

IDR
Internet Draft
Intended status: Standards Track
Expires: September 10, 2015

Luyuan Fang
Microsoft
Chandra Ramachandran
Juniper Networks
Fabio Chiussi
Cisco Systems
Yakov Rekhter

March 9, 2015

BGP-LU for HSDN Label Distribution
draft-fang-idr-bgplu-for-hsdn-00

Abstract

This document describes the use of BGP Labeled Unicast (BGP-LU) with modified BGP Route Reflector (RR) operation for label distribution in the Hierarchical SDN (HSDN) control plane for the hyper-scale Data Center (DC) and cloud networks.

Status of this Memo

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

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

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

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/lid-abstracts.html>

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

Copyright Notice

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

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

Table of Contents

1. Introduction	3
2. Terminology	5
3. Label Mapping Server/RR	5
3.1. Peer Community	6
4. BGP-LU Procedures	6
4.1. Advertisements from UP Destination	7
4.2. Advertisements from UPBN	8
4.3. Advertisements from LM-RR	8
4.4. Route Resolution	9
5. Illustration of BGP-LU Procedures	9
5.1. UP2-1 Destinations	9
5.2. UP1-1 Destinations	10
5.3. UP0 Destinations	11
5.4. Packets from Server3 to Server1	12
6. Context Labels	13
6.1. UP2-1 Destinations	14
6.2. UP1-1 Destinations	14
6.3. UP0 Destinations	14
7. Security Considerations	15
8. IANA Considerations	15
9. Normative References	15
10. Informative References	15
Authors' Addresses	15

1. Introduction

Hierarchical SDN (HSDN) [I-D.fang-mpls-hsdn-for-hsdc] is an architectural solution to scale a hyper-scale cloud consisting of many Data Centers (DCs) interconnected by a Data Center Interconnect (DCI) to tens of millions of physical underlay endpoints, while efficiently handling both Equal Cost Multi Path (ECMP) load-balanced traffic and any-to-any end-to-end Traffic Engineered (TE) traffic. The HSDN reference model, operation, and requirements are described in [I-D.fang-mpls-hsdn-for-hsdc].

HSDN is designed to allow the physical decoupling of control and forwarding, and have the LFIBs configured by a controller according to a full SDN approach. Such a controller-centric approach is described in [I-D.fang-mpls-hsdn-for-hsdc].

However, HSDN is also meant to support the traditional distributed routing and label distribution protocol approach to distribute the labels. This hybrid approach may be particularly useful during technology migration. This document specifies the use of BGP Labeled Unicast (BGP-LU) for label distribution and LFIB configuration in the HSDN control plane.

In the HSDN architecture, the DC/DCI network is partitioned into hierarchical underlay partitions (UPs) such that the number of destinations in each UP does not increase beyond the limit imposed by capabilities of network nodes. Once the DC cloud has been partitioned to the desired configuration, the traffic from a source endpoint to a destination endpoint uses a stack of labels, one label per each level in the hierarchy, whose semantics indicate to the forwarding network nodes at each level which destination in its local UP should forward the packet to. The label semantics can also identify a specific path (or group of paths) in the UP, rather than simply a destination.

In other words, the label stack indirectly represents the UPs that the packet should traverse to reach the destination end device.

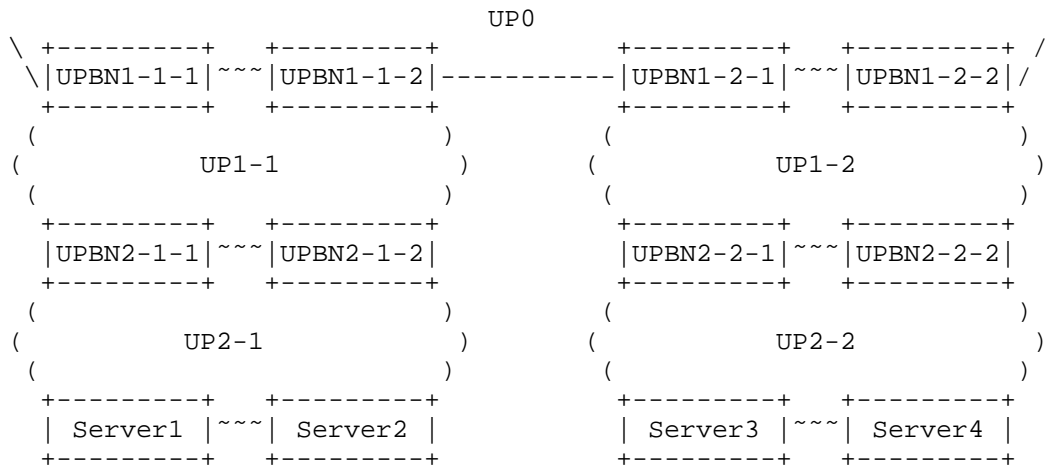


Figure 1 - Example topology with 2 levels of partitioning

Considering the example partitioning in Figure 1, which has 3 levels in the hierarchy, packets from Device3 to Device1 require 3 Path Labels (PLs).

- Top label (PL0) will forward the packet to one of the UPBN1-1 nodes, which are grouped as UPBG1-1 (which is a destination in UP0)
- Next label (PL1) will forward the packet to one of the UPBN2-1 nodes, which are grouped as UPBG2-1 (which is a destination in UP1-1)
- Next label (PL2) will forward the packet to Device1 (which is a destination in UP2-1)

This document proposes BGP-LU based procedures for:

- How UPBN learns the destinations in its UP and the label that should be installed in LFIB to forward traffic to these destinations
- How UPBN learns the context labels used by other UPBN destinations in the partition if the DC operator implements a policy of using locally assigned labels on UPBNs

The procedures specified in the document are applicable to ECMP traffic in HSDN based DC cloud architectures.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

This document inherits the terminology defined in [I-D.fang-mpls-hsdn-for-hsdc] and additionally introduces the following terms that apply when BGP-LU based control plane is used to realize HSDN architecture.

- o RR: BGP Route Reflector.
- o BGP Peer Group: Collection of BGP peers for which a set of policies are applied on a BGP speaker.
- o Label Mapping Server: A node present in each Underlay Partition that allocates labels for destinations in the partition.
- o Label Mapping RR (LM-RR): A modified or customized BGP RR that uses BGP-LU to advertise label bindings for destinations in UP. In other words, Label Mapping RR is an implementation of Label Mapping Server that uses BGP-LU to advertise the labels for partition destinations.
- o Peer Community: An IP based extended community carried in BGP update that represents UPBG of a partition.
- o Route Resolver: A single or a collection of entities that provides the MPLS label stack to reach a destination underlay end device.

3. Label Mapping Server/RR

This document specifies modifications to BGP Route Reflection procedures defined in [RFC4456] in order to provide a mechanism for Label Mapping Server of a UP to distribute labels for UP destinations. These modified procedures are applicable to both BGP RR server as well as the RR client and implementations must activate these procedures based on user-configured policy.

- When LM-RR receives BGP-LU advertisement [RFC3107] (i) whose NLRI and BGP next-hop are the same, and (ii) that contains a valid Peer-Community (specified in Section 3.1), then LM-RR looks up its local database of known Peer community values. If the Peer-Community value is new, then LM-RR adds the newly learnt Peer-community in the database.

- When LM-RR receives IP route (IP version 4 AFI) whose (i) NLRI and BGP next-hop are different, and (ii) BGP next-hop belongs to a known Peer-Community from its local database, then LM-RR performs the following actions. Otherwise, LM-RR behaves like vanilla BGP RR specified in [RFC4456].
 - o LM-RR checks whether a label has been allocated for {UP destination, Peer-Community} pair. If not, LM-RR allocates a label for {UP destination, Peer-Community} and let us call this label PL1.
 - o LM-RR originates a BGP-LU advertisement for the same IP destination containing PL1 in the L-BGP NLRI but does not modify the BGP next-hop attribute from the received IP route. In other words, the modified LM-RR procedures result in LM-RR effectively reflecting a BGP-LU route in response to a vanilla IP route when the above specified conditions are met.

Conceptually, the above procedures result in LM-RR (implementing Label Mapping Server of a UP) allocating a "partition-unique" label for every destination in the partition, and all UPBNs of the partition forwarding MPLS packet with that label to the particular destination in the partition. It should be noted that LM-RR of a UP may allocate labels having any structure that reflects the UP hierarchy as specified in Section 5 of [I-D.fang-mpls-hsdn-for-hsdc].

In other words, while the label allocated by LM-RR is strictly "partition- unique", the DC operator may apply a label allocation policy that would result in same label allocated for same destination across partitions (which may be true for destinations that are present "up" the hierarchy).

3.1. Peer Community

This document introduces a new extended community that enables the receiving iBGP speaker to group the iBGP peers into a community. The application of new extended community in this document is that it allows the receiving BGP speaker to determine which iBGP peers belong to a UPBG.

The details of the new extended community are TBD.

4. BGP-LU Procedures

The BGP-LU based control plane mechanism specified in this document assumes the following set of policies be applied on various network nodes. The policy configurations required are as follows.

- Each UPBN of a UP is configured to be a BGP RR and all UP destinations are configured as clients of UPBN. That is, each end-device in the lowest level UP has one iBGP peering session with each UPBN of the UP. Note that it is not essential that UPBNs are configured as BGP RRs and the same outcomes described from the procedures below may be arrived by not configuring UPBNs as RRs.
- Each UPBN of a non-UP0 partition is also connected to its higher level partition. For example, UPBN_j of UP_j will be connected to UP_i where $i=j-1$. UPBN_j has one iBGP peering session with each UPBN_i, and one iBGP peering session with UP_j destinations. Note that on UPBN_j, iBGP peering sessions with UPBN_i and UP_j destinations are configured to be in different BGP peer groups.
- UPBN_j has a policy to automatically export destinations learnt from UPBN_i peer group to UP_j peer group (where $i=j-1$). But UPBN_j does not export destinations learnt from UP_j peer group to UPBN_i peer group. This export policy on UPBN_j limits the number of BGP advertisements that any network node in UP_i has to process apart from limiting the number of LFIB entries in network nodes.
- If UPBNs learn destinations in its UP and its parent UP using IGP, then UPBN_j runs two separate IGP routing instances one corresponding to UP_j and one corresponding to UP_i (where $i=j-1$). UPBN_j does not leak any route between the IGP instances. Alternatively, UPBN_j may learn destinations in UP_j and UP_i using other mechanisms and such mechanisms are outside the scope of this document.
- Each Underlay Partition (UP) has one Label Mapping RR (or LM-RR) that is responsible for advertising the labels allocated for the destinations in that UP. Each UPBN of the UP is configured with UPBG that it belongs to and has iBGP peering session with LM-RR.
- LM-RR also maintains iBGP peering session(s) with Route Resolver. How Router Resolver is realized is outside the scope of this document, but for the purpose of this document it should be noted that LM-RR of a UP uses this iBGP peering session(s) to advertise the labels allocated for UP destinations to Resolver.

4.1. Advertisements from UP Destination

- Each End-device in a UP advertises a "vanilla" IP route in BGP to all UPBNs of the UP (UPBNs are configured as RRs while UP destinations are configured as clients). The advertisement contains NLRI and BGP nexthop attribute set to the address of the end-device that advertises the route.

- Each UPBN_j that is a destination in UP_i (where $i=j-1$) advertises a BGP-LU route with NULL label to UPBN_i. The advertisement contains NLRI and BGP nexthop attribute set to the address of UPBN_j.
 - o For UP_i, the destinations may be UPBN_j where $j=i-1$.
 - o BGP-LU routes originated by UPBN_j will have "Peer- Community" appended to the route where the "Peer-Community" corresponds to UPBG_j that UPBN_j belongs to.

4.2. Advertisements from UPBN

- When UPBN receives the advertisements from UP destinations (as specified in Section 4.1), UPBN re-advertises these UP destinations to LM-RR.
 - o Each UPBN in the UP also originates a "vanilla" IP route corresponding to each destination in the UP
 - o UPBN removes the "Peer-Community" corresponding to UPBN_j if present in the advertisement received from UP destination.
 - o If the UP destination is learnt from a BGP-LU route, the UPBN removes the label in its re-advertisement to LM-RR.
- UPBN also originates a BGP-LU route with NULL label for itself i.e. with NLRI and BGP nexthop attribute set to self
 - o UPBN_i routes will have "Peer-Community" appended where the "Peer-Community" corresponds to UPBG_i that UPBN belongs to.

4.3. Advertisements from LM-RR

- When LM-RR receives a BGP advertisement that contains "Peer-Community" (that denotes a UPBG), it allocates a label from its label space if a label is not already allocated for the destinations advertised by peers that belong to the UPBG.
 - o LM-RR determines whether the originator is a UPBN by using the route already advertised by the UPBN with "Peer- Community".
 - o LM-RR allocates one label per UPBG per destination. As all UPBNs of a UP belong to one UPBG, LM-RR will allocate N labels for a UP with N destinations.
- LM-RR originates in the UP a BGP-LU route for each "vanilla" IP route learnt from UPBN (see Section 4.2).

- o Label value that is set in BGP-LU route is equal to the label LM-RR has allocated for the UP destination per UPBG.
 - o LM-RR retains the BGP nexthop attribute present in the "vanilla" IP advertisement.
 - o LM-RR also advertises the BGP-LU route to Route Resolver in order to provide enough information to Route Resolver to recursively resolve a remote End-device destination. Note that the method of realizing Route Resolver is beyond the scope of this document.
- When UPBN receives the BGP-LU route (per UP destination) originated from LM-RR, it installs the label in its LFIB and sets up the LFIB entry to forward MPLS packet received with that label to the specific UP destination.
- o If the UP destination has been learnt via a "vanilla" IP route without an associated "Peer-Community", then all ECMP paths in the LFIB entry terminate at the UP destination (which is an End-device in this case).
 - o If the UP destination has been learnt via a BGP-LU route with a corresponding "Peer-Community" then the ECMP paths in the LFIB entry terminate at all lower level UPBNs having the same "Peer-Community".

4.4. Route Resolution

As a consequence of the procedures described in Section 4.1 to 4.3, Route Resolver of the DC cloud will have the knowledge of the destinations in all UPs and the UPBNs that have advertised those UP destinations. Route Resolver uses this information to construct MPLS label stack to forward the packet to desired destination End-device.

The mechanism with which Route Resolver is implemented in the DC cloud is outside the scope of this document.

5. Illustration of BGP-LU Procedures

This sections provides an example of the procedures described in Section 4 using the example topology provided in Figure 1.

5.1. UP2-1 Destinations

UPBN2-1-1 originated routes:

```
{Server1, NH: UPBN2-1-1}
```

{Server2, NH: UPBN2-1-1}

{UPBN2-1-1, NH: UPBN2-1-1, Label: NULL, Peer-Community: UPBG2-1}

UPBN2-1-2 originated routes:

{Server1, NH: UPBN2-1-2}

{Server2, NH: UPBN2-1-2}

{UPBN2-1-2, NH: UPBN2-1-2, Label: NULL, Peer-Community: UPBG2-1}

LM-RR-2-1 originated routes:

{Server1, NH: UPBN2-1-1, Label: PL2-1}

{Server1, NH: UPBN2-1-2, Label: PL2-1}

{Server2, NH: UPBN2-1-1, Label: PL2-2}

{Server2, NH: UPBN2-1-2, Label: PL2-2}

Note that LM-RR has to advertise these routes to Route Resolver also. The method of realizing Route Resolver is beyond the scope of this document.

5.2. UP1-1 Destinations

UPBN1-1-1 originated routes:

{UPBN2-1-1, NH: UPBN1-1-1}

{UPBN2-1-2, NH: UPBN1-1-1}

{UPBN1-1-1, NH: UPBN1-1-1, Label: NULL, Peer-Community: UPBG1-1}

UPBN1-1-2 originated routes:

{UPBN2-1-1, NH: UPBN1-1-2}

{UPBN2-1-2, NH: UPBN1-1-2}

{UPBN1-1-2, NH: UPBN1-1-2, Label: NULL, Peer-Community: UPBG1-1}

LM-RR-1-1 originated routes:

{UPBN2-1-1, NH: UPBN1-1-1, Label: PL1-1}

{UPBN2-1-1, NH: UPBN1-1-2, Label: PL1-1}

{UPBN2-1-2, NH: UPBN1-1-1, Label: PL1-1}

{UPBN2-1-2, NH: UPBN1-1-2, Label: PL1-1}

Note that same label PL1-1 is assigned because UPBN2-1-1 and UPBN2-1-2 belong to the same UPBG2-1.

5.3. UP0 Destinations

UPBN1-1-1 originated routes:

{UPBN1-1-1, NH: UPBN1-1-1, Label: NULL, Peer-Community: UPBG1-1}

UPBN1-1-2 originated routes:

{UPBN1-1-1, NH: UPBN1-1-1, Label: NULL, Peer-Community: UPBG1-1}

UPBN1-2-1 originated routes:

{UPBN1-1-1, NH: UPBN1-2-1}

{UPBN1-1-2, NH: UPBN1-2-1}

{UPBN1-2-1, NH: UPBN1-2-1, Label: NULL, Peer-Community: UPBG1-2}

UPBN1-2-2 originated routes:

{UPBN1-1-1, NH: UPBN1-2-2}

{UPBN1-1-2, NH: UPBN1-2-2}

{UPBN1-2-2, NH: UPBN1-2-2, Label: NULL, Peer-Community: UPBG1-2}

LM-RR-0 originated routes:

{UPBN1-1-1, NH: UPBN1-2-1, Label: PL0-1}

{UPBN1-1-2, NH: UPBN1-2-1, Label: PL0-1}

{UPBN1-1-1, NH: UPBN1-2-2, Label: PL0-1}

{UPBN1-1-2, NH: UPBN1-2-2, Label: PL0-1}

Note that same label PL0-1 is allocated because UPBN1-1-1 and UPBN1-1-2 belong to the same UPBG1-1.

5.4. Packets from Server3 to Server1

1. When Server3 wants to send traffic to Server1, it requests the "Route Resolver" for the label stack and immediate nexthop to send the packet to.
2. Resolver will return the label stack {PL0-1, PL1-1, PL2-1} and nexthop of UPBN1-2-1. As the UPBN destinations in higher level are advertised into lower levels, Server3 can be assumed to have enough information to forward the packet to UPBN1-2-1.
 - a. By configuring UPBN1-2-1 to advertise its address as FEC in UP1-2 that is in turn leaked into UP2-2, Server3 can be assumed to have a LSP to UPBN1-2-1 to forward the MPLS packet having label stack {PL0-1, PL1-1, PL2-1}
 - b. Note that Resolver may also return two nexthops UPBN1-2-1 and UPBN1-2-2 along with the label stack {PL0-1, PL1-1, PL2-1}.
 - c. Alternatively, UPBN2-2-1 and UPBN2-2-2 may be made to learn PL0-1 label that should be forwarded to UPBG1-2. In such a case, the Resolver need not provide nexthop information but only provides label stack {PL0-1, PL1-1, PL2-1} to Server3 and Server3 will always load balance MPLS packets to UPBNs of UP2-2.
3. From the BGP-LU route advertised learnt from UP0 (Section 5.3), UPBN1-2-1 should have already installed in LFIB an entry from PL0-1 with ECMP paths to UPBN1-1-1 and UPBN1-1-2.
 - a. UPBN1-2-1 pops PL0-1 and load balances the packets to UPBN1-1-1 and UPBN1-1-2.
 - b. Before forwarding the MPLS packet with label stack {PL1-1, PL2-1}, UPBN1-2-1 pushes the LSP label for UPBN1-1-1 or

UPBN1-1-2 depending on where the packet is forwarded to.

4. When UPBN1-1-1 or UPBN1-1-2 receives the packet with label stack {PL1-1, PL2-1}, either of them should have already installed an LFIB entry for PL1-1 (Section 5.2). In this example, let us assume that it is UPBN1-1-1 that has received the packet.
 - a. UPBN1-1-1 receives PL1-1 as top of label stack because PHP is assumed to be in place.
 - b. UPBN1-1-1 pops PL1-1 and load balances the packets UPBN2-1-1 and UPBN2-1-2.
 - c. Before forwarding the MPLS packet with label stack {PL2-1}, UPBN1-1-1 pushes the LSP label for UPBN2-1-1 or UPBN2-1-2 depending on where the packet is forwarded to.
5. When UPBN2-1-1 or UPBN2-1-2 receives the packet with label stack {PL2-1}, either of them should have already installed an LFIB entry for PL2-1 (Section 5.1). In this example, let us assume it is UPBN2-1-1 that has received the packet.
 - a. UPBN2-1-1 pops PL2-1 and forwards the packet to Server1.
 - b. Before forwarding the packet, UPBN2-1-1 pushes the LSP label for Server1.
6. Server1 receives the packet that only contains VL if any corresponding to the overlay service instance.

6. Context Labels

The label allocated by LM-RR for a UP destination is "partition-unique" and should be installed in LFIB of all UPBNs of that UP. This assumes that UPBNs of the UP are capable of setting aside some labels to avoid "partition-unique" labels from colliding with UPBN platform label space. With the following modifications to the procedures described in Section 4, this assumption is no longer required.

- If the destination in a UP_i is a UPBN for a lower level UP (say UP_j where j=i+1), then instead of originating BGP-LU route with NULL label in UP_i it originates BGP-LU route with a non-NULL label allocated from its platform label space. The label represents the "context" label space corresponding to UP_j and the LFIB entries in the context LFIB correspond to the UP_j destinations (change to procedure in Section 4.1).
- LM-RR of UP_i should act as vanilla BGP-RR for BGP-LU routes also

and not execute any special procedures (by allocating label) when it receives BGP-LU route from UPBNj.

The consequence of the modified procedures involving context labels is that whenever UPBN forwards MPLS packet to next UPBG, it has to push an additional label to enable the receiving UPBN (in next UPBG) to lookup in appropriate context LFIB for forwarding the packet.

The following example illustrates the changes to the advertisements to the example in Section 5 for the use of context labels.

6.1. UP2-1 Destinations

There is no change in advertisements listed in Section 5.1. This is because the UP2-1 destinations are Servers that are not UPBNs belong to any UPBG.

6.2. UP1-1 Destinations

Apart from the advertisements listed in Section 5.2, the UPBNs of UP2-1 that are the destinations in UP1-1 advertise a BGP-LU route with non-NULL label. So when UPBN1-1-1 and UPBN1-1-2 forward traffic to UPBG2-1, they push either CL2-1 or CL2-2 depending on whether the traffic is forwarded to UPBN2-1-1 or UPBN2-1-2.

UPBN2-1-1 originated routes:

```
{UPBN2-1-1, NH: UPBN2-1-1, Label: CL2-1, Peer-Community: UPBG2-1}
```

UPBN2-1-2 originated routes:

```
{UPBN2-1-2, NH: UPBN2-1-2, Label: CL2-2, Peer-Community: UPBG2-1}
```

6.3. UP0 Destinations

All BGP-LU routes that UPBNs in UP0 originate advertising their own address as destination should contain non-NUL label. The UPBNs should install LFIB entry corresponding to the label advertised to lookup in the context LFIB corresponding to LM-RR-0.

UPBN1-1-1 originated routes:

```
{UPBN1-1-1, NH: UPBN1-1-1, Label: CL0-1-1, Peer-Community: UPBG1-1}
```

UPBN1-1-2 originated routes:

```
{UPBN1-1-1, NH: UPBN1-1-1, Label: CL0-1-2, Peer-Community: UPBG1-1}
```

UPBN1-2-1 originated routes:

{UPBN1-2-1, NH: UPBN1-2-1, Label: CL0-2-1, Peer-Community: UPBG1-2}

UPBN1-2-2 originated routes:

{UPBN1-2-2, NH: UPBN1-2-2, Label: CL0-2-2, Peer-Community: UPBG1-2}

7. Security Considerations

TBD.

8. IANA Considerations

TBD

9. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3107] Rekhter, Y. and E. Rosen, "Carrying Label Information in BGP-4", RFC 3107, May 2001.
- [RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, April 2006.

10. Informative References

- [I-D.fang-mpls-hsdn-for-hsdc] L. Fang, et. al., "MPLS-Based Hierarchical SDN for Hyper-Scale DC/Cloud", draft-fang-mpls-hsdn-for-hsdc-01 (work in progress), March 2015.

Authors' Addresses

Luyuan Fang
Microsoft
5600 148th Ave NE
Redmond, WA 98052
Email: lufang@microsoft.com

Chandra Ramachandran
Juniper Networks
Electra, Exora Business Park Marathahalli - Sarjapur Outer Ring Road
Bangalore, KA 560103

India
Email: csekar@juniper.net

Fabio Chiusi
Cisco
500 108th Avenue N.E., Suite 500
Bellevue, Washington 98004
Email: fchiussi@cisco.com

Yakov Rekhter