reusing for the BGP-LS SPF leverages the existing controller implementations.

Another potential advantage of BGP SPF is that both IPv6 and IPv4 can be supported in the same address family using the same topology. Although not described in this version of the document, multi-topology extensions can be used to support separate IPv4, IPv6, unicast, and multicast topologies while sharing the same NLRI.

Finally, the BGP SPF topology can be used as an underlay for other BGP address families (using the existing model) and realize all the above advantages. A simplified peering model using IPv6 link-local addresses as next-hops can be deployed similar to [RFC5549].

1.2. 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.

2. BGP Peering Models

Depending on the requirements, scaling, and capabilities of the BGP speakers, various peering models are supported. The only requirement is that all BGP speakers in the BGP SPF routing domain receive link-state NLRI on a timely basis, run an SPF calculation, and update their data plane appropriately. The content of the Link NLRI is described in Section 4.2.

2.1. BGP Single-Hop Peering on Network Node Connections

The simplest peering model is the one described in section 5.2.1 of [RFC7938]. In this model, EBGP single-hop sessions are established over direct point-to-point links interconnecting the SPF domain nodes. For the purposes of BGP SPF, Link NLRI is only advertised if a single-hop BGP session has been established and the Link-State/SPF address family capability has been exchanged [RFC4790] on the corresponding session. If the session goes down, the corresponding Link NLRI will be withdrawn. Topologically, this would be equivalent to the peering model in [RFC7938] where there is a BGP session on every link in the data center switch fabric.

2.2. BGP Peering Between Directly Connected Network Nodes

In this model, BGP speakers peer with all directly connected network nodes but the sessions may be multi-hop and the direct connection discovery and liveliness detection for those connections are independent of the BGP protocol. How this is accomplished is outside the scope of this document. Consequently, there will be a single session even if there are multiple direct connections between BGP speakers. For the purposes of BGP SPF, Link NLRI is advertised as long as a BGP session has been established, the Link-State/SPF address family capability has been exchanged [RFC4790] and the corresponding link is considered is up and considered operational. This is much like the previous peering model only peering is on a single loopback address and the switch fabric links can be unnumbered. However, there will be the same unnumber of sessions as with the previous peering model unless there are parrallel links between switches in the fabric.

2.3. BGP Peering in Route-Reflector or Controller Topology

In this model, BGP speakers peer solely with one or more Route Reflectors [RFC4456] or controllers. As in the previous model, direct connection discovery and liveliness detection for those connections are done outside the BGP protocol. More specifically, the Liveliness detection is done using BFD protocol described in [RFC5880]. For the purposes of BGP SPF, Link NLRI is advertised as long as the corresponding link is up and considered operational.

This peering model, known as sparse peering, allows for many fewer BGP sessions and, consequently, instances of the same NLRI received from multiple peers. It is discussed in greater detail in [I-D.ietf-lsvr-applicability].

3. BGP-LS Shortest Path Routing (SPF) SAFI

In order to replace the Phase 1 and 2 decision functions of the existing Decision Process with an SPF-based Decision Process and streamline the Phase 3 decision functions in a backward compatible manner, this draft introduces the BGP-LS-SFP SAFI for BGP-LS SPF operation. The BGP-LS-SPF (AF 16388 / SAFI TBD1) [RFC4790] is allocated by IANA as specified in the Section 6. A BGP speaker using the BGP-LS SPF extensions described herein MUST exchange the AFI/SAFI using Multiprotocol Extensions Capability Code [RFC4760] with other BGP speakers in the SPF routing domain.

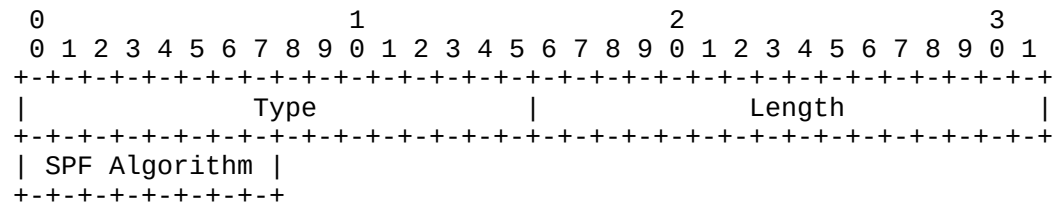
4. Extensions to BGP-LS

[RFC7752] describes a mechanism by which link-state and TE information can be collected from networks and shared with external components using BGP protocol. It describes both the definition of BGP-LS NLRI that describes links, nodes, and prefixes comprising IGP link-state information and the definition of a BGP path attribute (BGP-LS attribute) that carries link, node, and prefix properties and attributes, such as the link and prefix metric or auxiliary Router-IDs of nodes, etc.

The BGP protocol will be used in the Protocol-ID field specified in table 1 of [I-D.ietf-idr-bgpls-segment-routing-epe]. The local and remote node descriptors for all NLRI will be the BGP Router-ID (TLV 516) and either the AS Number (TLV 512) [RFC7752] or the BGP Confederation Member (TLV 517) [RFC8402]. However, if the BGP Router-ID is known to be unique within the BGP Routing domain, it can be used as the sole descriptor.

4.1. Node NLRI Usage and Modifications

The SPF capability is a new Node Attribute TLV that will be added to those defined in table 7 of [RFC7752]. The new attribute TLV will only be applicable when BGP is specified in the Node NLRI Protocol ID field. The TBD TLV type will be defined by IANA. The new Node Attribute TLV will contain a single-octet SPF algorithm as defined in [RFC8402].



The SPF Algorithm may take the following values:

0 - Normal Shortest Path First (SPF) algorithm based on link metric. This is the standard shortest path algorithm as computed by the IGP protocol. Consistent with the deployed practice for link-state protocols, Algorithm 0 permits any node to overwrite the SPF path with a different path based on its local policy.
1 - Strict Shortest Path First (SPF) algorithm based on link metric. The algorithm is identical to Algorithm 0 but Algorithm 1 requires that all nodes along the path will honor the SPF routing decision. Local policy at the node claiming support for Algorithm 1 MUST NOT alter the SPF paths computed by Algorithm 1.

Note that usage of Strict Shortest Path First (SPF) algorithm is defined in the IGP algorithm registry but usage is restricted to [I-D.ietf-idr-bgpls-segment-routing-epe]. Hence, its usage for BGP-LS SPF is out of scope.

When computing the SPF for a given BGP routing domain, only BGP nodes advertising the SPF capability attribute will be included the Shortest Path Tree (SPT).

4.2. Link NLRI Usage

The criteria for advertisement of Link NLRI are discussed in Section 2.

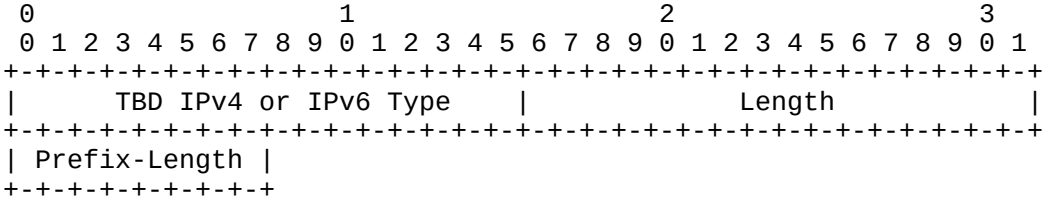
Link NLRI is advertised with local and remote node descriptors as described above and unique link identifiers dependent on the addressing. For IPv4 links, the links local IPv4 (TLV 259) and remote IPv4 (TLV 260) addresses will be used. For IPv6 links, the local IPv6 (TLV 261) and remote IPv6 (TLV 262) addresses will be used. For unnumbered links, the link local/remote identifiers (TLV 258) will be used. For links supporting having both IPv4 and IPv6 addresses, both sets of descriptors may be included in the same Link NLRI. The link identifiers are described in table 5 of [RFC7752].

The link IGP metric attribute TLV (TLV 1095) as well as any others required for non-SPF purposes SHOULD be advertised. Algorithms such

as setting the metric inversely to the link speed as done in the OSPF MIB [RFC4750] MAY be supported. However, this is beyond the scope of this document.

4.2.1. BGP-LS Link NLRI Attribute Prefix-Length TLVs

Two BGP-LS Attribute TLVs to BGP-LS Link NLRI are defined to advertise the prefix length associated with the IPv4 and IPv6 link prefixes. The prefix length is used for the optional installation of prefixes corresponding to Link NLRI as defined in Section 5.3.



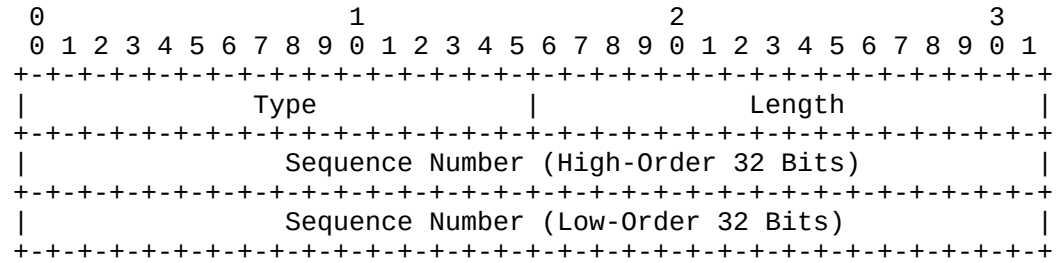
Prefix-length - A one-octet length restricted to 1-32 for IPv4 Link NLIR endpoint prefixes and 1-128 for IPv6 Link NLRI endpoint prefixes.

4.3. Prefix NLRI Usage

Prefix NLRI is advertised with a local node descriptor as described above and the prefix and length used as the descriptors (TLV 265) as described in [RFC7752]. The prefix metric attribute TLV (TLV 1155) as well as any others required for non-SPF purposes SHOULD be advertised. For loopback prefixes, the metric should be 0. For non-loopback prefixes, the setting of the metric is a local matter and beyond the scope of this document.

4.4. BGP-LS Attribute Sequence-Number TLV

A new BGP-LS Attribute TLV to BGP-LS NLRI types is defined to assure the most recent version of a given NLRI is used in the SPF computation. The TBD TLV type will be defined by IANA. The new BGP-LS Attribute TLV will contain an 8-octet sequence number. The usage of the Sequence Number TLV is described in Section 5.1.



Sequence Number

The 64-bit strictly increasing sequence number is incremented for every version of BGP-LS NLRI originated. BGP speakers implementing this specification MUST use available mechanisms to preserve the sequence number's strictly increasing property for the deployed life of the BGP speaker (including cold restarts). One mechanism for accomplishing this would be to use the high-order 32 bits of the sequence number as a wrap/boot count that is incremented anytime the BGP router loses its sequence number state or the low-order 32 bits wrap.

When incrementing the sequence number for each self-originated NLRI, the sequence number should be treated as an unsigned 64-bit value. If the lower-order 32-bit value wraps, the higher-order 32-bit value should be incremented and saved in non-volatile storage. If by some chance the BGP Speaker is deployed long enough that there is a possibility that the 64-bit sequence number may wrap or a BGP Speaker completely loses its sequence number state (e.g., the BGP speaker hardware is replaced or experiences a cold-start), the phase 1 decision function (see Section 5.1) rules will insure convergence, albeit, not immediately.

5. Decision Process with SPF Algorithm

The Decision Process described in [RFC4271] takes place in three distinct phases. The Phase 1 decision function of the Decision Process is responsible for calculating the degree of preference for each route received from a BGP speaker's peer. The Phase 2 decision function is invoked on completion of the Phase 1 decision function and is responsible for choosing the best route out of all those available for each distinct destination, and for installing each chosen route into the Loc-RIB. The combination of the Phase 1 and 2 decision functions is characterized as a Path Vector algorithm.

The SPF based Decision process replaces the BGP best-path Decision process described in [RFC4271]. This process starts with selecting only those Node NLRI whose SPF capability TLV matches with the local

BGP speaker's SPF capability TLV value. Since Link-State NLRI always contains the local descriptor [RFC7752], it will only be originated by a single BGP speaker in the BGP routing domain. These selected Node NLRI and their Link/Prefix NLRI are used to build a directed graph during the SPF computation. The best paths for BGP prefixes are installed as a result of the SPF process.

When BGP-LS-SPF NLRI is received, all that is required is to determine whether it is the best-path by examining the Node-ID and sequence number as described in Section 5.1. If the received best-path NLRI had changed, it will be advertised to other BGP-LS-SPF peers. If the attributes have changed (other than the sequence number), a BGP SPF calculation will be scheduled. However, a changed NLRI MAY be advertised to other peers almost immediately and propagation of changes can approach IGP convergence times. To accomplish this, the MinRouteAdvertisementIntervalTimer and MinASOriginationIntervalTimer [RFC4271] are not applicable to the BGP-LS-SPF SAFI. Rather, SPF calculations SHOULD be triggered and dampened consistent with the SPF backoff algorithm specified in [RFC8405].

The Phase 3 decision function of the Decision Process [RFC4271] is also simplified since under normal SPF operation, a BGP speaker would advertise the NLRI selected for the SPF to all BGP peers with the BGP-LS/BGP-LS-SPF AFI/SAFI. Application of policy would not be prevented however its usage to best-path process would be limited as the SPF relies solely on link metrics.

5.1. Phase-1 BGP NLRI Selection

The rules for NLRI selection are greatly simplified from [RFC4271].

1. If the NLRI is received from the BGP speaker originating the NLRI (as determined by the comparing BGP Router ID in the NLRI Node identifiers with the BGP speaker Router ID), then it is preferred over the same NLRI from non-originators. This rule will assure that stale NLRI is updated even if a BGP-LS router loses its sequence number state due to a cold-start.

2. If the Sequence-Number TLV is present in the BGP-LS Attribute, then the NLRI with the most recent, i.e., highest sequence number is selected. BGP-LS NLRI with a Sequence-Number TLV will be considered more recent than NLRI without a BGP-LS Attribute or a BGP-LS Attribute that doesn't include the Sequence-Number TLV.

3. The final tie-breaker is the NLRI from the BGP Speaker with the numerically largest BGP Router ID.

When a BGP speaker completely loses its sequence number state, i.e., due to a cold start, or in the unlikely possibility that that sequence number wraps, the BGP routing domain will still converge. This is due to the fact that BGP speakers adjacent to the router will always accept self-originated NLRI from the associated speaker as more recent (rule # 1). When BGP speaker reestablishes a connection with its peers, any existing session will be taken down and stale NLRI will be replaced by the new NLRI and stale NLRI will be discarded independent of whether or not BGP graceful restart is deployed, [RFC4724]. The adjacent BGP speaker will update their NLRI advertisements in turn until the BGP routing domain has converged.

The modified SPF Decision Process performs an SPF calculation rooted at the BGP speaker using the metrics from Link and Prefix NLRI Attribute TLVs [RFC7752]. As a result, any attributes that would influence the Decision process defined in [RFC4271] like ORIGIN, MULTI_EXIT_DISC, and LOCAL_PREF attributes are ignored by the SPF algorithm. Furthermore, the NEXT_HOP attribute value is preserved but otherwise ignored during the SPF or best-path.

5.2. Dual Stack Support

The SPF-based decision process operates on Node, Link, and Prefix NLRIs that support both IPv4 and IPv6 addresses. Whether to run a single SPF instance or multiple SPF instances for separate AFs is a matter of a local implementation. Normally, IPv4 next-hops are calculated for IPv4 prefixes and IPv6 next-hops are calculated for IPv6 prefixes. However, an interesting use-case is deployment of [RFC5549] where IPv6 next-hops are calculated for both IPv4 and IPv6 prefixes. As stated in Section 1, support for Multiple Topology Routing (MTR) is an area for future study.

5.3. SPF Calculation based on BGP-LS NLRI

This section details the BGP-LS SPF local routing information base (RIB) calculation. The router will use BGP-LS Node, Link, and Prefix NLRI to populate the local RIB using the following algorithm. This calculation yields the set of intra-area routes associated with the BGP-LS domain. A router calculates the shortest-path tree using itself as the root. Variations and optimizations of the algorithm are valid as long as it yields the same set of routes. The algorithm below supports Equal Cost Multi-Path (ECMP) routes. Weighted Unequal Cost Multi-Path are out of scope. The organization of this section owes heavily to section 16 of [RFC2328].

The following abstract data structures are defined in order to specify the algorithm.

- Local Route Information Base (RIB) - This is abstract contains reachability information (i.e., next hops) for all prefixes (both IPv4 and IPv6) as well as the Node NLRI reachability. Implementations may choose to implement this as separate RIBs for each address family and/or Node NLRI.

- Link State NLRI Database (LSNDB) - Database of BGP-LS NLRI that facilitates access to all Node, Link, and Prefix NLRI as well as all the Link and Prefix NLRI corresponding to a given Node NLRI. Other optimization, such as, resolving bi-directional connectivity associations between Link NLRI are possible but of scope of this document.

- Candidate List - This is a list of candidate Node NLRI with the lowest cost Node NLRI at the front of the list. It is typically implemented as a heap but other concrete data structures have also been used.

The algorithm is comprised of the steps below:

1. The current local RIB is invalidated. The local RIB is built again from scratch. The existing routing entries are preserved for comparision to determine changes that need to be installed in the global RIB.

2. The computing router’s Node NLRI is installed in the local RIB with a cost of 0 and as as the sole entry in the candidate list.

3. The Node NLRI with the lowest cost is removed from the candidate list for processing. The Node corresponding to this NLRI will be referred to as the Current Node. If the candidate list is empty, the SPF calculation has completed and the algorithm proceeds to step 6.

4. All the Prefix NLRI with the same Node Identifiers as the Current Node will be considered for installation. The cost for each prefix is the metric advertised in the Prefix NLRI added to the cost to reach the Current Node.

    * If the prefix is not in the local RIB, the prefix is installed and will inherit the Current Node’s next hops.

    * If the prefix is in the local RIB and the cost is greater than the Current route’s metric, the Prefix NLRI does not contribute to the route and is ignored.

    * If the prefix is in the local RIB and the cost is less than the current route’s metric, the Prefix is installed with the

Current Node's next-hops replacing the local RIB route's next-hops and the metric being updated.

* If the prefix is in the local RIB and the cost is same as the current route's metric, the Prefix is installed with the Current Node's next-hops being merged with local RIB route's next-hops.

5. All the Link NLRI with the same Node Identifiers as the Current Node will be considered for installation. Each link will be examined and will be referred to in the following text as the Current Link. The cost of the Current Link is the advertised metric in the Link NLRI added to the cost to reach the Current Node.

   * Optionally, the prefix(es) associated with the Current Link are installed into the local RIB using the same rules as were used for Prefix NLRI in the previous steps.

   * The Current Link's endpoint Node NLRI is accessed (i.e., the Node NLRI with the same Node identifiers as the Link endpoint). If it exists, it will be referred to as the Endpoint Node NLRI and the algorithm will proceed as follows:

     + All the Link NLRI corresponding the Endpoint Node NLRI will be searched for a back-link NLRI pointing to the current node. Both the Node identifiers and the Link endpoint identifiers in the Endpoint Node's Link NLRI must match for a match. If there is no corresponding Link NLRI corresponding to the Endpoint Node NLRI, the Endpoint Node NLIR fails the bi-directional connectivity test and is not processed further.

     + If the Endpoint Node NLRI is not on the candidate list, it is inserted based on the link cost and BGP Identifier (the latter being used as a tie-breaker).

     + If the Endpoint Node NLRI is already on the candidate list with a lower cost, it need not be inserted again.

     + If the Endpoint Node NLRI is already on the candidate list with a higher cost, it must be removed and reinserted with a lower cost.

   * Return to step 3 to process the next lowest cost Node NLRI on the candidate list.

6. The local RIB is examined and changes (adds, deletes, modifications) are installed into the global RIB.

5.4. NEXT_HOP Manipulation

A BGP speaker that supports SPF extensions MAY interact with peers that don't support SPF extensions. If the BGP-LS address family is advertised to a peer not supporting the SPF extensions described herein, then the BGP speaker MUST conform to the NEXT_HOP rules specified in [RFC4271] when announcing the Link-State address family routes to those peers.

All BGP peers that support SPF extensions would locally compute the Loc-RIB next-hops as a result of the SPF process. Consequently, the NEXT_HOP attribute is always ignored on receipt. However, BGP speakers SHOULD set the NEXT_HOP address according to the NEXT_HOP attribute rules specified in [RFC4271].

5.5. IPv4/IPv6 Unicast Address Family Interaction

While the BGP-LS SPF address family and the IPv4/IPv6 unicast address families install routes into the same device routing tables, they will operate independently much the same as OSPF and IS-IS would operate today (i.e., "Ships-in-the-Night" mode). There will be no implicit route redistribution between the BGP address families. However, implementation specific redistribution mechanisms SHOULD be made available with the restriction that redistribution of BGP-LS SPF routes into the IPv4 address family applies only to IPv4 routes and redistribution of BGP-LS SPF route into the IPv6 address family applies only to IPv6 routes.

Given the fact that SPF algorithms are based on the assumption that all routers in the routing domain calculate the precisely the same SPF tree and install the same set of routes, it is RECOMMENDED that BGP-LS SPF IPv4/IPv6 routes be given priority by default when installed into their respective RIBs. In common implementations the prioritization is governed by route preference or administrative distance with lower being more preferred.

5.6. NLRI Advertisement and Convergence

A local failure will prevent a link from being used in the SPF calculation due to the IGP bi-directional connectivity requirement. Consequently, local link failures should always be given priority over updates (e.g., withdrawing all routes learned on a session) in order to ensure the highest priority propagation and optimal convergence.

Delaying the withdrawal of non-local routes is an area for further study as more IGP-like mechanisms would be required to prevent usage of stale NLRI.

5.7. Error Handling

When a BGP speaker receives a BGP Update containing a malformed SPF Capability TLV in the Node NLRI BGP-LS Attribute [RFC7752], it MUST ignore the received TLV and the Node NLRI and not pass it to other BGP peers as specified in [RFC7606]. When discarding a Node NLRI with malformed TLV, a BGP speaker SHOULD log an error for further analysis.

6. IANA Considerations

This document defines an AFI/SAFI for BGP-LS SPF operation and requests IANA to assign the BGP-LS/BGP-LS-SPF (AFI 16388 / SAFI TBD1) as described in [RFC4750].

This document also defines four attribute TLVs for BGP LS NLRI. We request IANA to assign TLVs for the SPF capability, Sequence Number, IPv4 Link Prefix-Length, and IPv6 Link Prefix-Length from the "BGP-LS Node Descriptor, Link Descriptor, Prefix Descriptor, and Attribute TLVs" Registry.

7. Security Considerations

This extension to BGP does not change the underlying security issues inherent in the existing [RFC4271], [RFC4724], and [RFC7752].

8. Management Considerations

This section includes unique management considerations for the BGP-LS SPF address family.

8.1. Configuration

In addition to configuration of the BGP-LS SPF address family, implementations SHOULD support the configuratio of the INITIAL_SPF_DELAY, SHORT_SPF_DELAY, LONG_SPF_DELAY, TIME_TO_LEARN, and HOLDDOWN_INTERVAL as documented in [RFC8405].

8.2. Operational Data

In order to troubleshoot SPF issues, implementations SHOULD support an SPF log including entries for previous SPF computations, Each SPF log entry would include the BGP-LS NLRI SPF triggering the SPF, SPF scheduled time, SPF start time, SPF end time, and SPF type if

different types of SPF are supported. Since the size of the log will be finite, implementations SHOULD also maintain counters for the total number of SPF computations of each type and the total number of SPF triggering events. Additionally, to troubleshoot SPF scheduling and backoff [RFC8405], the current SPF backoff state, remaining time-to-learn, remaining holddown, last trigger event time, last SPF time, and next SPF time should be available.

9. Acknowledgements

The authors would like to thank Sue Hares, Jorge Rabadan, Boris Hassanov, Dan Frost, and Fred Baker for their review and comments.

10. Contributors

In addition to the authors listed on the front page, the following co-authors have contributed to the document.

Derek Yeung
Arrcus, Inc.
derek@arrcus.com

Gunter Van De Velde
Nokia
gunter.van_de_velde@nokia.com

Abhay Roy
Cisco Systems
akr@cisco.com

Venu Venugopal
Cisco Systems
venuv@cisco.com

11. References

11.1. Normative References

[I-D.ietf-idr-bgpls-segment-routing-epe] Previdi, S., Filsfils, C., Patel, K., Ray, S., and J. Dong, "BGP-LS extensions for Segment Routing BGP Egress Peer Engineering", draft-ietf-idr-bgpls-segment-routing-epe-15 (work in progress), March 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>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[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>.

[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>.

[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>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8405] Decraene, B., Litkowski, S., Gredler, H., Lindem, A., Francois, P., and C. Bowers, "Shortest Path First (SPF) Back-Off Delay Algorithm for Link-State IGPs", RFC 8405, DOI 10.17487/RFC8405, June 2018, <https://www.rfc-editor.org/info/rfc8405>.

11.2. Information References

[I-D.ietf-lsvr-applicability] Patel, K., Lindem, A., Zandi, S., and G. Dawra, "Usage and Applicability of Link State Vector Routing in Data Centers", draft-ietf-lsvr-applicability-00 (work in progress), July 2018.

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

[RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <https://www.rfc-editor.org/info/rfc4456>.

[RFC4724] Sangli, S., Chen, E., Fernando, R., Scudder, J., and Y. Rekhter, "Graceful Restart Mechanism for BGP", RFC 4724, DOI 10.17487/RFC4724, January 2007, <https://www.rfc-editor.org/info/rfc4724>.

[RFC4750] Joyal, D., Ed., Galecki, P., Ed., Giacalone, S., Ed., Coltun, R., and F. Baker, "OSPF Version 2 Management Information Base", RFC 4750, DOI 10.17487/RFC4750, December 2006, <https://www.rfc-editor.org/info/rfc4750>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC4790] Newman, C., Duerst, M., and A. Gulbrandsen, "Internet Application Protocol Collation Registry", RFC 4790, DOI 10.17487/RFC4790, March 2007, <https://www.rfc-editor.org/info/rfc4790>.

[RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, DOI 10.17487/RFC4915, June 2007, <https://www.rfc-editor.org/info/rfc4915>.

[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>.

[RFC5549] Le Faucheur, F. and E. Rosen, "Advertising IPv4 Network Layer Reachability Information with an IPv6 Next Hop", RFC 5549, DOI 10.17487/RFC5549, May 2009, <https://www.rfc-editor.org/info/rfc5549>.

[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>.

Authors' Addresses

Keyur Patel
Arrcus, Inc.

Email: keyur@arrcus.com

Acee Lindem
Cisco Systems
301 Midenhall Way
Cary, NC 27513
USA

Email: acee@cisco.com

Shawn Zandi
Linkedin
222 2nd Street
San Francisco, CA 94105
USA

Email: szandi@linkedin.com

Wim Henderickx
Nokia
Antwerp
Belgium

Email: wim.henderickx@nokia.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 31, 2019

R. Bush
Arrcus & IIJ
K. Patel
Arrcus

September 27, 2018

Link State Over Ethernet
draft-ymbk-lsvr-lsoe-02

Abstract

Used in Massive Data Centers (MDCs), BGP-SPF and similar protocols need link neighbor discovery, link encapsulation data, and Layer 2 liveness. The Link State Over Ethernet protocol provides link discovery, exchanges supported encapsulations (IPv4, IPv6, ...), discovers encapsulation addresses (Layer 3 / MPLS identifiers) over raw Ethernet, and provides layer 2 liveness checking. The interface data are pushed directly to a BGP-LS API, obviating the need for centralized controller architectures. This protocol is intended to be more widely applicable to other upper layer routing protocols which need link discovery and characterisation.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [RFC2119] only when they appear in all upper case. They may also appear in lower or mixed case as English words, without normative meaning. See [RFC8174].

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 31, 2019.

Table of Contents

1. Introduction

The Massive Data Center (MDC) environment presents unusual problems of scale, e.g. O(10,000) devices, while its homogeneity presents opportunities for simple approaches. Approaches such as Jupiter Rising [JUPITER] use a central controller to deal with scaling, while BGP-SPF [I-D.ietf-lsvr-bgp-spf] provides massive scale out without centralization using a tried and tested scalable distributed control plane, offering a scalable routing solution in Clos and similar environments. But BGP-SPF and similar higher level device-spanning protocols need link state and addressing data from the network to build the routing topology. LLDP has scaling issues, e.g. in extending a PDU beyond 1,500 bytes.

Link State Over Ethernet (LSOE) provides brutally simple mechanisms for devices to

- Discover each other's Layer 2 (MAC) Addresses,
- Run Layer 2 keep-alive messages for liveness continuity,
- Discover each other's unique IDs (ASN, RouterID, ...),
- Discover mutually supported encapsulations, e.g. IP/MPLS,
- Discover Layer 3 and/or MPLS addressing of interfaces of the link encapsulations,
- Enable layer 3 link liveness such as BFD, and finally
- Present these data, using a very restricted profile of a BGP-LS [RFC7752] API, to BGP-SPF which computes the topology and builds routing and forwarding tables.

This protocol may be more widely applicable to a range of routing and similar protocols which need link discovery and characterisation.

2. Terminology

Even though it concentrates on the Ethernet layer, this document relies heavily on routing terminology. The following are some possibly confusing terms:

Association: An established, vis OPEN messages, session between two LSOE capable devices,

ASN: Autonomous System Number [RFC4271], a BGP identifier for an originator of Layer 3 routes, particularly BGP announcements.

BGP-LS: A mechanism by which link-state and TE information can be collected from networks and shared with external components using the BGP routing protocol. See [RFC7752].

BGP-SPF A hybrid protocol using BGP transport but a Dijkstra SPF decision process. See [I-D.ietf-lsvr-bgp-spf].

Clos: A hierarchic subset of a crossbar switch topology commonly used in data centers.

Encapsulation: Address Family Indicator and Subsequent Address Family Indicator (AFI/SAFI). I.e. classes of addresses such as IPv4, IPv6, MPLS, ...

Frame: An Ethernet Layer 2 packet.

MAC Address: Media Access Control Address, essentially an Ethernet address, six octets.

MDC: Massive Data Center, commonly thousands of TORs.

Message: A complete application layer message. These may be longer than will fit in a single Ethernet frame.

PDU: Protocol Data Unit, essentially an application layer packet. A Message may need to be broken into multiple PDUs to make it through MTU or other restrictions.

RouterID: An 32-bit identifier unique in the current routing domain, see [RFC4271] updated by [RFC6286].

SPF: Shortest Path First, an algorithm for finding the shortest paths between nodes in a graph; AKA Dijkstra’s algorithm.

TOR: Top Of Rack switch, aggregates the servers in a rack and connects to aggregation layers of the Clos tree, AKA the Clos spine.

ZTP: Zero Touch Provisioning gives devices initial addresses, credentials, etc. on boot/restart.

3. Background

LSOE assumes a datacenter scale and topology, but can accommodate richer topologies which contain potential cycles.

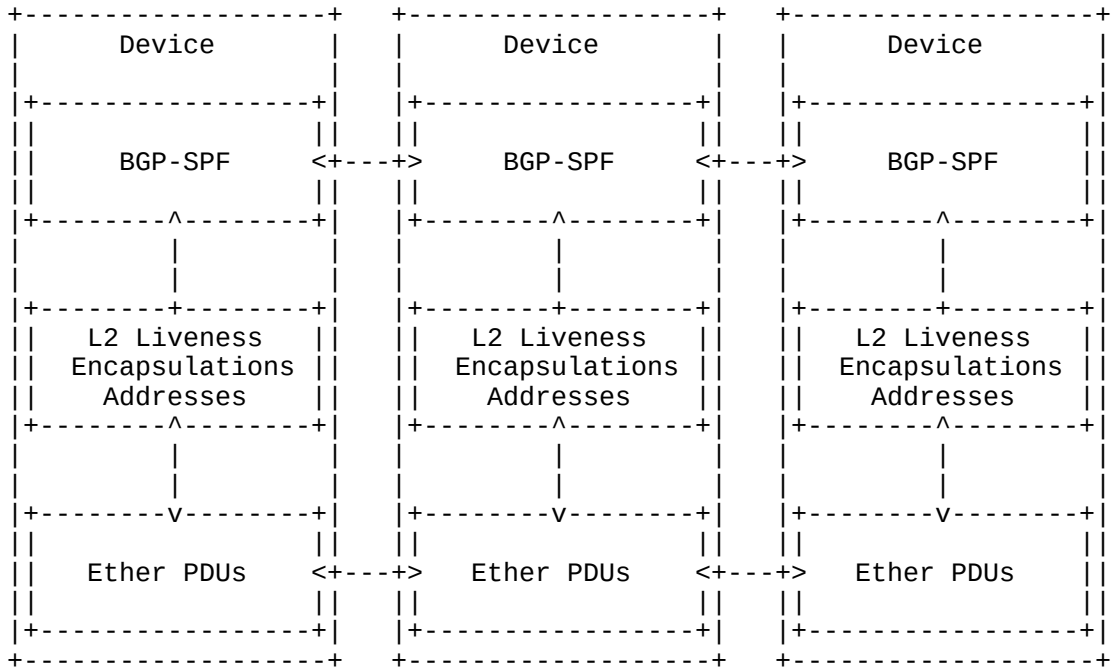
While LSOE is designed for the MDC, there are no inherent reasons it could not run on a WAN; though, as it is simply a discovery protocol, it is not clear that this would be useful. The authentication and

authorisation needed to run safely on the WAN are not provided in detail in this version of the protocol, although future versions/ extensions could expend on them.

LSOE assumes a new IEEE assigned EtherType (TBD).

4. Top Level Overview

- Devices discover each other on Ethernet links
- MAC addresses and Link State are exchanged over Ethernet
- Layer 2 Liveness Checks are begun
- Encapsulation data are exchanged and IP-Level Liveness Checks done
- A BGP-like protocol is assumed to use these data to discover and build a topology database



There are two protocols, the Ethernet discovery and the interface to the upper level BGP-like protocol:

- Layer 2 Ethernet protocols are used to exchange Layer 2 data, i.e. MAC addresses, and layer 2.5 and 3 identifiers (not payloads), i.e. ASNs, Encapsulations, and interface addresses.

- A Link Layer to BGP API presents these data up the stack to a BGP protocol or an other device-spanning upper layer protocol, presenting them using the BGP-LS BGP-like data format.

The upper layer BGP family routing protocols cross all the devices, though they are not part of these LSOE protocols.

To simplify this document, Layer 2 Ethernet framing is not shown.

5. Ethernet to Ethernet Protocols

Two devices discover each other and their respective MAC addresses by sending multicast HELLO messages (Section 5.3). To allow discovery of new devices coming up on a multi-link topology, devices send periodic HELLOs forever, see Section 8.1.

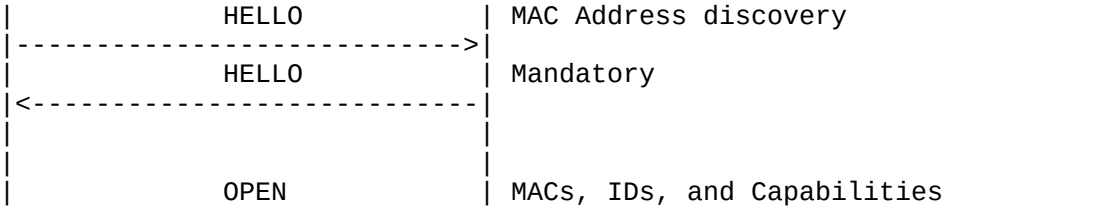
Once a new device is recognized, both devices attempt to negotiate and establish peering by sending unicast OPEN messages (Section 5.4). In an established peering, Encapsulations (Section 5.5) may be announced and modified. When two devices on a link have compatible Encapsulations and addresses, i.e. the same AFI/SAFI and the same subnet, the link is announced via the BGP-LS API.

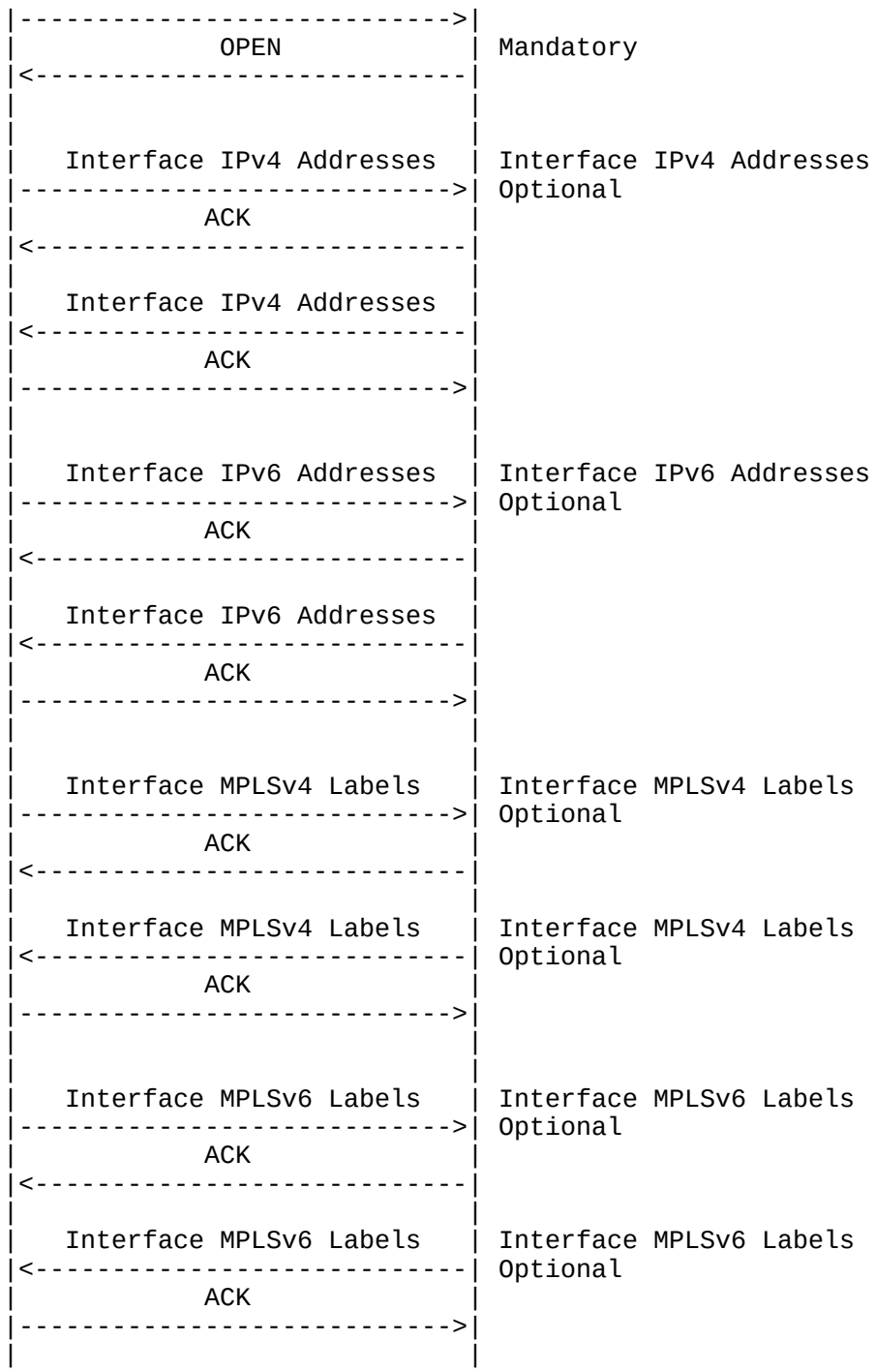
5.1. Inter-Link Ether Protocol Overview

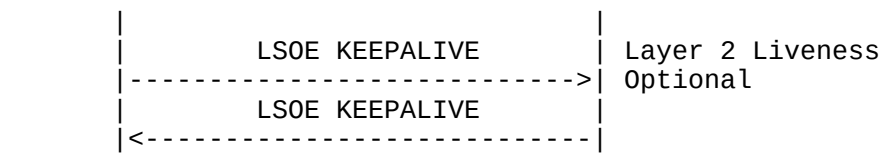
The HELLO, Section 5.3, is a priming message. It is an Ethernet multicast of a minimal message with the simple goal of discovering the Ethernet MAC address(es) of devices reachable via an interface.

The HELLO and OPEN, Section 5.4, messages, which are used to discover and exchange MAC address and IDs, are mandatory; other messages are optional; though at least one encapsulation MUST be agreed at some point.

The following is a ladder-style sketch of the Ethernet protocol exchanges:




OPEN
Mandatory
Interface IPv4 Addresses
Interface IPv4 Addresses
Optional
ACK
Interface IPv4 Addresses
ACK
Interface IPv6 Addresses
Interface IPv6 Addresses
Optional
ACK
Interface IPv6 Addresses
ACK
Interface MPLSv4 Labels
Interface MPLSv4 Labels
Optional
ACK
Interface MPLSv4 Labels
Interface MPLSv4 Labels
Optional
ACK
Interface MPLSv6 Labels
Interface MPLSv6 Labels
Optional
ACK
Interface MPLSv6 Labels
Interface MPLSv6 Labels
Optional
ACK



5.2. Messages, PDUs, and Frames

An LSOE Message occupies one or more Ethernet Frames. We refer to the payload in an Ethernet frame as a PDU (Protocol Data Unit).

If a message will not fit in a single frame/PDU, likely due to MTU issues, then the message is broken into multiple PDUs each in its own frame, each with a full header. The OPEN message and the Encapsulation messages provide for multi-PDU packaging.

The Flags field indicates whether the message required multiple PDUs and if the current PDU is the last of a multi-PDU sequence; see Section 5.2.1.

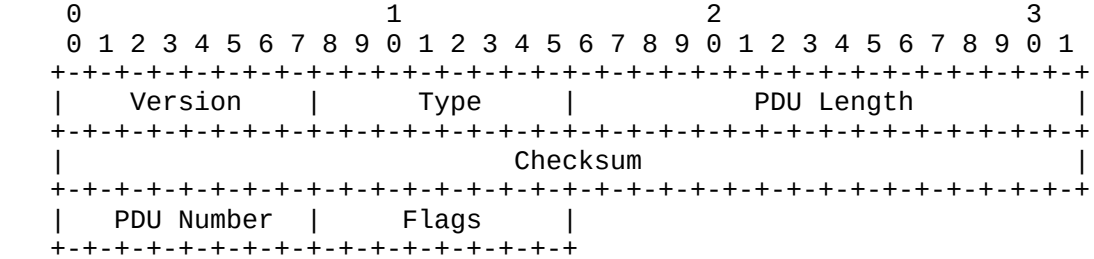
The PDU Number field of a multi-PDU message must monotonically increase, modulo 256, see [RFC1982]. This and the Flags field may be used to re-assemble long messages.

5.2.1. Frame TLV

The basic LSOE message is a typical TLV (Type Length Value) PDU with a Version number in front.

Aside from the HELLO, Section 5.3, and KEEPALIVE, Section 5.7, an LSOE PDU has a PDU Sequence Number and a Flags field to allow assembly of out order PDUs/frames of messages too long to fit in one Ethernet frame.

All LSOE frames/PDUs, other than the HELLO and KEEPALIVE, have the following header:



The fields of the basic LSOE header are as follows:

Version: Version number of the protocol, currently 0.

Type: An integer differentiating message payload types

0 - HELLO
1 - OPEN
2 - KEEPALIVE
3 - Encapsulation ACK
4 - IPv4 Announcement
5 - IPv6 Announcement
6 - MPLS IPv4 Announcement
7 - MPLS IPv6 Announcement
8-255 Reserved

PDU Length: Total number of octets in the PDU including all payloads and fields

Checksum: A 32 bit hash over the message to detect bit flips, see Section 5.2.1.1

PDU Number: 0..255, a monotonically increasing value, modulo 256, see [RFC1982].

Flags: A bit field, the low order bit is number 0

0 - One of a multi-Frame sequence
1 - last of a multi-Frame sequence
2-7 - Reserved

5.2.1.1. The Checksum

There is a reason conservative folk use a checksum in UDP. And as many operators stretch to jumbo frames (over 1,500 octets) longer checksums are the conservative approach.

For the purpose of computing a checksum, the checksum field itself is assumed to be zero.

Sum up 32-bit unsigned ints in a 64-bit long, then take the high-order section, shift it right, rotate, add it in, repeat until zero.

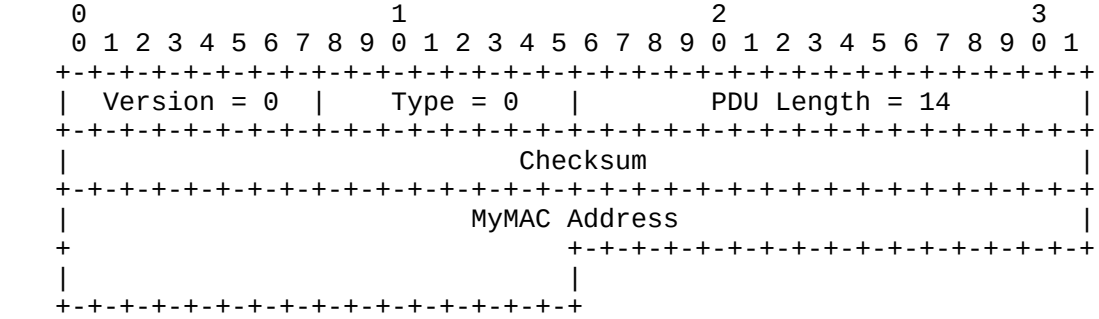
```
#include <stddef.h>
#include <stdint.h>

/* The F table from Skipjack, and it would work for the S-Box. */
static const uint8_t sbox[256] = {
  0xa3,0xd7,0x09,0x83,0xf8,0x48,0xf6,0xf4,0xb3,0x21,0x15,0x78,
  0x99,0xb1,0xaf,0xf9,0xe7,0x2d,0x4d,0x8a,0xce,0x4c,0xca,0x2e,
  0x52,0x95,0xd9,0x1e,0x4e,0x38,0x44,0x28,0x0a,0xdf,0x02,0xa0,
  0x17,0xf1,0x60,0x68,0x12,0xb7,0x7a,0xc3,0xe9,0xfa,0x3d,0x53,
  0x96,0x84,0x6b,0xba,0xf2,0x63,0x9a,0x19,0x7c,0xae,0xe5,0xf5,
  0xf7,0x16,0x6a,0xa2,0x39,0xb6,0x7b,0x0f,0xc1,0x93,0x81,0x1b,
  0xee,0xb4,0x1a,0xea,0xd0,0x91,0x2f,0xb8,0x55,0xb9,0xda,0x85,
  0x3f,0x41,0xbf,0xe0,0x5a,0x58,0x80,0x5f,0x66,0x0b,0xd8,0x90,
  0x35,0xd5,0xc0,0xa7,0x33,0x06,0x65,0x69,0x45,0x00,0x94,0x56,
  0x6d,0x98,0x9b,0x76,0x97,0xfc,0xb2,0xc2,0xb0,0xfe,0xdb,0x20,
  0xe1,0xeb,0xd6,0xe4,0xdd,0x47,0x4a,0x1d,0x42,0xed,0x9e,0x6e,
  0x49,0x3c,0xcd,0x43,0x27,0xd2,0x07,0xd4,0xde,0xc7,0x67,0x18,
  0x89,0xcb,0x30,0x1f,0x8d,0xc6,0x8f,0xaa,0xc8,0x74,0xdc,0xc9,
  0x5d,0x5c,0x31,0xa4,0x70,0x88,0x61,0x2c,0x9f,0x0d,0x2b,0x87,
  0x50,0x82,0x54,0x64,0x26,0x7d,0x03,0x40,0x34,0x4b,0x1c,0x73,
  0xd1,0xc4,0xfd,0x3b,0xcc,0xfb,0x7f,0xab,0xe6,0x3e,0x5b,0xa5,
  0xad,0x04,0x23,0x9c,0x14,0x51,0x22,0xf0,0x29,0x79,0x71,0x7e,
  0xff,0x8c,0x0e,0xe2,0x0c,0xef,0xbc,0x72,0x75,0x6f,0x37,0xa1,
  0xec,0xd3,0x8e,0x62,0x8b,0x86,0x10,0xe8,0x08,0x77,0x11,0xbe,
  0x92,0x4f,0x24,0xc5,0x32,0x36,0x9d,0xcf,0xf3,0xa6,0xbb,0xac,
  0x5e,0x6c,0xa9,0x13,0x57,0x25,0xb5,0xe3,0xbd,0xa8,0x3a,0x01,
  0x05,0x59,0x2a,0x46
};

  /* non-normative example C code, constant time even */

  uint32_t sbox_checksum_32(const uint8_t *b, const size_t n)
  {
    uint32_t sum[4] = {0, 0, 0, 0};
    uint64_t result = 0;
    for (size_t i = 0; i < n; i++)
      sum[i & 3] += sbox[*b++];
    for (int i = 0; i < sizeof(sum)/sizeof(*sum); i++)
      result = (result << 8) + sum[i];
    result = (result >> 32) + (result & 0xFFFFFFFF);
    result = (result >> 32) + (result & 0xFFFFFFFF);
    return (uint32_t) result;
  }
```

5.3. HELLO

The HELLO message is unique in that it is a multicast Ethernet frame. It solicits response(s) from other device(s) on the link. See Section 8.1 for why multicast is used.

All other LSOE messages are unicast Ethernet frames, as the peer's MAC Address is known after the HELLO exchange.

When an interface is turned up on a device, it SHOULD issue a HELLO periodically. The interval is set by configuration.



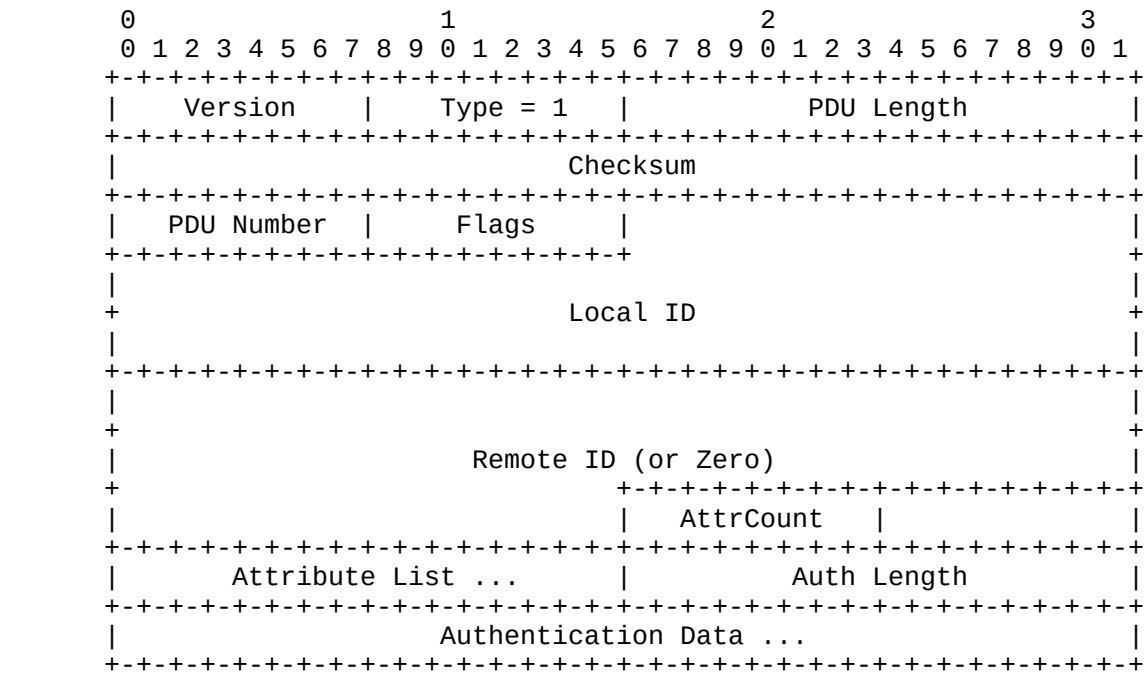
If more than one device responds, one adjacency is formed for each unique (MAC address) response. LSOE treats the adjacencies as separate links.

When a HELLO is received from a MAC address where there is no established LSOE adjacency, the receiver SHOULD respond with an OPEN message. The two devices establish an LSOE adjacency by exchanging OPEN messages.

The PDU Length is the octet count of the entire PDU, including the Version, the Type, the PDU Length field itself, the Checksum, and the following payload.

5.4. OPEN

Each device has learned the other's MAC address from the HELLO exchange, see Section 5.3. Therefore the OPEN and subsequent messages are unicast, as opposed to the HELLO's multicast, Ethernet frames.



An ID can be an ASN with high order bits zero, a classic RouterID with high order bits zero, a catenation of the two, a 80-bit ISO System-ID, or any other identifier unique to a single device in the current routing space. IDs are big-endian.

When the local device sends an OPEN without knowing the remote device's ID, the Remote ID MUST be zero. The Local ID MUST NOT be zero.

AttrCount is the number of attributes in the Attribute List. Attributes are single octets whose semantics are user-defined.

A node may have zero or more user-defined attributes, e.g. spine, leaf, backbone, route reflector, arabica, ...

Attribute syntax and semantics are local to an operator or datacenter; hence there is no global registry. Nodes exchange their attributes only in the OPEN message.

Auth Length is the length in octets of the Authentication Data, not including the Auth Length itself. If there are no Authentication Data, the Auth Length MUST BE zero.

The Authentication Data are specific to the operational environment. A failure to authenticate is a failure to start the LSOE association, and HELLOs MUST BE restarted.

Once two devices know each other’s MAC addresses, and have exchanged OPEN messages, Layer 2 KEEPALIVEs (see Section 5.7) SHOULD be started to ensure Layer 2 liveness and keep the association semantics alive. The timing and acceptable drop of the KEEPALIVE messages SHOULD be configured.

If a properly authenticated OPEN arrives from a device with which the receiving device believes it already has an LSOE association (OPENs have already been exchanged), the receiver MUST assume that the sending device has been reset. All discovered data MUST BE withdrawn via the BGP-LS API and the recipient MUST respond with a new OPEN.

5.5. The Encapsulations

Once the devices know each other’s MAC addresses, know each other’s upper layer identities, have means to ensure link state, etc., the LSOE ’association’ is considered established, and the devices SHOULD announce their interface encapsulation, addresses, (and labels).

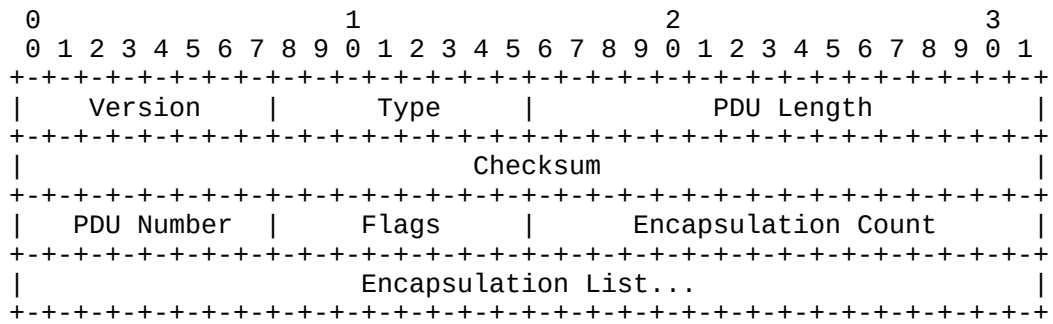
The Encapsulation types the peers exchange may be IPv4 Announcement (Section 5.5.3), IPv6 Announcement (Section 5.5.4), MPLS IPv4 Announcement (Section 5.5.6), MPLS IPv6 Announcement (Section 5.5.7), or possibly others not defined here.

The sender of an Encapsulation message MUST NOT assume that the peer is capable of the same Encapsulation Type. An Encapsulation ACK (Section 5.6) merely acknowledges receipt. Only if both peers have sent the same Encapsulation Type is it safe to assume that they are compatible for that type.

Further, to consider a link of a type to formally be established so that it may be pushed up to upper layer protocols, the addressing for the type must be compatible, i.e. on the same IPvX subnet.

5.5.1. The Encapsulation Message Skeleton

The header for all encapsulation messages is as follows:



If the length of an Encapsulation message exceeds the PDU size limit on media, the message is broken into multiple PDUs. See Section 5.2.

The Encapsulation Exchange is over an unreliable transport so the PDU Number and ACKs are used for reassembly.

The Receiver MUST acknowledge the Encapsulation PDU with a Type=3, Encapsulation ACK PDU (Section 5.6) with the Encapsulation Type being that of the encapsulation being announced, see Section 5.6 and the PDU Number being the PDU Number of the Encapsulation PDU being ACKnowledged.

If the Sender does not receive an ACK in one second, they SHOULD retransmit. After a user configurable number of failures, the LSOE association should be considered dead and the OPEN process SHOULD be restarted.

An Encapsulation message MAY describe multiple addresses of the encapsulation type.

An Encapsulation message of Type T replaces all previous encapsulations of Type T.

To remove all encapsulations of Type T, the sender uses and Encapsulation Count of zero.

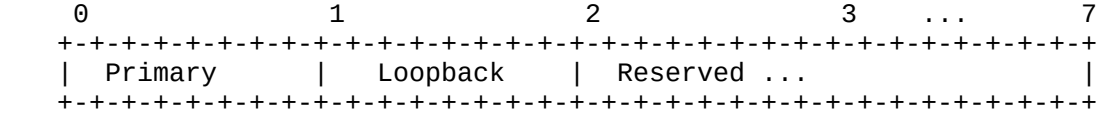
If an interface has multiple addresses for an encapsulation type, one address SHOULD be marked as primary, see Section 5.5.2.

If a loopback address needs to be exposed, e.g. for iBGP peering, then it should be marked as such, see Section 5.5.2.

If a sender has multiple links on the same interface, separate data, ACKs, etc. must be kept for each peer.

Over time, multiple Encapsulation messages may be sent for an interface as configuration changes.

5.5.2. Prim/Loop Flags

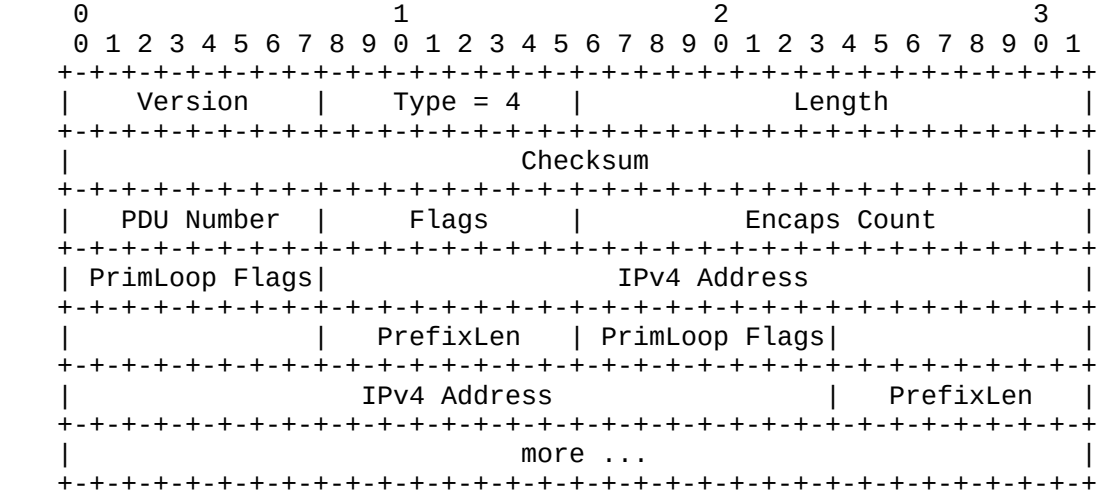


Each Encapsulation interface address MAY be marked as a primary address, and/or a loopback, in which case the respective bit is set to one.

Only one address MAY be marked as primary for an encapsulation type.

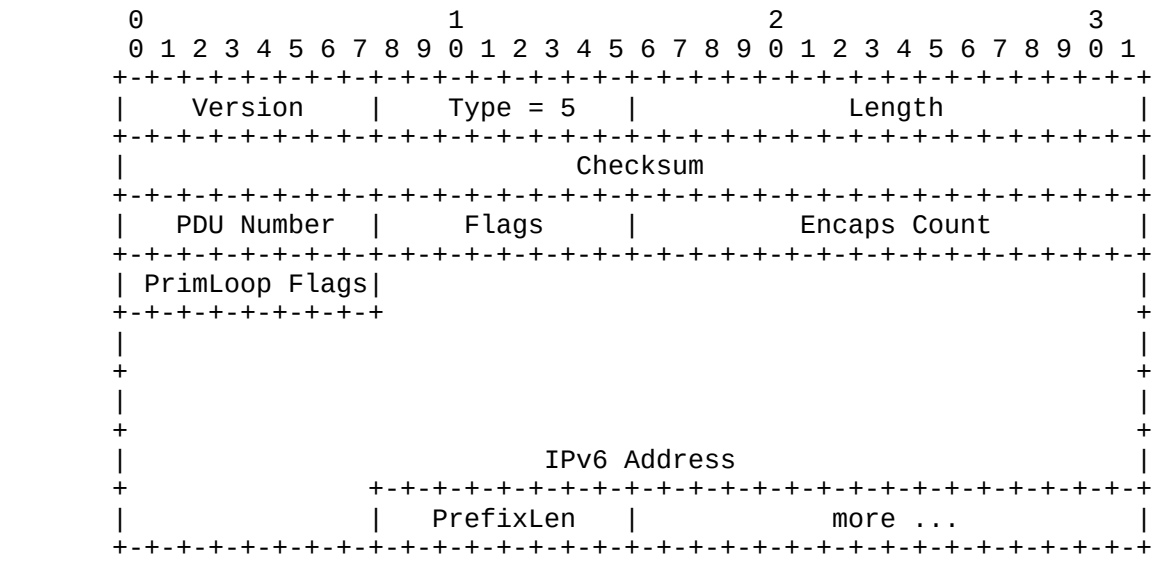
5.5.3. IPv4 Encapsulation

The IPv4 Encapsulation describes a device's ability to exchange IPv4 packets on one or more subnets. It does so by stating the interface's address and the prefix length.



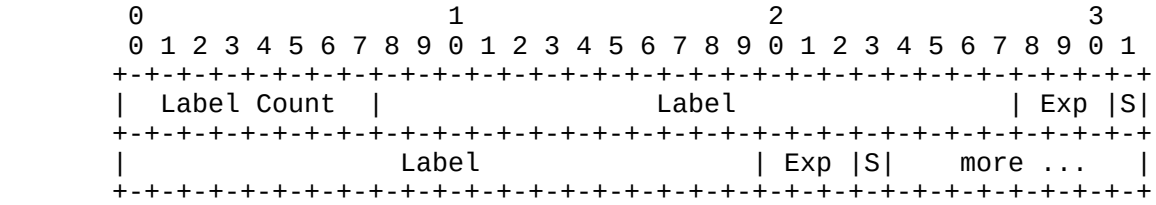
5.5.4. IPv6 Encapsulation

The IPv6 Encapsulation describes a device's ability to exchange IPv6 packets on one or more subnets. It does so by stating the interface's address and the prefix length.



5.5.5. MPLS Label List

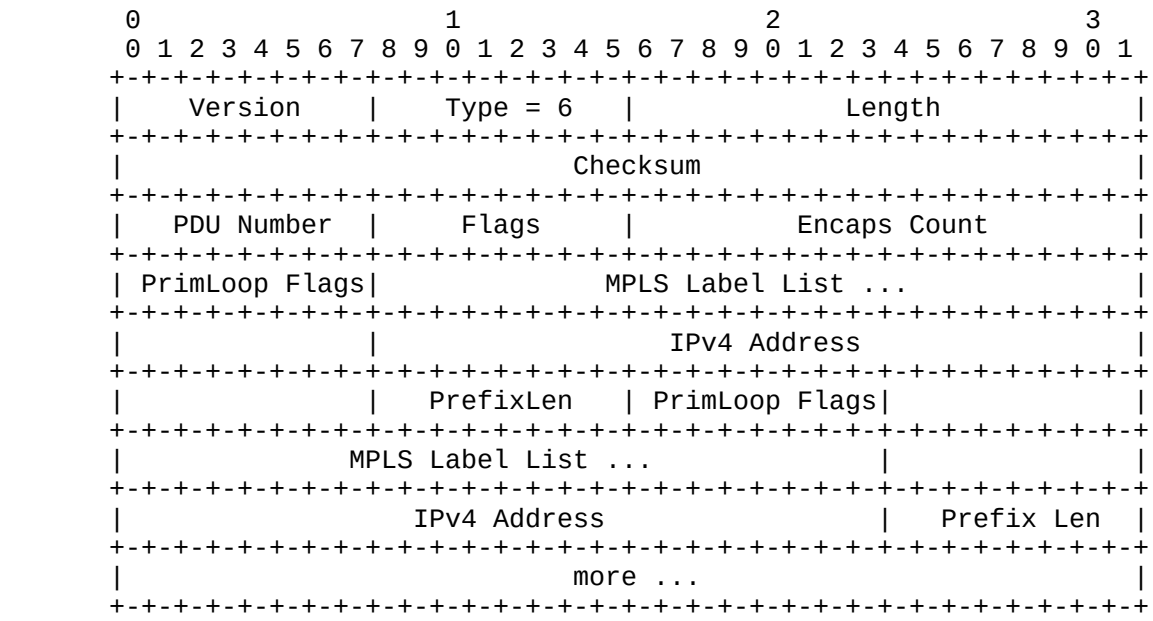
As an MPLS enabled interface may have a label stack, see [RFC3032], a variable length list of labels is needed.



A Label Count of zero is an implicit withdraw of all labels for that prefix on that interface.

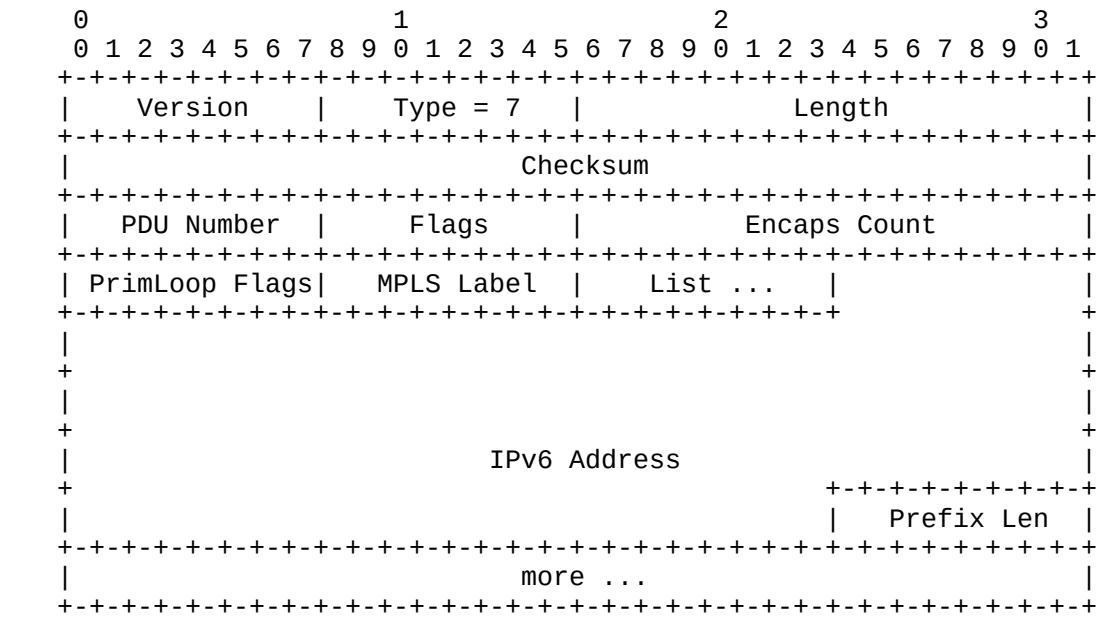
5.5.6. MPLS IPv4 Encapsulation

The MPLS IPv4 Encapsulation describes a device’s ability to exchange labeled IPv4 packets on one or more subnets. It does so by stating the interface’s address and the prefix length.

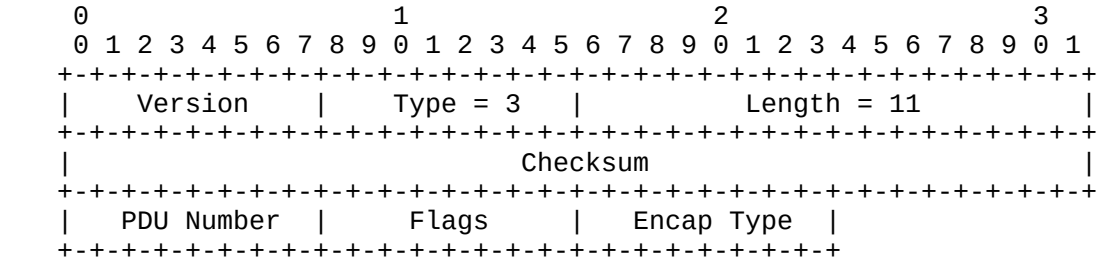


5.5.7. MPLS IPv6 Encapsulation

The MPLS IPv6 Encapsulation describes a device’s ability to exchange labeled IPv6 packets on one or more subnets. It does so by stating the interface’s address and the prefix length.



5.6. Encapsulation ACK



The header is as described in Section 5.5.

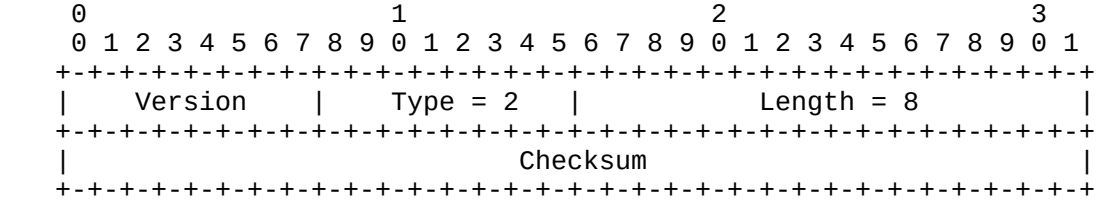
The PDU Number is the number of the received PDU being acknowledged. This provides is a lock-step exchange of an Encapsulation message.

The Encap Type is the Type of the Encapsulation Message being acknowledged.

5.7. KEEPALIVE - Layer 2 Liveness

LSOE devices MUST exchange occasional Layer 2 KEEPALIVE messages to ensure association continuity.

They SHOULD be exchanged at a configured frequency. One per second is the default. Layer 3 liveness, such as BFD, will likely be more aggressive.



6. Layers 2.5 and 3 Liveness

Ethernet liveness is continuously tested by KEEPALIVE messages, see Section 5.7. As layer 2.5 or layer 3 connectivity could still break, liveness above layer 2 SHOULD be frequently tested using BFD ([RFC5880]) or a similar technique.

This protocol assumes that one or more Encapsulation addresses will be used to ping, BFD, or whatever the operator configures.

7. The North/South Protocol

Thus far, a one-hop point-to-point link discovery protocol has been defined.

The nodes know the unique node identifiers (ASNs, RouterIDs, ...) and Encapsulations on each link interface.

Full topology discovery is not appropriate at the Ethernet layer, so Dijkstra a la IS-IS etc. is assumed to be done by higher level protocols.

Therefore the node identifiers, link Encapsulations, and state changes are pushed North via a small subset of the BGP-LS API. The upper layer routing protocol(s), e.g. BGP-SPF, learn and maintain the topology, run Dijkstra, and build the routing database(s).

For example, if a neighbor’s IPv4 Encapsulation address changes, the devices seeing the change push that change Northbound.

7.1. Use BGP-LS as Much as Possible

BGP-LS [RFC7752] defines BGP-like PDUs describing link state (links, nodes, link prefixes, and many other things), and a new BGP path attribute providing Northbound transport, all of which can be

ingested by upper layer protocols such as BGP-SPF; see Section 4 of [I-D.ietf-lsvr-bgp-spf].

For IPv4 links, TLVs 259 and 260 are used. For IPv6 links, TLVs 261 and 262. If there are multiple addresses on a link, multiple TLV pairs are pushed North, having the same ID pairs.

7.2. Extensions to BGP-LS

The Northbound protocol needs a few minor extensions to BGP-LS. Luckily, others have needed the same extensions.

Similarly to BGP-SPF, the BGP protocol is used in the Protocol-ID field specified in table 1 of [I-D.ietf-idr-bgpls-segment-routing-epe]. The local and remote node descriptors for all NLRI are the ID's described in Section 5.4. This is equivalent to an adjacency SID or a node SID if the address is a loopback address.

Label Sub-TLVs from [I-D.ietf-idr-bgp-ls-segment-routing-ext] Section 2.1.1, are used to associate one or more MPLS Labels with a link.

8. Discussion

This section explores some trade-offs taken and some considerations.

8.1. HELLO Discussion

There is the question of whether to allow an intermediate switch to be transparent to discovery. We consider that an interface on a device is a Layer 2 or a Layer 3 interface. In theory it could be a Layer 3 interface with no encapsulation or Layer 3 addressing currently configured.

A device with multiple Layer 2 interfaces, traditionally called a switch, may be used to forward frames and therefore packets from multiple devices to one interface, I, on an LSOE speaking device. Interface I could discover a peer J across the switch. Later, a prospective peer K could come up across the switch. If I was not still sending and listening for HELLOs, the potential peering with K could not be discovered. Therefore, interfaces MUST continue to send HELLOs as long as they are turned up.

8.2. HELLO versus KEEPALIVE

Both HELLO and KEEPALIVE are periodic. KEEPALIVE might be eliminated in favor of keeping only HELLOs. But currently KEEPALIVE is unicast, has a checksum, is acknowledged, and thus more firmly verifies association existence.

This warrants discussion.

8.3. Open Issues

VLANs/SVIs/Subinterfaces

9. Security Considerations

The protocol as is MUST NOT be used outside a datacenter or similarly closed environment due to lack of formal definition of the authentication and authorisation mechanism. These will be worked on in a later effort, likely using credentials configured using ZTP.

Many MDC operators have a strange belief that physical walls and firewalls provide sufficient security. This is not credible. All MDC protocols need to be examined for exposure and attack surface.

It is generally unwise to assume that on the wire Ethernet is secure. Strange/unauthorized devices may plug into a port. Mis-wiring is very common in datacenter installations. A poisoned laptop might be plugged into a device’s port.

Malicious nodes/devices could mis-announce addressing, form malicious associations, etc.

For these reasons, the OPEN message’s authentication data exchange SHOULD be used. [A mandatory to implement authentication is in development.]

10. IANA Considerations

This document requests the IANA create a registry for LSOE Message Type, which may range from 0 to 255. The name of the registry should be LSOE-Message-Type. The policy for adding to the registry is RFC Required per [RFC5226], either standards track or experimental. The initial entries should be the following:

Message Code	Message Name
0	HELLO
1	OPEN
2	KEEPALIVE
3	Encapsulation ACK
4	IPv4 Announce / Withdraw
5	IPv6 Announce / Withdraw
6	MPLS IPv4 Announce / Withdraw
7	MPLS IPv6 Announce / Withdraw
8-255	Reserved

This document requests the IANA create a registry for LSOE Flag Bits, which may range from 0 to 7. The name of the registry should be LSOE-Flag-Bits. The policy for adding to the registry is RFC Required per [RFC5226], either standards track or experimental. The initial entries should be the following:

Bit	Bit Name
0	One of a multi-Frame sequence
1	last of a multi-Frame sequence
2-7	Reserved

This document requests the IANA create a registry for LSOE PL Flag Bits, which may range from 0 to 7. The name of the registry should be LSOE-PL-Flag-Bits. The policy for adding to the registry is RFC Required per [RFC5226], either standards track or experimental. The initial entries should be the following:

Bit	Bit Name
0	Primary
1	Loopback
2-7	Reserved

11. IEEE Considerations

This document needs a new EtherType.

12. Acknowledgments

The authors thank Cristel Pelsser for multiple reviews, Jeff Haas for review and comments, Joe Clarke for a useful review, John Scudder deeply serious review and comments, Larry Kreeger for a lot of layer 2 clue, Martijn Schmidt for his contribution, Rob Austein for reviews

and checksum code, Russ Housley for checksum discussion and sBox, and Steve Bellovin for checksum advice.

13. References

13.1. 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-idr-bgpls-segment-routing-epe]
Previdi, S., Filsfils, C., Patel, K., Ray, S., and J. Dong, "BGP-LS extensions for Segment Routing BGP Egress Peer Engineering", draft-ietf-idr-bgpls-segment-routing-epe-15 (work in progress), March 2018.

[I-D.ietf-lsvr-bgp-spf]
Patel, K., Lindem, A., Zandi, S., and W. Henderickx, "Shortest Path Routing Extensions for BGP Protocol", draft-ietf-lsvr-bgp-spf-02 (work in progress), August 2018.

[RFC1982] Elz, R. and R. Bush, "Serial Number Arithmetic", RFC 1982, DOI 10.17487/RFC1982, August 1996, <http://www.rfc-editor.org/info/rfc1982>.

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

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <http://www.rfc-editor.org/info/rfc3032>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <http://www.rfc-editor.org/info/rfc4271>.

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

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <http://www.rfc-editor.org/info/rfc5880>.

[RFC6286] Chen, E. and J. Yuan, "Autonomous-System-Wide Unique BGP Identifier for BGP-4", RFC 6286, DOI 10.17487/RFC6286, June 2011, <http://www.rfc-editor.org/info/rfc6286>.

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

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

13.2. Informative References

[JUPITER] Singh, A., Germano, P., Kanagala, A., Liu, H., Provost, J., Simmons, J., Tanda, E., Wanderer, J., HAP.lzle, U., Stuart, S., Vahdat, A., Ong, J., Agarwal, A., Anderson, G., Armistead, A., Bannon, R., Boving, S., Desai, G., and B. Felderman, "Jupiter rising", Communications of the ACM Vol. 59, pp. 88-97, DOI 10.1145/2975159, August 2016.

Authors' Addresses

Randy Bush
Arrcus & IIJ
5147 Crystal Springs
Bainbridge Island, WA 98110
United States of America

Email: randy@psg.com

Keyur Patel
Arrcus
2077 Gateway Place, Suite #250
San Jose, CA 95119
United States of America

Email: keyur@arrcus.com